

**A LIGHTWEIGHT APPROACH TOWARDS MUTUAL AUTHENTICATION IN A  
SMART GRID ENVIRONMENT**

by

DEBSMITA GHOSH

B.Tech in Computer Science and Engineering, Vellore Institute of Technology University,

Vellore, India, 2013

A thesis

presented to Ryerson University

in partial fulfillment of the  
requirements for the degree of

Master of Applied Science

In the Program of

Computer Networks

Toronto, Ontario, Canada, 2015

©Debsmita Ghosh 2015

## **Author's declaration**

DEBSMITA GHOSH

2015

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.

# A Lightweight Approach towards Mutual Authentication in a Smart Grid Environment

©Debsmita Ghosh, 2015

Master of Applied Science  
in Computer Networks  
Ryerson University

## **Abstract**

Smart grids allow automated meter readings and facilitate two-way communications between the smart meters and utility control centers. As the smart grid becomes more intelligent, it becomes increasingly vulnerable to cyber-attacks. Smart grid security mainly focuses on mutual authentication and key management techniques. An impeding factor in grid security is the memory and processing constraints of the smart meters. The aim of this thesis is to propose a lightweight mutual authentication protocol with an effective key renewal mechanism between a residential smart meter and a gateway. The authentication protocol proposed in the thesis, guarantees source authentication, data integrity, message confidentiality, as well as non-repudiation. The security analysis renders this protocol robust against several attacks. Furthermore, its performance analysis provides meticulous results as to how the proposed protocol is efficient in terms of computation overhead, average delay and buffer occupancy at the gateway.

## **Acknowledgements**

Firstly, I would like to express my sincere gratitude to my supervisor Dr. Truman Yang for the insightful advice and continuous support of my Master's research. His constant motivation and involvement helped me throughout the process of doing my thesis.

Besides my supervisor, I would like to thank my Program Director, Dr. Ngok-Wah Ma, for giving me this opportunity and providing constant encouragement.

My deepest gratitude to my loving family who made this journey easy for me with their constant support and love.

# Contents

<b>List of Figures</b> .....	<b>vii</b>
<b>List of Tables</b> .....	<b>viii</b>
<b>List of Abbreviations</b> .....	<b>ix</b>
<b>Chapter 1: Introduction</b> .....	<b>1</b>
1.1 Smart Grid .....	1
1.2 Advantages of Smart Grid Technology .....	1
1.3 Importance of Lightweight Authentication Protocol .....	5
1.4 Motivation .....	9
1.5 Thesis Contributions .....	10
<b>Chapter 2: Background Knowledge</b> .....	<b>12</b>
2.1 Advanced Metering Infrastructure .....	12
2.2 Demand Response .....	12
2.3 Topology of Smart Grid .....	13
2.4 Smart Grid Communications .....	15
2.4.1 HAN Communication Protocol.....	16
2.4.2 BAN Communication Protocol.....	17
2.4.3 WAN Communication Protocol.....	18
2.5 Cryptography Concepts .....	19
2.5.1 ID-based Cryptography .....	19
2.5.2 Bilinear Pairing .....	21
2.5.3 Zero-knowledge Password Proof (ZKPP) .....	22
2.5.4 Secure Remote Password Protocol (SRP) .....	23
2.5 Related Works .....	25
<b>Chapter 3: Proposed Mutual Authentication Protocol</b> .....	<b>31</b>
3.1 Setup Phase .....	35
3.2 Pre-authentication Protocol .....	36
3.3 Description of Pre-authentication Protocol .....	37
3.4 Mutual Authentication Protocol .....	39

3.5 Description of Authentication Protocol.....	40
3.6 Key Renewal Protocol .....	45
3.6.1 Key Renewal Mechanism .....	45
3.6.2 Description of Key Renewal Mechanism .....	46
3.7 Security Analysis .....	49
<b>Chapter 4: Performance Analysis .....</b>	<b>57</b>
4.1 Communication Overhead .....	59
4.2 Average Delay .....	62
4.3 Buffer Occupancy .....	65
<b>Chapter 5: Conclusion.....</b>	<b>68</b>
<b>Bibliography .....</b>	<b>70</b>

## List of Figures

Figure 2.1 NIST Conceptual Reference Model for Smart Grid.....	15
Figure 2.2 Hierarchy of smart meters .....	16
Figure 3.1 Pre-Authentication Phase Exchange.....	30
Figure 3.2 Mutual Authentication Protocol Exchange .....	31
Figure 3.3 Key Renewal Mechanism.....	41
Figure 4.1 OpenSSL Algorithm Processing Delay .....	58
Figure 4.2 Communication Overhead.....	61
Figure 4.3 Average decryption/verification delay .....	64
Figure 4.4 Average delay experienced for proposed protocol.....	64
Figure 4.5 Buffer occupancy at BAN GW.....	67

## List of Tables

Table 1.1 Approved Hashing Algorithms .....	7
Table 1.2 Approved Symmetric Key Algorithms .....	7
Table 1.3 Approved Asymmetric Key Algorithms .....	7
Table 3.1 Notation used in proposed protocol .....	28
Table 3.2 Hash Functions used in the protocol .....	31
Table 3.3 Notation for Key Renewal Protocol .....	40
Table 4.1 Parameter Notation for Performance Analysis .....	53
Table 4.2 Computational costs of proposed protocol .....	53
Table 4.3 Computational costs of Nicanfar <i>et al.</i> protocol .....	53
Table 4.4 Encryption/Signature in the proposed protocol .....	56
Table 4.5 Encryption/Signature in the Nicanfar <i>et al.</i> protocol .....	57
Table 4.6 Simulation Parameters .....	59



## **List of Abbreviations**

3G	3 <sup>rd</sup> Generation
AES	Advanced Encryption Standard
AMI	Advanced Metering Infrastructure
BAN	Building Area Network
BDHP	Bilinear Diffie-Hellman Protocol
CA	Certificate Authority
CRL	Certificate Revocation List
DER	Distributed Energy Resources
DLP	Discrete Logarithmic Protocol
DR	Demand Response
EAP	Extensible Authentication Protocol
ECC	Elliptic Curve Cryptography
ECDSA	Elliptic Curve Digital Signature Algorithm
EDGE	Enhanced Data rates for Global Evolution
EIBC	Enhanced ID-based Cryptography
EPPDR	Efficient Privacy Preserving Demand Response
FTP	File Transfer Protocol
GW	Gateway
HAN	Home Area Network
HSPDA	High Speed Downlink Packet Access
HTTP	HyperText Transfer Protocol
IBC	ID-based Cryptography
IEC	International Electrotechnical Commission (ITU-TM 3000)
IEEE	Institute of Electrical and Electronics Engineers
ISO	International Organization for Standardization
kB	kilobytes
KGS	Key Generating Server
kV	kilovolts
kW	kilowatts

LTE	Long Term Evolution
LTR	Long Term Refreshment
MATLAB	Matrix Laboratory
MB	megabyte
MD5	Message Digest 5
MDMS	Meter Data Management System
MTR	Medium Term Refreshment
mW	milliwatts
NAN	Neighborhood Area Network
NIST	National Institute of Standards and Technology
NISTIR	National Institute of Standards and Technology Interagency Report
PKG	Private Key Generator
PKI	Public Key Infrastructure
PLC	Power Line Carriers
PRNG	Pseudo Random Number Generators
RAM	Random Access Memory
RF	Radio Frequency
RSA	Rivest, Shamir, and Adelman
SAML	Security Assertions Markup Language
SAS	Security and Authentication Server
SEP	Smart Energy Profile
SHA	Secure Hash Algorithm
SRP	Secure Remote Password
SSH	Secure SHell
SSL	Secure Sockets Layer
STR	Short Term Refreshment
TCP/IP	Transmission Control Protocol/Internetworking Protocol
TDES	Triple Data Encryption Standard
TLS	Transport Layer Security
TOU	Time-of-Use
WAN	Wide Area Network

WiFi	Wireless Fidelity
WiMAX	Worldwide Interoperability for Microwave Access
WMN	Wireless Mesh Network
WPAN	Wireless Personal Area Networking
XOR	Exclusive OR
ZKPP	Zero Knowledge Password Proof

# **Chapter 1**

## **Introduction**

### **1.1 Smart Grid**

The coexistence of intelligent devices and the traditional power grid is termed as smart grid technology. Smart grid follows a distributed mode of control over the power system, as opposed to the centralized approach adopted by the traditional grid. The NIST 3.0 framework, released in October 2014, mentions that the smart grid is the inclusion of communication and information technologies to the traditional power grid, and enabling duplex communication between smart meters and utility control centers [2].

The protection system provides security and privacy protection services, failure management and also performs detailed grid reliability analysis. The smart grid provides a lot of benefits over the traditional power grid. At the same time, it is vulnerable to a multitude of cyber-attacks, unlike the traditional power grid. For instance, with relation to grid security, NIST believes that two-way communication mechanism between smart meters and other electric devices, is also its Achilles' heel.

### **1.2 Advantages of Smart Grid Technology**

Below are some of the main advantages of smart grid over traditional power grid.

#### **a) Distributed approach**

The traditional grid has a centralized approach towards power generation; it consists of conventional power stations. These power stations generate electricity by consuming gas, nuclear energy and coal, as well as by solar plants and hydroelectric dams. As mentioned earlier, traditional

power grids are always located long distances away from habitation areas. Hence, the generated electricity is transmitted over long distances before reaching the end users. Being centralized may prove to be costly if a generator or power plant fails because each power plant is large-scale and feeds electricity to a vast number of customers. In that case, the electricity generated by other power plants will be redirected to the customer base experiencing a deficiency of electricity. As a result, the customers, directly supplied by that power plant, face a drop in electricity availability. Hence, a cascading effect is initiated by the failure of a single power plant. In contrast, smart grid use a distributed approach to power generation. As a result, failure of a generator/power plant is managed by the distributed sources of renewable energy. For instance, surplus energy provided by solar panels and eco-friendly sources are tapped into during periods of high energy demand.

#### **b) Distributed and Renewable Energy Resources**

Distributed Energy Resources (DER) are small, grid-connected devices that generate or store distributed energy. These systems are located close to their consumption sources, hence their capacities do not exceed 10 megawatts (which is a measure of output of power plants). It is decentralized, easily expansible and modular. When compared to the traditional grid, smart grid is more flexible, reliable and better adapted to compensate for grid failures. DERs are small-scale and if any one fails then the remaining DERs can smoothly manage power supply to the whole grid. Moreover, DERs consist of renewable sources of energy. For example, biomass and biogas, solar and wind power, hydro, geothermal power are common DERs. Hence, smart grid promotes the usage of renewable energy sources, and reduces dependence on exhaustible sources of energy. The smart grid definitely has large power plants providing majority of the electricity supply. But the inclusion of DERs helps to lower environmental impacts, lower carbon emissions significantly

and assist in the increasing demand for electricity, thus reducing the load on conventional power systems.

### **c) Advanced Metering Infrastructure (AMI)**

The bidirectional exchange communication between the smart meters and utility control centers is called AMI. Unlike the manual reading of meters in the traditional grid, AMI allows automated reading of smart meters. Also, these readings are regularly conveyed from the smart meters to the utility control centers. Furthermore, if needed, the utility control centers can also send control commands to the smart meters. These commands may be remote connect/disconnect commands, firmware updates or pricing signals. Some smart meters also have in-home displays that enable the consumer to see the amount of electricity consumed. Traditional power grids do not allow this kind of dynamic, two-way approach between the utility and meters. Electricity alone was transmitted from the utility towards the meter. All other functions such as connection/disconnection and meter reading were done manually.

### **d) Demand Response**

The in-home devices consume electricity. This usage is measured by the smart meter located in the house. The Home Area Network smart meter then conveys these meter readings to the Building Area Network gateway at regular intervals. The frequency of sending meter reports may vary from once every 15 minutes to once every hour. The Building Area Network gateway is a collector node. It collects meter readings from several smart meters and then sends it to the utility control center. Sometimes, there is another collector node present between the utility control centers called the Neighborhood Area Network (NAN). In this case, the Building Area Network gateway forwards its meter readings to the Neighborhood Area Network gateway, which transmits them to the control center. Upon reaching the utility control center, the meter reports are stored by the

Metering Data Management System (MDMS). The consumer is billed for power usage based on these reports. Also, these readings provide information to the Metering Data Management System to monitor power usage during various periods in a day. A database of meter reports help to analyze consumer behavior. It provides valuable insight regarding electricity consumption during day and night, across different seasons and also regions. These results are then used as input to frame Demand Response policies. Demand Response helps to attain a balance between the production and consumption of electricity. It, thus, helps to reduce the load on generation by distributing the power consumption across periods of high peak and low peak. Furthermore, renewable resources of energy aid in generating energy which is stored in reserves. This stored energy is tapped during peak periods.

#### **e) Time-of-Use (TOU) Pricing**

Demand Response policies and Time-of-Use techniques help curb energy consumption. Electricity is made expensive by increasing its price during peak periods. This is done to dissuade people from consuming energy unless it is a necessity. Also, during off-peak periods, electricity rates are reduced to encourage people to use energy during those periods. For instance, use of washing machines may be postponed until after 9pm when the generation surpasses the consumption. This technique helps to distribute energy usage across the entire day. This is done to help consumers use electricity in a distributed manner so that generation and consumption may be balanced.

### **1.3 Importance of Lightweight Authentication Protocol**

If an active adversary is successful in obtaining and manipulating the meter readings, he may alter the readings to reflect incorrect usage. If this happens on a large-scale, it will significantly hamper the restricted energy resources and the economy as well. A passive adversary, on the other hand, may collect reports for a long duration of time for a specific house. By analyzing the meter

readings, the attacker will be able to understand the number of occupants in the house, the time at which the house is empty or the occupants are asleep, and other information describing the activity occurring inside the house. The attacker may use this information to launch an attack on the house. Hence, meter readings are extremely sensitive and must be protected. A limiting factor is the memory and processor capabilities of the smart meter device. For instance, a HAN smart meter configuration may comprise of MSP430-F4270 microcontroller along with 128 KB of flash and RAM memory [17].

Efficient protocols and mutual authentication schemes are already in use in the smart grid industry, but they also incur additional overhead. A few instances that increase overhead are long key sizes, ciphers and certificates, maintenance of Public Key Infrastructure (PKI), keeping track of Certificate Revocation Lists and timers. Furthermore, as the grid becomes smarter, it becomes increasingly vulnerable to software attacks. Smart meter devices depend on communication protocols such as TCP/IP, HTTP and FTP to exchange data. By default, these protocols do not have security built into them [18]. These conditions highlight the need for a lightweight authentication protocol between smart meters. Extensive research is being conducted in devising lightweight approaches using techniques such as Diffie-Hellman, ECC-based cryptography and ID-based cryptography. Encryption is insufficient to ensure complete security to the meter readings. A combination of encryption techniques, hashing function, authentication mechanisms, as well as key renewal schemes provide a well-rounded security protocol for a system.

The NISTIR 7628 Guidelines for Smart Grid Cyber Security has a list of certain hashing functions as well as symmetric and asymmetric key algorithms valid until and after the year 2030 [17]. A detailed view of these is presented below:



Table 1.1  
Approved Hashing Algorithms

<b>Algorithm</b>	<b>Secure Hash Algorithm (SHA)</b>
Key Lengths (valid from 2011 until 2029)	SHA-224 is acceptable
Key Lengths (currently used and valid after 2030)	SHA-256, SHA-384 and SHA-512 are acceptable

Table 1.2  
Approved Symmetric Key Algorithms

<b>Algorithm</b>	<b>Advanced Encryption Standard (AES)</b>	<b>Triple Data Encryption Standard (TDES)</b>
Key Lengths (valid from 2011 until 2029)	AES-128. AES-192 and AES-256 are acceptable	3-key TDES (security strength necessary is 112 bits or higher)
Key Lengths (currently used and valid after 2030)	AES-128. AES-192 and AES-256 are acceptable	Not approved after 2030

Table 1.3  
Approved Asymmetric Key Algorithms

<b>Algorithm</b>	<b>RSA digital signature</b>	<b>Elliptic Curve Digital Signature Algorithm (ECDSA)</b>
Key Lengths (valid from 2011 until 2029)	RSA-2048 is acceptable	ECDSA2 with curves P, K and B of key sizes 224, 233 and 233 respectively

Table 1.3 (continued)  
Approved Asymmetric Key Algorithms

Key Lengths (currently used and valid after 2030)	RSA-3072 is acceptable	ECDSA2 with curves P of size 256, 384 and 521, curves K of sizes 283, 409 and 571, and curves B of sizes 283, 409 and 571
--	------------------------	---

Bekara et al. [19] propose an ID-based authentication mechanism for AMI architecture. This authentication mechanism guarantees authentication, non-repudiation and integrity, while preserving end-customer's privacy. A combination of symmetric key cryptography with ID-based cryptography is used to provide source authentication. The scheme introduces several trusted entities in the smart grid topology. These trusted entities are actually Private Key Generators for implementing IBC. Each PKG issues a certificate with an expiration date to the remaining Private Key Generators. Hence, the System Setup phase involves creation and exchange of Private Key Generator certificates. Furthermore, ECDSA is used to generate signatures over these Private Key Generator certificates. The second phase is the Private Key Generation phase where the smart grid entities such as gateways and smart meters contact Private Key Generators to obtain their ID-based public/private key pair. Lastly, Data Source Authentication phase first establishes mutual authentication between the two entities and then data transfer commences between them. The disadvantages of this scheme include certificate retrieval by the entities and certificate maintenance at the directories. This increases the overhead for the authentication process. Apart from this, the usage of ECDSA involves signature generation and verification over the certificates. This authentication scheme has been made on the assumption of including additional Private Key Generators of higher memory and processing capabilities to the smart grid environment.

The paper proposed by Nicanfar et al. [6] proposes a mutual authentication scheme between a HAN smart meter and an authentication server. The proposed protocol use Secure Remote Password protocol and decrease the number of steps in SRP from five to three. The proposed protocol also reduces the number of exchanges from four to three. It is essentially based upon Enhanced ID-based Cryptography (EIBC). EIBC is a scheme developed by the Nicanfar et al. in a previous publication titled ‘EIBC: Enhanced Identity-Based Cryptography, a conceptual design’. EIBC essentially uses True Random Number Generator and Pseudorandom Number Generator to keep changing the secret master key of the PKG, along with the public/private keys of the meters. The authentication scheme is robust against several attacks. Also, the key renewal mechanism is efficient in terms of refreshing the public/private and multicast keys. However, it requires synchronization of three timers between the smart meter and authentication server. This adds to the overhead of the protocol. Also, as mentioned previously, having two random number generators means memory consumption for storing the generator states. Hence, the protocol doesn’t favor scaling in smart grid environment.

Fouda et al. [3] proposes a lightweight mutual authentication protocol and generates a shared session key on the basis of computational Diffie-Hellman exchange protocol. The protocol is applicable between the HAN smart meters and Building Area Network gateway, each of which have a public/private key pair issued by a CA. The paper describes the protocol steps after the HAN smart meter and Building Area Network gateway have extracted and verified their certificates. Fouda et al. use computational Diffie-Hellman scheme to establish mutual authentication. The generated shared session key is then combined with hash-based authentication code techniques to authenticate messages between the two entities. The proposed protocol is successful in establishing a semantically secure shared key in the mutual authentication

environment. The main disadvantage of this protocol is usage of RSA protocol to establish authentication. The involvement of CA and CRLs is a costly process for devices with limited resources such as HAN smart meters.

In EPPDR [8], Li et al. propose a protocol that uses homomorphic encryption to attain secure demand response exchanges in a smart grid environment. The protocol achieves forward secrecy, by renewing the users' key after appropriate intervals. It also achieves entity authentication, and message integrity and confidentiality. Homomorphic encryption is a method in which plaintext is encrypted using algebraic expression. Li et al. combine homomorphic encryption with pairing-based cryptography to create the mutual authentication process. In this paper, authentication process is applicable between control center and Building Area Network, as well as between HAN and Building Area Network. Once two entities have successfully established a session key between each other, then message exchange commences. The messages are signed using ID-based signature mechanism. A drawback of the protocol is the absence of explicit key confirmation. As the session key is generated separately at the two entities, it is advisable to confirm the key before commencing message exchange.

## **1.4 Motivation**

A smart grid is considered secure if the meter readings reach the correct destination, which maybe a collector or utility control center, without being intercepted or captured by adversaries. This is possible by implementing an efficient authentication protocol between the smart meter and the collector or utility control center. In smart grid, an authentication protocol should offer availability, integrity, confidentiality, and non-repudiation. A distinguishing feature of smart grid security is the sensitive nature of the meter readings. If the meter readings become available to adversaries, that could lead to catastrophic results. Hence, a tradeoff is struck between the level of security and

lightweight protocol. The second distinguishing feature of smart grid security is the need for a lightweight protocol owing to limited capability of smart meters. The aim should also be to reduce the packet processing times at the Building Area Network or Neighborhood Area Network collectors. The main focus of smart grid security is to provide sufficient security for reliable meter readings transmission, while keeping the computation and memory consumption costs as low as possible.

## 1.5 Thesis Contributions

The proposed authentication protocol describes an efficient lightweight scheme to provide mutual authentication between the HAN smart meter and Building Area Network gateway. This scheme provides source authentication, data integrity, message confidentiality, and non-repudiation as well. In addition, a key renewal scheme is also described, which ensures forward secrecy. The proposed protocol is secure against Replay, Man-in-the-Middle, Known Session Key, Impersonation and Key Control Attacks. On comparison with the efficient mutual authentication protocol proposed by Nicanfar *et al.* [6], the proposed protocol utilizes lesser number of computation operations, while achieving the same results in terms of message security. To be specific, the main difference between these two protocols is that the proposed scheme uses Pairing-based Cryptography, whereas the other protocol uses Enhanced ID-based Cryptography (EIBC). In addition, the proposed protocol is compared with ECDSA, which is currently used in smart meter authentication. The parameters of comparison between these two schemes are as follows: communication overhead, average delay, and buffer occupancy. In each case, the proposed protocol proves to be more lightweight and efficient. Firstly, the proposed protocol incurs a communication overhead of 98 bytes, whereas ECDSA incurs 255 bytes (mainly owing to the ECDSA signature and certificate). Secondly, the Building Area Network gateway has an average

delay of 0.01ms and 0.05 seconds in the proposed protocol and the ECDSA scheme respectively. Lastly, using ECDSA, the Building Area Network gateway exhausts its 1128 KB buffer while handling an incoming message rate of 100 messages every 15 minutes across a simulation period of 8 hours. On the other hand, the proposed protocol consumes 775 KB in the same simulation environment. Hence, proving that it is scalable as well as lightweight.

## **Chapter 2**

### **Background Knowledge**

#### **2.1 Advanced Metering Infrastructure (AMI)**

Advanced metering infrastructure (AMI) is the name of the architecture that allows automated, two-way communication of information and control commands between a smart meter and a utility company. The main significance of AMI is to inform the utility companies about the real-time meter readings and allow consumers to manage their electricity consumption effectively based on time based pricing. Traditional power grids allow electricity to travel from the power plant to the smart meters. In addition to this, smart grids allow the meters to send their meter readings to the utility control centers, as opposed to the manual reading of the meters in the traditional power grid.

Smart grids also enable the utility control centers to send control commands to the smart meters. These commands are mostly dynamic. For instance, utility companies offer services to disconnect power if a certain usage threshold is exceeded. Customers might opt for such services to be alerted of their usage. This is another advantage of the smart grid because such services help customers to restrict their usage and lower their electricity bills, thus conserving energy. This occurs when the MDMS within the utility control center reads the meter readings, compares it against the threshold and immediately sends a disconnect command to the concerned smart meter. The electric supply is immediately restored upon the customer's request.

#### **2.2 Demand Response**

The Federal Energy Regulatory Commission defines demand response (DR) as: “Changes in electric usage by end-use customers from their normal consumption patterns in response to

changes in the price of electricity over time, or to incentive payments designed to induce lower electricity use at times of high wholesale market prices or when system reliability is jeopardized.” DR is an essential part of the smart grid; it is also a major difference between the traditional power grid and smart grid. As discussed earlier, MDMS analyzes the meter readings and generates reports which estimate the peak and off-peak periods, energy consumption across different seasons and different times in a single day. These readings provide valuable information to the DR to understand the energy usage pattern of the end users. DR deliberately changes the prices of electricity consumption across different time periods. This is known as time based pricing. Pricing rate of electricity consumption per kW is low during off-peaks, and high during on-peak hours. This time based pricing works as an incentive for many customers to use their electric appliances during off-peak hours. For instance, many homes use washing machines at night. This also works to the advantage of the utility company as the disparity between the demand and supply of electricity narrows, and this helps in reducing the load on the power generators. Furthermore, the excess electricity generated by DERs and stored at the MDMS, is tapped when demand exceeds production.

## **2.3 Topology of Smart Grid**

The topology of the smart grid has been adapted from the NIST Conceptual Reference Model for Smart Grid. Smart grid architecture consists of four main domains: generation, transmission, distribution and consumers.



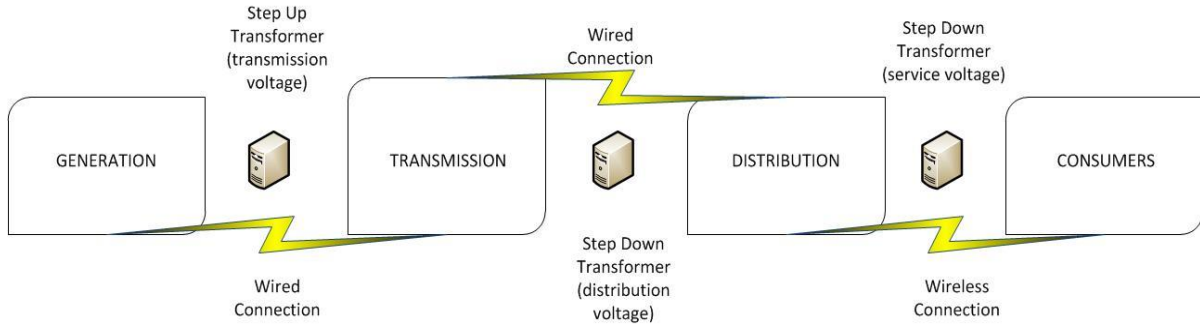


Figure 2.1: NIST Conceptual Reference Model for Smart Grid

The generation domain consists of the large-scale power plants and small-scale DERs that generate electricity. This is followed by the transmission domain consisting of step-up transformers (transmission voltage), transmission substations and transmission lines that aid in transmitting the electricity to the next domain, the distribution domain. The distribution domain consists of step-down transformers (distribution voltage and service voltage), distribution substations and distribution lines. Lastly, the consumers include the smart meters at the homes or businesses that directly use electricity. Consumers may be residential or commercial. Electrical sensors and circuit breakers are placed along the entire length of the communication medium between smart meters and generators to constantly monitor voltage and flow.

Smart meters also have a hierarchy of their own. The lowest level of the hierarchy consists of the meters installed at the home/business, and is called the Home Area Network (HAN) smart meter. Several HAN smart meters regularly send their meter readings to a designated Building Area Network (BAN) gateway, which is the next level in the hierarchy. Lastly, a number of Building Area Network gateway send the collection of meter readings to the Neighborhood Area Network (NAN) gateway. The NAN gateway then forward these meter readings to the utility center. The utility centers are located in the distribution substations.

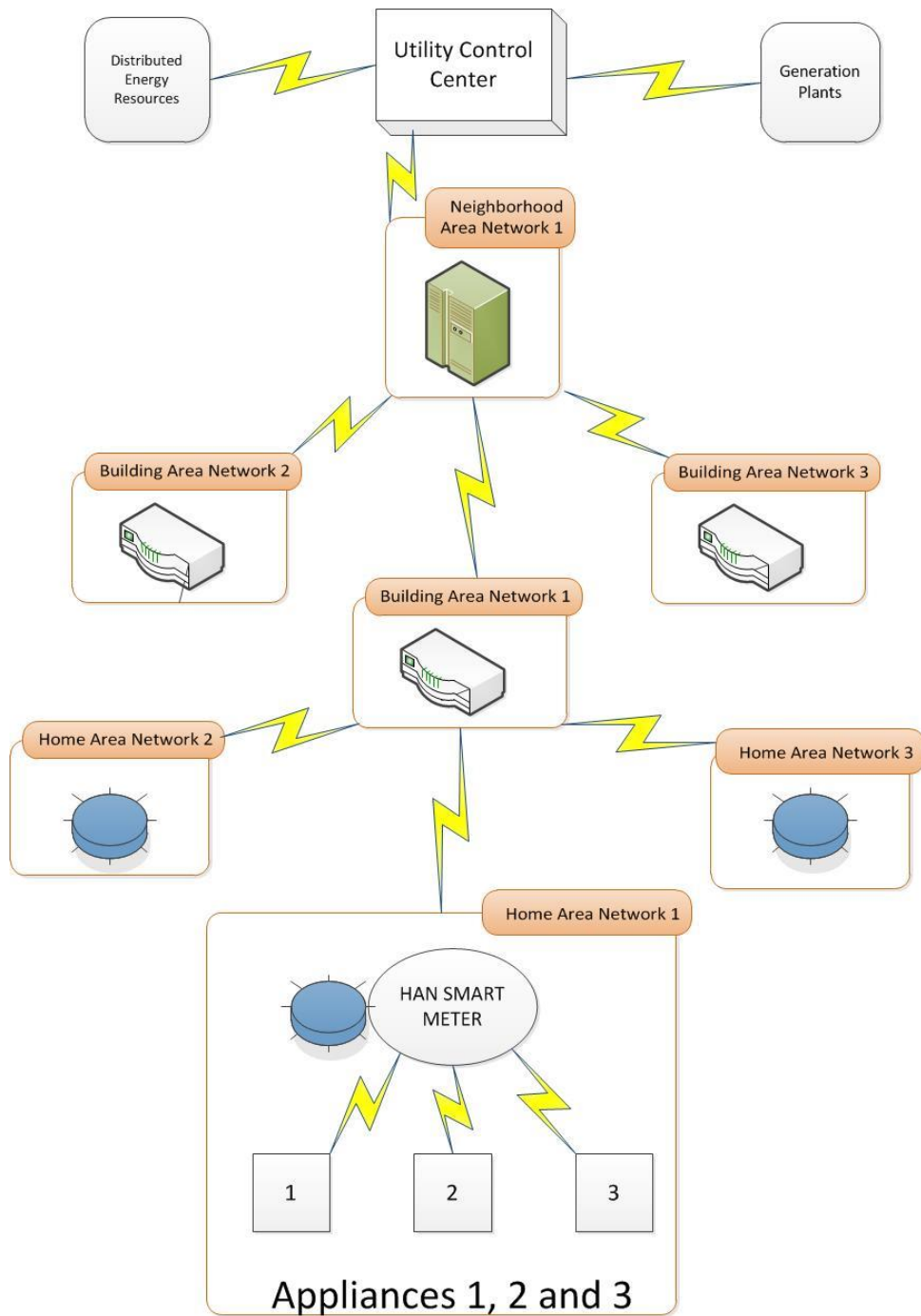


Figure 2.2 Hierarchy of smart meters

## 2.4 Smart Grid Communications

The communication technology used in the smart grid is a combination of wireless and wired technology. The generation and transmission domains are entirely based on wired technology such

as optical fiber or power line carriers (PLC). Optical fiber technology is advantageous because it is flexible, suitable for the core network, and capable to carry high volume of traffic with the least latency [3]. The consumer domain favors wireless technology to communicate with the distribution domain. The distribution domain consists of wireless technology at the end connected to the consumer domain, and wired technology for the end connected to the transmission domain. The Smart Grid environment primarily consists of three areas – HAN, BAN and WAN.

#### **2.4.1 HAN Communication Protocol**

The HAN consists of the home appliances and the smart meter recording their electricity consumption. The communication protocol connecting the smart meter with the corresponding home devices requires an average range of tens of meters. The potential technologies that fulfill this requirement are ZigBee, Wi-Fi, Ethernet, Z-Wave and PLC.

The electric appliances communicate with the HAN smart meter mostly through ZigBee wireless technology. ZigBee is referred to as IEEE 802.15.4 standard. ZigBee is preferred over other wireless technologies such as Wi-Fi or Bluetooth because it consumes the least amount of power and delivers high performance. It is apt for small, low-cost and low-power devices, such as HAN smart meters. HAN smart meters are extremely constrained in terms of memory and computation power. IEEE 802.15 is for Wireless Personal Area Networking (WPAN).

In North America, majority of the smart meters deployed have ZigBee radios. ZigBee is built with low-power sensors, and has lower processing and memory requirements when compared to Wi-Fi or Bluetooth. Also, ZigBee provides the maximum power offered by any unlicensed 2.4 GHz spectrum, which is 100 mW. ZigBee is a mesh network and devices having ZigBee radios possess the ability to self-heal in case of a network fault. The ZigBee Smart Energy Profile provides utilities with an open standard for implementing HAN communications [11]. SEP helps

define the standard working of HAN smart meter devices like thermostats and in-home displays. SEP version 2, which is to be released, focusses on interoperability between Wi-Fi and ZigBee standards. The main advantages of ZigBee are its simplicity, scalability, ability to support thousands of devices on a single network, and extreme tolerance towards interference when compared against radio devices, such as Bluetooth and Wi-Fi [11].

### **2.4.2 BAN Communication Protocol**

The Building Area Network connects the gateway to several HAN smart meters. The range of communication in this area is in the range of hundreds of meters. ZigBee, Wi-Fi, PLC and cellular technologies may be used in this area. RF mesh and 3G (EDGE and HSDPA) are commonly favored because of their extensive coverage [10]. Wi-Fi and WiMAX are preferred over PLC because they are cost-effective and flexible. If the distance between the Building Area Network gateway and HAN smart meter exceeds the coverage distance for the Wi-Fi, then WiMAX is used. Moreover, WiMAX is faster than Wi-Fi. Wireless Mesh Networks are gaining in popularity.

Traditionally, networks consist of a few wired access points or wireless hotspots to provide connectivity to users. WMNs connect hundreds of wireless mesh nodes across a large area. Wireless devices are fitted with radio transmitters which enable them to function like a wireless router. Commonly used Wi-Fi standards are 802.11a, b and g. WMNs boast of several advantages.

#### **a) Dynamic Routing**

They employ dynamic routing. Whenever a new wireless node gets activated, it immediately joins the surrounding wireless network. The routing protocol running within this wireless network enables the exchange of routing tables between the new node and its neighbors. Hence, the new node gains knowledge about how to reach destinations in the fastest manner. The nodes automatically choose the quickest and safest path and this process is known as dynamic routing.

### **b) Cost-effective**

In a WMN, only a single node needs to be physically wired to the Internet. That is a huge advantage compared to other wireless technologies. WMNs are cost-effective as there is wiring or cabling costs. They operate upon pre-existing Wi-Fi standards so new technology is not needed to set up WMN connections.

### **c) Self-configuring & self-healing**

Mesh networks are self-configuring. Inclusion of new nodes does not require any manual effort. These networks are self-healing as well. The network automatically selects the best route to a destination based on the existing topology. Any faults in the network is automatically taken in consideration and the next best route is adopted for packet delivery.

## **2.4.3 WAN Communication Protocol**

The WAN is the network connecting the Building Area Network gateway with the utility control centers. As the coverage distance needed is in tens of kilometers, potential technologies are Ethernet, microwave, WiMAX, 3G/LTE, and fiber optic links. Wired technologies are typically favored in the network connection between the Building Area Network gateways and the utility control centers, and further on to the transmission and generation sites. The reason being that wired connections are more robust and secure compared to the wireless connections. Wireless connections are favored primarily because of their low cost and scalability. At the same time, they are more prone to security attacks and faults. As the smart grid network beyond the Building Area Network gateway is mostly permanent, hence fiber optics serve as the best mode of communication in the WAN [3].

## **2.5 Cryptography Concepts**

The proposed mutual authentication protocol uses a combination of various cryptographic concepts such as pairing-based cryptography, ID-based private and public keys, Zero Knowledge Password proof (ZKPP) and Secure Remote Password protocol (SRP). The resulting authentication protocol ensures confidentiality and data integrity, as well as mutual meter authentication and forward secrecy of session keys. A key renewal mechanism has also been proposed which uses valid timers and sequence numbers to prevent replay attacks. Forward secrecy is ensured and Known Session Key attacks are prevented by incorporating hash of previous message and disposing previous session keys.

### **2.5.1 ID-based Cryptography**

Adi Shamir introduced the concept of identity-based cryptography in 1984. This technique replaces traditional digital certificates with unique identifying attributes, such as email addresses or phone numbers, for encryption and signature verification [12]. Elimination of digital certificates simplifies the process of authenticating users. Prior to identity-based cryptography, a node wanting to communicate with other nodes would have to issue a request to the Certificate Authority (CA) by providing proof of identity. The CA would then verify its identity and send a CA-signed certificate to that node. After this, the node can send its certificate to any node it wants to communicate with. The receiver node, upon receiving the certificate, will forward it to its CA to verify its authenticity. If verification is successful, then the receiver node authenticates this node. Hence, one-way authentication is achieved. Similarly, the process is repeated by the receiver node. The sender node verifies the receiver node's certificate to authenticate it. If the receiver node is who it really claims to be, then the certificate is approved and mutual authentication is established. Unfortunately, this process is time-consuming and memory and computation intensive. Hence, it

is not recommended for limited-resource devices such as smart meters. Moreover, the number of message exchanges and maintenance of additional lists such as Certificate Revocation List (CRL) increase overhead.

IBC replaces the CA with a Private Key Generator (PKG). Before the system nodes enter into mutual authentication processes with one another, the PKG generates a master private-public key pair. The master public key is distributed to all the system nodes. The following procedure describes the encryption and decryption using IBC:-

1. **Encryption Process:** Node A wants to send message M to node B. As node identifiers are openly distributed in such a system, therefore node A uses node B's identifier and the master public key to encrypt M. This produces the cipher text C. Node A sends C to node B. The ease of using IBC is that node A did not have to make prior arrangements to be able to send a message to node B, unlike in the traditional certificate process. Node A already knew about node B's identifier and the master public key.
2. **Decryption Process:** Upon receiving C from A, node B now has to contact the PKG to get hold of its secret private key to be able to decrypt C. Node B sends sufficient proof to the PKG to prove its identity. For instance, if the unique identifier was an email address, then the PKG would send a nonce on that email address. If the same nonce was then sent from the node B to the PKG, then the PKG would accept that node B is who it really claims to be. The PKG then transmits node B's private key to it over a secure channel. This secure channel may be an SSL link that allows node B to download its private key. Node B is now able to successfully decrypt C to obtain plaintext M.

The following procedure describes the identity-based signature process:

1. **Signature:** Node A wants to send a signed message to node B. Upon receiving its private key from the PKG, node A creates a signature  $S$  for message  $M$  and sends it to node B, along with the plaintext message  $M$ . Signature ensures data integrity as well as non-repudiation of a message. In other words, because the message is signed with a private key and private keys are secret, hence the sender cannot deny having sent the signed message.
2. **Verification:** Upon receiving  $M$  and  $S$  from node A, node B applies node A's identifier and the master public key on  $M$ . If the generated signature is the same as  $S$ , then node B accepts the message  $M$ . Else, it rejects message  $M$ .

### 2.5.2 Bilinear Pairing

Bilinear degenerate maps are mathematical functions, which when used in combination with ID-based cryptography, produces computationally efficient cryptographic systems. A bilinear map is a pairing function which produces a mapping of elements from one cyclic group to another cyclic group, provided both cyclic groups are of the same prime order. The discrete log problem of the first group is hard.

Bilinear maps are considered to be secure because they are chosen as one-way functions. In other words, it is easy to calculate the result from a known set of pair of elements, but it is hard vice-versa. This property is known as the Bilinear Diffie-Hellman Assumption because the Bilinear Diffie-Hellman problem is reducible to the inverse operation of bilinear mapping functions [13]. A generic bilinear mapping function  $e$  is  $e(m, A, n, B) = e(n, A, m, B)$ .

It is to be noted that the  $.$  operator represents the multiplication operation between the two operands. Once again, it is easy to perform a multiplication operation between  $m$  and  $A$ , but it is computationally infeasible to determine the value of  $m$ , when  $A$  and  $m.A$  are known [12].



Weil and Tate pairings are popular ID-based bilinear pairing systems. In 2001, Boneh and Franklin first proposed the concept of using ID-based cryptography with bilinear maps. This came to be known as the Weil pairing. Ever since the Weil pairing first emerged, ID-based and bilinear mapping cryptography started gaining momentum and now a combination of the two is known as pairing-based cryptography [12].

Let  $G$  be an additive cyclic group of prime order  $q$ , and  $G_T$  be a multiplicative cyclic group of prime order  $q$ ; let  $g$  be the generator of these cyclic groups. In order to build cryptographic systems, pairing-based cryptography utilizes a symmetric bilinear pairing between two elements of an additive group to an element of a multiplicative group. In the proposed protocol, we have used the map  $e: G_1 \times G_2 \rightarrow G_T$ , where  $G_1 = G_2 = G$ . This mapping satisfies the properties stated below:

1. Bilinearity:  $\forall x, y \in \mathbb{Z}_q^*, \forall A, B \in G: e(A^x, B^y) = e(A, B)^{xy} \in G_T$  (where  $\mathbb{Z}_q^*$  is a set of integers of prime order  $q$ )
2. Non-degeneracy:  $e(A, B) \neq 1$
3. Computability: there exists an efficient algorithm to compute  $e$

Another measure of non-degeneracy is that for all  $B \in G$ , if  $e(A, B) = 1$ , then  $A = \infty$ . We refer to [1] for a more detailed description on bilinear pairing and its applications.

### **2.5.3 Zero-knowledge Password Proof (ZKPP)**

Zero-knowledge password proof is a technique in which node  $A$  (prover) proves to node  $B$  (verifier) that it possesses knowledge of a password without actually knowing the password. This possession of knowledge about the password works as a verification that the node may be trusted. The password belongs to node  $B$  and never leaves node  $B$ . Node  $B$  generates a verifier related to this password and conveys this verifier to all nodes it wants to communicate with. This technique

works as an advantage for systems using password-authenticated key agreement (PAKE) protocol because it is robust against off-line dictionary attacks, as is mentioned in IEEE P1363.2. In IEEE P1363.2, ZKPP is defined as “An interactive zero knowledge proof of knowledge of password-driven data shared between a prover and the corresponding verifier.”

In the proposed protocol, the HAN smart meter has a password. It wants to prove its identity to the Building Area Network gateway using this password but without disclosing it. Hence, it generates a corresponding verifier to that password. Passwords are not unique but to ensure uniqueness of the verifiers, HAN smart meters' identifiers are hashed with the passwords. This is to because Building Area Network gateways store verifiers for several smart meters, and a unique verifier will increase the robustness of the protocol against attacks. In the pre-authentication phase, where the HAN smart meter registers itself with the Building Area Network gateway, the HAN will convey its identifier and verifier through a secure communication channel. Once the authentication process commences, the HAN smart meter and Building Area Network gateway calculate values using the verifier independently. The values generated at both these ends are compared and if discrepancies exist, then the authentication process is aborted.

#### **2.5.4 Secure Remote Password Protocol (SRP)**

SRP is also a modified password-authenticated key agreement protocol. SRP is more secure than SSH protocol, and faster than Diffie-Hellman key exchange in terms of user authentication and data integrity. Compared to Kerberos protocol, SRP doesn't rely on third parties. The SRP is instantiated with the client node selecting a small random salt. The client node also shares a password with the server node. The client node generates a variable  $x$  which stores the hash of the password and salt. Also, to further create the verifier related to that password, the client node performs an exponentiation operation on the generator of a multiplicative group  $g$  using variable

x. It should be mentioned that all arithmetic operations are within integers modulo  $N$ ,  $Z_N$ . In other words,  $g^x$  is actually  $g^x \bmod N$ , where  $N=2q+1$  ( $N$  and  $q$  are both prime numbers). It is extremely important to generate a large  $N$  to increase the field space for integers making number guessing difficult for the intruders.

The verifier and salt values are stored in a table at the server node. As the server serves multiple client nodes, therefore the client node's identifying username is used as an index to retrieve the corresponding salt and verifier. At the client node, the variable  $x$  is discarded because it is equivalent to the password. The aforementioned steps occur before the client node enters in an exchange to register itself with the server node. When the exchange protocol commences, the client node will start by sending its identifying username and variable  $A$  (generated using random number  $a$ ) to the server node. The server node will index its table using the client node's identifying username to retrieve its related salt and verifier. It then selects a random number  $b$  and incorporates that with the verifier and  $k$  to generate a variable  $B$ . Value  $k$  is a hash of  $N$  and  $g$ , generated by both client and server nodes. The importance of  $k$  lies in the fact that it prevents 2-for-1 guesses during impersonation attacks on server nodes by intruders. The server node sends the salt and the variable  $B$  to the client node. Both client and server nodes then generate variable  $u$  which is the hash of  $A$  and  $B$ .

The client node generates a value using  $a$ ,  $B$  and common variables such as  $u$ ,  $x$  and  $k$ . It then creates the hash for the generated value. Similarly, the server node also generates value using  $b$ ,  $A$ ,  $v$  and  $u$ . It further creates a hash of this value. The registration process is validated on the grounds that both the hash values are equal. This is because despite using different sets of variables, the values generated at both the ends is the same. Hence, their respective hashes are also the same. The following protocol provides the exchange steps in details:-

1. Client  $\longrightarrow$  Server : Identifying username and  $A = g^a$
2. Server  $\longrightarrow$  Client : salt and  $B = kv + g^b$
3. Both:  $u = H(A, B)$
4. Client:  $C = (B - kg^x)^{(a+ux)} = (kv + g^b - kg^x)^{(a+ux)} = (kg^x + g^b - kg^x)^{(a+ux)} = (g^b)^{a+ux}$
5. Client:  $K_{client} = H(C)$
6. Server:  $S = (Av^u)^b = (g^av^u)^b = (g^a(g^x)^u)^b = (g^{a+ux})^b = (g^b)^{a+ux}$
7. Server:  $K_{server} = H(S)$

At the end of the exchange protocol, the client and server nodes now have a symmetric session key. The two nodes need to explicitly confirm that their keys match in order to complete authentication process.

SRP has several advantages. Firstly, a node is able to prove its identity to another node using the concept of ZKPP. Secondly, SRP is resistant to dictionary attacks. Thirdly, it is resistant to brute force attacks because a passive adversary or man-in-the-middle is unable to access the password during the authentication protocol because the password is never exchanged. Furthermore, without gaining the password, the attacker is never able to impersonate any nodes. Hence, impersonation attacks are resisted. It is important to mention that SRP version 6 allows a single password guess per connection. In addition to this, SRP version 6 is also used for strong authentication in SSL/TLS, EAP and SAML. It is in the process of being standardized in IEEE P1363 and ISO/IEC 11770-4.

## 2.6 Related Work

The proposed mutual authentication protocol has been evaluated by two methods. The first mode of evaluation is a comparison between the proposed protocol and the protocol proposed by Nicanfar *et al* [6]. This comparison is drawn based on number of computation operations used in

each protocol. The number of encryption/decryption and signature operations have also been compared. The second mode of evaluation compares the performance of the proposed protocol against that of ECDSA-256 based on factors such as computation overhead incurred, memory consumed and delay experienced at the Building Area Network smart meter.

This section evaluates the performance of the proposed protocol by comparing it with a smart grid mutual authentication protocol, put forward by Nicanfar *et al* [6]. The protocol proposed by Nicanfar *et al* is an efficient mutual authentication and key management scheme between a HAN smart meter and an authentication server. Nicanfar *et al* [6] improves on the Secure Remote Password protocol, and uses Enhanced Identity-Based Cryptography in its key-renewal mechanism.

Table 4.1  
Parameter Notations for Performance Analysis

$T_{bm}$	Time taken to execute a bilinear map operation
$T_{mul}$	Time taken to execute a scalar multiplication operation
$T_{add}$	Time taken to execute an addition operation
$T_{sub}$	Time taken to execute a subtraction operation
$T_{xor}$	Time taken to execute an XOR operation
$T_{exp}$	Time taken to execute a modular exponentiation operation
$T_{pow}$	Time taken to execute a power operation
$T_h$	Time taken to execute a hash function

Table 4.2  
Computational costs of the proposed protocol

	<b>HAN side</b>	<b>BAN side</b>
Proposed protocol	$5T_h + 2T_{exp} + 1T_{bm} + 1T_{mul} + 1T_{sub}$	$4T_h + 1T_{exp} + 3T_{mul} + 1T_{bm} + 1T_{add} + 2T_{pow}$

Table 4.3  
Computational costs of Nicanfar *et al* protocol

	<b>SM side</b>	<b>SAS side</b>
Nicanfar <i>et al</i> protocol	$10T_h + 2T_{exp} + 1T_{xor} + 1T_{sub} + 1T_{mul} + 1T_{add} + 1T_{pow}$	$8T_h + 1T_{exp} + 3T_{mul} + 1T_{add} + 1T_{xor} + 3T_{pow}$

Tables 4.2 and 4.3 presents the computational cost between the two authentication protocols. The need for a lightweight scheme rises because of the limited computational capabilities of the HAN SM. The Building Area Network gateway or authentication server, on the other hand, has much higher computational capabilities. The Building Area Network gateway has 128 KB of RAM and 1 MB of flash memory. To summarize, with regards to the HAN side/SM side, the proposed protocol uses lesser number of operations when compared to the protocol proposed by Nicanfar *et al*. The aforementioned statement can be proven by several instances. To begin with, the proposed protocol uses 5 hash operations, whereas the other protocol uses 10 hash operations. Hash operations are one of the least computationally intensive operations, but keeping in mind the memory constraints of the HAN smart meter, it is best to save memory and CPU power under any circumstances. The number of scalar multiplication, modular exponentiation and subtraction operations used in both protocols is the same. In addition to this, the proposed protocol has a bilinear mapping operation. On the other hand, the protocol proposed by Nicanfar *et al* has an XOR, addition and power operations.

Coming to the Building Area Network side/SAS side, the proposed protocol uses 4 hash function operations and 2 power operations, as compared to the protocol proposed by Nicanfar *et*

*al*, which uses 8 hash functions and 3 power operations. The number of addition, scalar multiplication and modular exponentiations between the two protocols is the same. In addition to this, the proposed protocol has a bilinear mapping operation, whereas the other protocol has one XOR operation. At this point, it is important to mention that in case of similar operations, the protocol proposed in this thesis is always using a lesser number of operations as compared to the other protocol. Even though hash operations and addition/subtraction operations are relatively inexpensive, the proposed protocol uses far less number of these operations. This measure helps to relieve the load off devices having memory and computation constraints, such as HAN smart meters that have a RAM of 6 to 8 KB alone.

A bilinear pairing operation is definitely more computationally intensive than the above mentioned operations. At the same time, the addition of bilinear mapping to this protocol is a necessity for several reasons. Firstly, bilinear mapping operations are extremely secure; the reason being that they are one-way functions. Hence, it is easy to calculate the product of a pair of operands, but it is hard to do the inverse. So, it is an easy operation to choose two operands from a group and map their result to a multiplicative group. But the probability of an adversary to locate these two operands from the product is close to negligible. Hence, the session key is protected from any chances of reproduction at the adversary's end. The multiplication operation used within the bilinear mapping also has its significance. For instance, we have  $e(a \cdot X, b \cdot Y) = e(b \cdot X, a \cdot Y)$  as the bilinear map. The multiplication operation  $a \cdot X$  is extremely easy, but finding a given  $X$  and  $a \cdot X$  is computationally infeasible.

The proposed protocol is based on IBC whereas the other protocol is based upon EIBC. EIBC has been developed by Nicanfar *et al*. Both protocols use a random number generator in order to generate the system parameters needed for calculating public private keys. This random

number generator is located within the PKG, which is the Building Area Network gateway in the proposed protocol. The protocol proposed by Nicanfar et al also uses a second random number generator. In other words, the entities such as smart meters also hold a pseudorandom number generator. In EIBC, the master secret key generated by the PKG is dependent on the random values returned by both the generators located at PKG as well as the entities. A PRNG requires memory consumption to maintain the state of each iteration. Moreover, if the entities are in the habit of processing meter readings frequently then that means higher number of memory cycles.

As far as key refreshment is concerned, the proposed protocol uses a combination of sequence number, valid periods and hash of the previous key. The HAN SM itself chooses valid period, as it is more regular in sending messages than the Building Area Network gateway. The Building Area Network GW simply receives this value from the HAN SM and processes messages according to its clock. The only information to be maintained by the HAN SM is the sequence number. Similarly, the Building Area Network GW is also expected to store a sequence number for each HAN SM that reports to it. Sequence numbers always start from the value 0. They hardly take any memory and the only operation involved is incrementing the counter by 1 every time. Moreover, because of the manner in which the proposed protocol is designed, there is no possibility of the sequence number being equal to a large integer value as it is reinitialized to the value 0 at a constant interval. Hence, sequence numbers are not going to occupy a vast amount of memory for storing or processing.

The key refreshment process for the other protocol is indeed a very intensive process. It involves 3 timers. The first timer STR marks the transition of the system moving into a new session state. The second timer MTR ensures that the PKG has selected a new set of variables by running its PRNG. Lastly, every time that the timer LTR goes off, the PKG reselects its master secret key



and the shared secret values. The maintenance of three timers is not an easy task as it involves the clocks of the PKG and the entities to be synchronized to each other. Hence, the PKG has to tune its clock to match that of each of the other entities. Any occurrence of clock drift in an entity will lead to the timers going off at different times for the PKG and the related entity. Furthermore, if the clock drift occurs within the PKG, then that will lead to disrupted services with each of the entities. Hence, three timers involve perfect synchronization amongst all the participating devices, which is a hindrance in the long-term maintenance of the smart grid network.

Table 4.4  
Encryption/Signature in the proposed protocol

	<b>HAN side</b>	<b>BAN side</b>
Encryption/Decryption	1 Encryption	1 Decryption
Signcryption	1 Signcryption	1 Verify

Table 4.5  
Encryption/Signature in Nicanfar *et al* protocol

	<b>SM side</b>	<b>SAS side</b>
Encryption/Decryption	1 Encryption	1 Decryption
	1 Decryption	1 Encryption
Sign/Verify	1 Sign	1 Verify
	1 Verify	1 Sign

As is shown in Table 4.4 the proposed protocol ensures mutual authentication with one encryption/decryption operation, and one signcryption operation. On the other hand, Table 4.5

shows that the other protocol has 2 encryption/decryption operations, and 2 sign/verify operations. As these operations are expensive for the resource-constrained smart meters, hence the proposed protocol proves to be more lightweight by using the common principles of SRP protocol, ID-based cryptography, bilinear mapping and ZKPP.

## Chapter 3

### Proposed Mutual Authentication Protocol

The proposed protocol ensures a lightweight mutual authentication and key renewal mechanism between the HAN smart meter and the Building Area Network gateway. It also provides confidentiality, authentication and integrity; the three essential requirements for Smart Grid security as mentioned by NIST [2]. In addition to this, forward secrecy is maintained during the key renewal process. The underlying framework upon which this protocol is built includes a combination of various mathematical and cryptography concepts. These are as follows:-

1. ID-based Cryptography
2. Zero Knowledge Password Proof
3. Secure Remote Password Protocol
4. Pairing-based Cryptography

Table 3.1  
Notation used in the proposed protocol

$ID_{HAN}$	identifier for the HAN SM
$ID_{BAN}$	identifier for the BAN GW
$G$	Additive cyclic group of prime order $q$
$G_T$	Multiplicative cyclic group of prime order $q$
$s$	master secret key of the BAN GW
$SK_{HAN}$	private key of HAN SM
$PK_{BAN}$	public key of BAN GW

Table 3.1 (continued)  
Notation used in proposed protocol

Pwd	password of the HAN SM
$x$	output of a hash function on salt, pwd and $SK_{HAN}$
$G$	generator of the additive cyclic group
$V$	password verifier at the HAN SM
$A$	random number selected by HAN SM
$B$	random number selected by BAN GW
$Z_q^*$	number space containing integers of prime order $q$
$H()$	hash function
$A$	variable storing an exponentiation operation using $\alpha$ at HAN SM
$B$	variable storing an exponentiation operation using $\beta$ at BAN GW
$E_k[msg]$	encryption operation on message msg using key $k$
$Q$	prime order of the cyclic groups (it is also a Sophie Germain prime)
$N$	This is equal to $2q+1$ (it is a safe prime)
$K$	In version 6a of SRP. This is a hash function on $N$ and $g$

Table 3.1 (continued)  
Notation used in proposed protocol

$U$	output of a hash function on A and B at BAN GW
$u'$	output of a hash function on A and B at HAN SM
$J$	bilinear map at BAN GW
$J'$	bilinear map at HAN SM
$W$	output of a hash function on A, B, TS and J at BAN GW
$W'$	output of a hash function on A, B, TS and J' at HAN SM
$seq$	sequence number
$key$	session key calculated at HAN SM
$key'$	session key calculated at BAN GW
$Sign_{key}$	stores signed session key at HAN SM
$Sign_{key}'$	stores signed session key at BAN GW
$SIGN_{key} [message]$	Signcryption on message using key

Table 3.2  
Hash Functions used in the protocol

$H_1(\cdot)$	$(0, 1)^* \times (0, 1)^* \rightarrow (0, 1)^*$
$H_2(\cdot)$	$(0, 1)^* \rightarrow G^*$
$H_3(\cdot)$	$Z_N^* \times G^* \rightarrow G^*$
$H_4(\cdot)$	$G^* \times G^* \times Z_q^* \times G_T^* \rightarrow G^*$
$H_5(\cdot)$	$Z_q^* \times Z_q^* \times G_T^* \rightarrow G^*$

### 3.1 Setup Phase

Let  $G$  be an additive cyclic group of prime order  $q$ , and  $G_T$  be a multiplicative cyclic group of prime order  $q$ ; let  $g$  be the generator of these cyclic groups. In order to build cryptographic systems, pairing-based cryptography utilizes a symmetric bilinear pairing between two elements of an additive group to an element of a multiplicative group [10]. In the proposed protocol, we have used the map  $e: G_1 \times G_2 \rightarrow G_T$ , where  $G_1 = G_2 = G$ . This mapping satisfies the properties stated below:

1. Bilinearity:  $\forall x, y \in Z_q^*, \forall A, B \in G: e(A^x, B^y) = e(A, B)^{xy} \in G_T$
2. Non-degeneracy:  $e(A, B) \neq 1$
3. Computability: there exists an efficient algorithm to compute  $e$

Another measure of non-degeneracy is that for all  $B \in G$ , if  $e(A, B) = 1$ , then  $A = \infty$ . We refer to [1] for a more detailed description on bilinear pairing and its applications. Let there be a bilinear parameter generator which runs an algorithm that takes in as input a security parameter  $L$ , and outputs the system's 5-tuple  $(q, g, G, G_T, e)$ .

1. In the proposed protocol, the BAN GW also acts as the PKG. Hence, as is the function of the PKG, the BAN GW determines the 5-tuple  $(q, g, G, G_T, e)$  by providing input  $L$  to the bilinear parameter generator.
2. The BAN GW randomly chooses a master secret key  $s \in \mathbb{Z}_q^*$ . It does not convey the master secret key to any other entity.
3. The cryptographic secure hash functions are determined by the BAN GW. This protocol utilizes 5 hash functions:

$$H_1(\cdot) : (0, 1)^* \times (0, 1)^* \rightarrow (0, 1)^*$$

$$H_2(\cdot) : (0, 1)^* \rightarrow G^*$$

$$H_3(\cdot) : \mathbb{Z}_N^* \times G^* \rightarrow G^*$$

$$H_4(\cdot) : G_T^* \rightarrow \mathbb{Z}_q^*$$

$$H_5(\cdot) : \mathbb{Z}_q^* \times \mathbb{Z}_q^* \times G_T^* \rightarrow G^*$$

4. The public parameters are  $(q, g, G, G_T, e, H_1, H_2, H_3, H_4)$ .

### 3.2 Pre-authentication Protocol

The following protocol registers the HAN smart meter with the BAN gateway. The HAN smart meter has a pre-allocated password and a unique identifier. Both the password and identifier is assigned by the smart meter manufacturer. The HAN smart meter and the BAN gateway cooperate with each other to exchange the following information via a secure channel [7]. The HAN smart meter registration process at the BAN gateway occurs as follows:

**HAN SM**

$$x = H_1(\text{pwd}, \text{ID}_{\text{HAN}})$$

$$v = g^x$$

**BAN GW**

Message A - AuthReq:  $\text{ID}_{\text{HAN}}, v$

$$\text{PK}_{\text{HAN}} = H_2(\text{ID}_{\text{HAN}})$$

$$\text{SK}_{\text{HAN}} = s \cdot \text{PK}_{\text{HAN}}$$

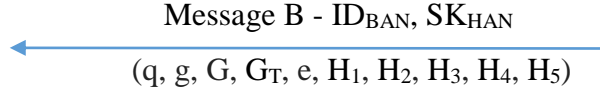


Figure 3.1: Pre-Authentication Phase Exchange

### 3.3 Description of Pre-authentication Protocol

Messages A and B are exchanged prior to commence of the mutual authentication protocol between the smart meters. Owing to its steady rise in popularity, WMN is considered as the communication protocol running between the HAN and Building Area Network smart meters. One of the many advantages of the WMN is its property of self-discovering and self-configuring. Whenever a new wireless node is introduced into an existing WMN, the routing protocol running within the WMN immediately detects it and shares the data transmission paths with it. Thus, with reference to the smart grid environment, when a new HAN SM joins an already established WMN, it is able to successfully construct its routing table with updates from its neighboring wireless nodes. The network shared among a Building Area Network GW and its HAN SMs can be designated as a wireless mesh subnetwork. Hence, a new HAN SM is configured into a WMN but it has still not achieved mutual authentication with the Building Area Network GW. Until it mutually authenticates the Building Area Network GW, all its messages containing meter readings will be dropped.

#### a) Message A

With respect to the Smart Grid environment discussed in this thesis, each smart meter bears a unique identifier. Also, each HAN SM contains a password, which is set by the manufacturer of that smart meter. Every password has its corresponding verifier, which is calculated by applying



an exponentiation operation. This password and verifier are required to execute the Zero Knowledge Password Proof. The routing protocol running in the WMN conveys the verifier to the closest BAN GW. The intermediate smart meters act as transit nodes and not as sink nodes. Message A conveys the identifier and the verifier of a HAN SM to the BAN GW through a secure channel [7]. On receiving such a message, the BAN GW understands that the HAN SM wants to participate in the mutual authentication protocol. So, the BAN GW stores the received identifier and verifier in its memory. A BAN GW has ten times more memory than a HAN [3]. After receiving Message A, the BAN GW functions as the PKG. In other words, the BAN GW uses hash function  $H_1$  to generate the public key for the HAN SM. Next, it applies its master secret key on this newly-generated public key to create the HAN SM's private key. In this manner, the HAN SM initiates the authentication process between itself and the BAN GW.

#### **b) Message B**

At this point, either Message B has successfully reached the HAN SM, or else it has failed to reach the HAN SM. The latter case would have occurred either because the message got lost on its way or because the BAN GW never received Message A in the first place. It is the responsibility of the HAN SM to resend Message A to the BAN GW. The BAN GW only issues Message B as a form of acknowledgement to Message A. If Message B was lost on its way to HAN SM, then it means that the BAN GW will have the identifier and verifier details of the HAN SM in its memory. In such a situation, resending Message A will simply overwrite the existing entry for that identifier. An identifier cannot have two entries because it is unique across the entire Smart Grid environment. If Message A never had reached the BAN GW in the first place, then the normal process ensues. Either way, Message A is sent from the BAN GW to the HAN SM. The end of the pre-authentication phase is marked by the successful receipt of Message B by the HAN SM.

Message B serves as an acknowledgement for the HAN SM having sent its authentication request to the designated BAN GW. The private key as well as the public system parameters is conveyed via Message B from the BAN GW to the HAN SM. Once the HAN SM receives Message B, it stores its private key and the public parameters. It then uses hash function  $H_1$  from the list of public parameters to create the public key for the BAN GW using the identifier sent by the BAN GW. The next step for the HAN SM is to randomly choose a number  $\alpha \in Z_q^*$ . The number  $\alpha$  is used in an exponentiation operation to generate the value of variable  $A$ .

### 3.4 Mutual Authentication Protocol

The following protocol comprises of the steps that authenticate the HAN smart meter and the BAN gateway to each other, and also generates a symmetric session key. The mutual authentication protocol is described below:

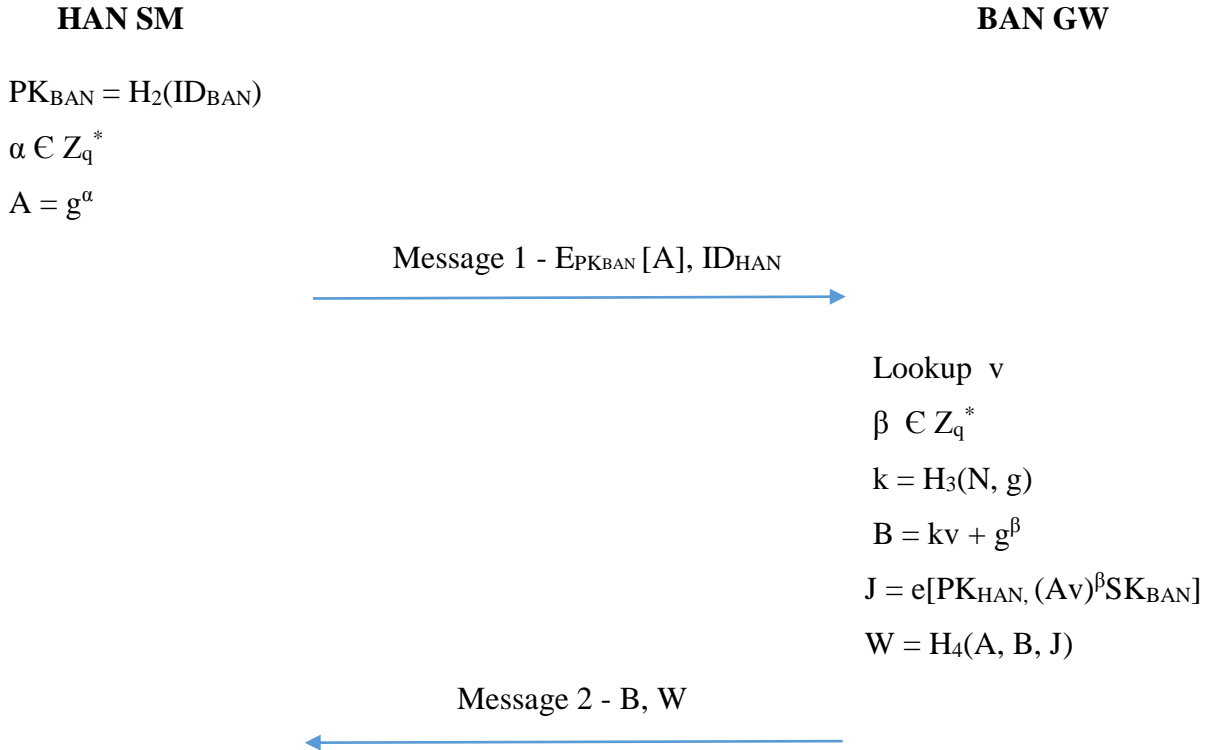



Figure 3.2: Mutual Authentication Protocol Exchange

$J' = e[SK_{HAN}, (B - kv)^{\alpha+x} PK_{BAN}]$   
 $k = H_3(N, g)$   
 $W' = H_4(A, B, J')$   
 If  $W' == W$ , then HAN SM trusts BAN GW  
 Initialize  $seq = 0$   
 $key = H_5(\text{valid-period}, seq, J')$   
 $Sign_{key} = SIGN_{SK_{HAN}, PK_{BAN}} [key]$

Message 3 -  $Sign_{key}, seq, \text{valid-period}$  

Store  $seq, \text{valid-period}$   
 $key' = H_5(\text{valid-period}, seq, J)$   
 $Sign_{key}' = SIGN_{SK_{BAN}, PK_{HAN}} [key]$   
 If  $Sign_{key}' == Sign_{key}$ , then BAN GW trusts HAN SM

Figure 3.2: Mutual Authentication Protocol Exchange (continued)

### 3.5 Description of Authentication Protocol

Messages 1, 2 and 3 constitute the main authentication process between the smart meters. In other words, a total of 3 messages are exchanged between two smart meters to ensure mutual authentication between the two. It is of utmost importance that the HAN SM has its private key, and the BAN GW has the verifier for that HAN SM before the onset of this phase. If that is not the case, then the authentication phase will fail. Firstly, the HAN SM would not be able to sign the session key after having received Message 2. Hence, Message 3 would not be generated at all. Secondly, if the BAN GW does not have the HAN SM's verifier beforehand, then after receiving Message 1 it will be unable to constitute the value of variable B. Either way, the authentication process cannot proceed any further. Messages 1, 2 and 3 are described in further details below.

### **a) Message 1**

As discussed previously, the HAN SM generates the public key of the BAN as well as variable  $A$  using the random number  $\alpha$ . These values are utilized by the HAN SM while composing Message 1. The HAN SM encrypts  $A$  using the public key of the BAN GW. It then sends the encrypted value of  $A$  along with its identifier to the BAN GW. Hence, Message 1 contains the HAN SM's identifier and the encrypted value of  $A$ . If Message 1 does not reach the designated BAN GW, then the receiving smart meter is not able to do anything with the contents of Message 1. The only piece of information it obtains is the identifier of the HAN SM, which can be shared in public freely. The receiver cannot decrypt the value of  $A$  as it does not have the private key of the designated BAN GW. Also, a passive adversary will not be able to gain access to the contents of the encrypted data without having the BAN GW's private key. Hence, the value of  $A$  will be protected in case Message 1 reaches an incorrect destination. It is important to guard value  $A$  because it is directly used to constitute the variable  $J$  that authenticates the BAN GW to the HAN SM. In case Message 1 does not reach the designated BAN GW successfully, then there is no immediate way for the HAN SM to detect this situation. The HAN SM will wait for the arrival of Message 2. Message 2 serves as an acknowledgement for Message 1.

Upon successful arrival of Message 1 at the correct BAN GW, the variable  $A$  is first decrypted using the BAN GW's private key. Also, the identifier sent by the HAN SM is used to lookup the verifier corresponding to that identifier. This is where the pre-authentication phase proves important. The BAN GW will not be successful if it has the wrong verifier corresponding to a HAN SM identifier. This is because the verifier is derived from the password associated with that specific HAN SM. It is to be noted that more than one HAN SM can share the same password, but an identifier has to be unique throughout the Smart Grid environment. It is because of this

reason that a verifier is always stored alongside an identifier. The combination of these two makes a unique combination across the whole Smart Grid environment. The password does not leave the HAN SM, but its verifier is conveyed to the BAN GW in the pre-authentication phase. As mentioned earlier, the password and verifier help enforce the Zero Knowledge Password Proof. ZKPP is a method that will ensure to the BAN GW that the HAN SM has the correct password to the verifier, without the BAN GW knowing the password itself. Actually, the HAN SM wishes to prove its identity using a combination of its password and identifier. As the identifier is already a publicly known piece of information, the HAN SM does not want to divulge its password. Hence, the verifier has a seminal role in this case because it helps prove that the BAN GW is interacting with the HAN SM having the corresponding password to that verifier. Once the authentication protocol begins, this verifier is used in the creation of variable J. If there is any discrepancy in the values of the verifier at the HAN and at the BAN, then that will lead to the protocol becoming unsuccessful owing to failure of ZKPP.

After storing the value of A and locating the verifier corresponding to the identifier of the HAN SM, the BAN GW then chooses a random number  $\beta \in \mathbb{Z}_q^*$ , which is used to compose B which is utilized in calculating variable J. Variable B also requires k. Referring to version 6a of SRP, k is derived by applying hash function  $H_3$  on N and g. N is a safe prime which is equal to  $2q+1$ , where q is the prime order of the cyclic groups used in pairing-based cryptography; g is the generator of these cyclic groups. The variable k is calculated by both sides HAN as well as BAN. Furthermore, when an active adversary impersonates a smart meter, then variable k helps to eliminate 2-for-1 guess.

The next step for the BAN GW is to use these values and create variable B, and then variable J. Variable J plays a seminal role in the mutual authentication process. It holds the bilinear

pairing map using the private key of the HAN SM, variable A received from HAN SM, and  $\beta$  selected by the BAN GW itself. The private key of the HAN SM is used because it is a piece of information that should be known to the HAN SM and BAN GW alone. An adversary that is trying to launch an attack from the outside will not have access to this information because the private key was sent through a secure channel in Message B. Also, an adversary impersonating the BAN GW, which also functions as the PKG, will not be able to generate the correct value of J without the verifier that was sent by the HAN SM to the designated BAN GW.

Variable A was sent to the BAN GW in an encrypted manner. The designated BAN GW alone has the private key to decrypt A. Variable J helps enforce one half of the pairing based cryptography because it stores the bilinear mapping at the BAN GW. After J is created, W is also devised by applying hash function on J. Finally, BAN GW constitutes Message 2 of B and W.

## **b) Message 2**

On receiving Message 2 from the BAN GW, the HAN SM goes on to store the received B. As mentioned above, k is calculated once again. The variable J' is constituted of B and other parameters. J' forms the other half of the bilinear pairing because it comprises of the bilinear mapping held at the HAN SM. Based on the properties of bilinear pairing, variables J and J' have to be equal in order for the HAN SM to be able to authenticate the BAN GW. As mentioned above, A is derived from  $\alpha$  at the HAN SM, and B is derived from  $\beta$  at the BAN GW. The variables A and B are then exchanged between the two smart meters. Hence, if W and W' are not equal to each other, then that reflects inconsistencies in its constituent variables resulting in the abortion of the authentication process.

Once it has been established that W is equal to W' then the HAN SM trusts the BAN GW. To complete the mutual authentication protocol, the BAN GW should also trust the HAN SM. To

do so, the HAN SM introduces a valid-period and a sequence number initialized to 0. It then forms the first session key by applying a hash function on the valid-period,  $J'$  and the sequence number. Furthermore, this session key is signed with the private key of the HAN SM and the BAN GW's public key. Message 3 will, therefore, contain one signed.

### **c) Message 3**

As mentioned previously, Message 3 contains a signcryption. Earlier, to ensure confidentiality, integrity and non-repudiation, the message would first be signed and then encrypted. This process was computationally intensive. Hence, it is not recommended for low power smart meter devices. In 1997, Yulian Zheng introduced the first signcryption process. Also, Zheng proposed a signcryption scheme that, when compared to traditional sign-then-encrypt practices, produced results that consumed 58% less computational and 40% less communication costs [11]. If user X wants to send a message to user Y, then it can produce a cipher text in the following manner. First, user X takes as input its own private key, the identifier of user Y, as well as the original message. Then, it produces the corresponding cipher text. When the receiver, user Y, receives this cipher text, then it takes as input its own private key, user X's identifier, and the cipher text. This returns the original message to user Y.

The BAN GW stores the received valid-period and sequence number. Using these values and the previously calculated  $J$ , the BAN GW creates a session key at its end. It performs signcryption on this key using its private key with HAN SM's public key. If both signcryptions are the same, then the mutual authentication process between these two smart meters are now complete. Hence, explicit key confirmation has been achieved.

## 3.6 Key Renewal Protocol

The proposed mutual authentication protocol has a key renewal process that also offers forward secrecy. Forward secrecy ensures that an adversary cannot gain access to older session keys despite having the current session key. In other words, if an adversary knows the current session key, then the security of the data using that session key is threatened. Owing to forward secrecy, the adversary will not be able to derive any of the previous keys from the current session key. Hence, the data exchanged prior to the usage of the current session key is completely protected.

### 3.6.1 Key Renewal Mechanism

Table 3.3  
Notation for Key Renewal Protocol

$I$	session identifier
$KEY_i[msg(j)]$	$j^{th}$ message is encrypted with of $i^{th}$ session key for session
$H[ ]$	Hash Function
$Seq$	sequence number
$valid-period_i$	validity of $i^{th}$ session key
$PK_{BAN}$	public key of BAN
$E_{PK}[valid-period ]$	encryption of valid-period using public key



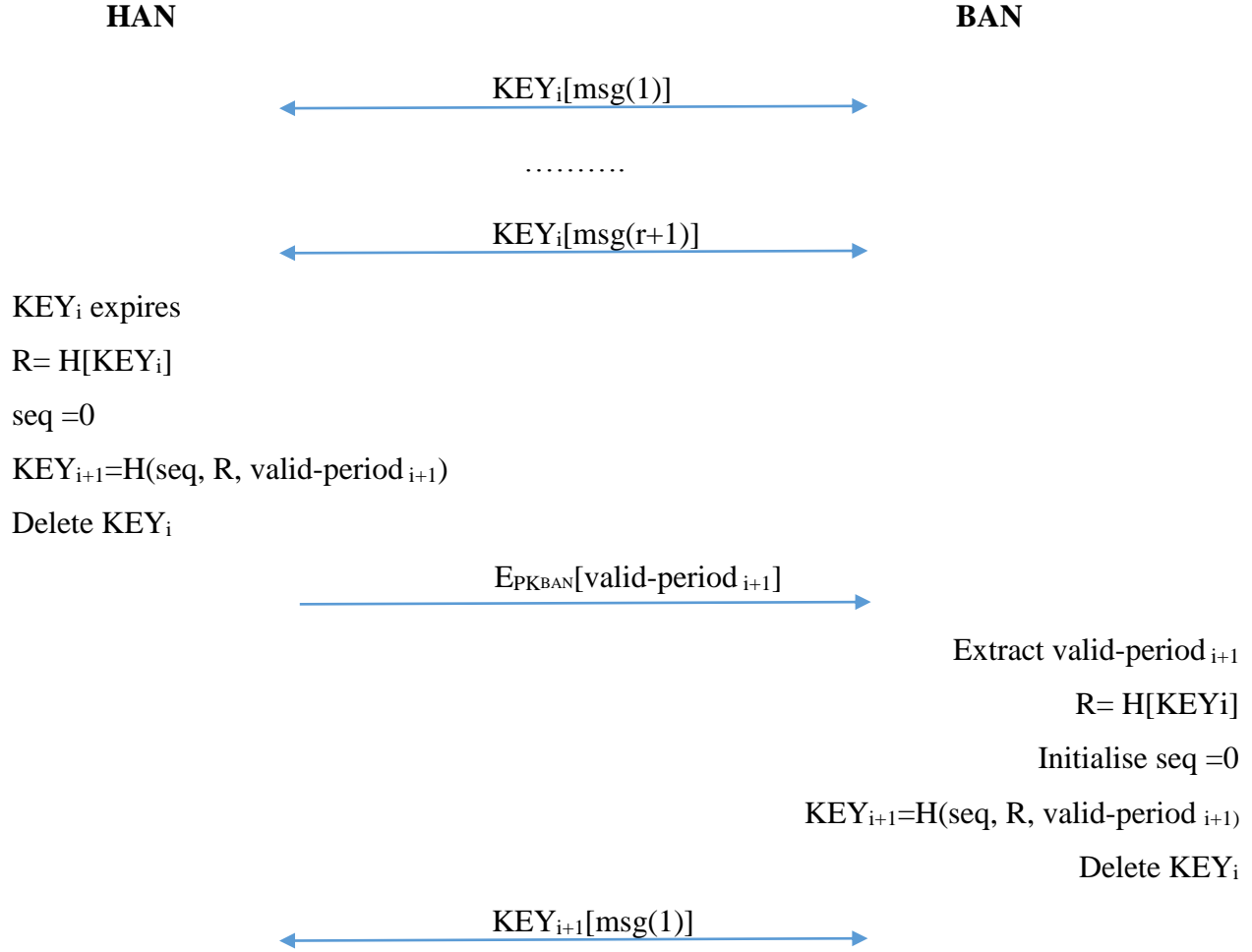


Figure 3.3: Key Renewal Mechanism

### 3.6.2 Description of Key Renewal Mechanism

The first session key is devised at both ends. The BAN GW authenticates the HAN SM only if the signed session key sent by the HAN SM is equal to the signed session key calculated by the BAN GW itself. The variable valid-period signifies the duration of key lifetime. In other words, it is the period of clock time during which the cryptographic key is valid. Every time a message encryption is to be done, the valid-period of the current session key is checked to make sure it has not expired. If it has expired, then a series of operations and actions are performed to generate the new session key.

In the proposed key-renewal process, forward secrecy is provided by two means. Firstly, whenever an active session key expires its validity period, it is immediately deleted to remove all its traces. Secondly, a current session key generation relies partially on its predecessor session key. At the time of expiration of the current session key, the smart meter will calculate its hash and use this hashed value to construct the new session key. It will then proceed to replace the expired session key with the current key session, and also delete the expired session key. In case of an adversary attack, the adversary is unable to retrieve the previous session keys because the hash function is a one-way function. All the previous messages containing meter readings are protected from the adversary.

A BAN GW acts as a PKG as well as a collection node for meter readings sent by HAN smart meters under it. Control commands are sent from the BAN GW to the HAN SM; meter readings are sent vice versa. The frequency of meter readings being reported to the BAN GW ranges from once every 5 minutes to once every hour. Control commands, on the other hand, are issued as per the demands of the situation. Unlike generation of meter usage reports, control commands are not generated on a regular basis. Hence, majority of the encryptions are done by the HAN SM which means that most of the time, it is the HAN SM which has to check for the validity of the current session key. So, the responsibility of choosing duration of clock time as its validity period belongs to the HAN SM. Every time a session key has exceeded its lifetime, the HAN SM will immediately use its local clock to set a new validity period, and reinitializes the sequence number to 0. It then encrypts this new validity period with the public key of the BAN GW. Meanwhile, the HAN SM also creates a hash of the expired session key. It then uses this hash along with the new validity period and the current sequence number to generate the new session key. The HAN SM then sends the encrypted validity period to the BAN GW. The BAN GW

retrieves the new validity period, reinitializes its sequence number to 0 and constitutes the new session key utilizing the validity period, sequence number as well as the hash of the previous session key. It then proceeds to delete the previous session key from its memory. It is important that the HAN SM initiates key renewal iteration by selecting the validity period and sending it across to the BAN GW. This helps in reducing the computation and time required at the BAN GW to monitor the validity period of each session key it shares with its associated HAN SMs, and renew it. Moreover, the HAN SM is more regular at verifying the lifetimes of the keys as compared to the BAN GW because meter readings are regularly reported from the HAN SM end. Therefore, the BAN GW enters into the key renewal process only upon receiving the new validity period from the HAN SM.

Sequence numbers are used in the key renewal process because they provide protection against message replay attacks. Messages with sequence numbers follow a simple policy. If the sequence number has not been used previously and is greater than the sequence number of the previous message, then it is acceptable. With every message exchange, the sequence number is incremented by 1. In the proposed protocol, at the time of generation of a new session key, the sequence number counter is initialized to 0 at both the HAN and the BAN. From then onwards, every message sent and every message received is marked by incrementing the sequence number by 1. For instance, the sequence number at the HAN SM and its corresponding BAN GW reads 7 at the present moment. The HAN SM sends out a meter reading and hence, its sequence number is now equal to 8. At the BAN SM, the sequence number is increased from 7 to 8 the moment that the meter reading from the HAN SM is received.

### 3.7 Security Analysis

Below we present a detailed description of how this protocol is resistant to several attacks. The assumption is made that an active or a passive attack can be made. Hence, a passive attacker may observe all exchanges between two smart meters. An active attacker, on the other hand, may make changes to the actual messages. It may also enter into a smart meter and take control of its operation.

#### a) Replay Attack

A replay attack occurs when an adversary intercepts the message exchange between the sender and the recipient. The adversary captures a message and then retransmits it at a later time. For instance, user A wants access to data guarded by user B. User B requests user A for its credentials to be able to authenticate its access. User A transmits its credentials to user B. Not known to them is that user C is eavesdropping on their message exchange. User C captures the message that user A meant to send to user B in order to be granted access. User C then retransmits this message to user B at a later time. User B authenticates the credentials and allows User C to gain access.

There are three methods of resisting replay attacks. Firstly, the inclusion of a session token. Referring to the scenario described in the previous paragraph, user B can send a session token along with the request for credentials to user A. User A will incorporate the session token either as padding or in a hash function with its authentication credentials. This message, upon reaching user B, will be verified by user B who will apply the same session token on its local copy of user A's authentication credentials. Secondly, the sender and recipient may include timestamps in their message exchanges. The time taken for a message to transmit between the two hosts is calculated. Also, a margin of a few milliseconds is allowed. If a message arrives after the expected duration, it is dropped. In our protocol, this attack may occur. Lastly, one-time passwords also help

authenticate single transactions. They expire either after a single use or are valid for a duration of time.

This attack is unsuccessful in the proposed protocol. The main reason for that is every message conveys a piece of information that is used to generate a value. This value also depends on information obtained previously. That means even if the adversary retransmits the message to the BAN and receives a reply, yet it would be unable to do much with the data received. This is owing to two reasons. Firstly, the sensitive data is always signed or else encrypted. Unless the adversary has the necessary keys, it cannot use this data effectively. Secondly, the plaintext information usually conveys time durations, sequence numbers or random numbers. These pieces of information are used in combination with a variety of previously obtained information. Thus, knowledge of these values alone is insufficient to predict a session key or decrypt the contents of a message. Maintaining timestamps between two users requires the synchronization of their clocks. This is feasible if there are less number of users, but incurs a significant amount of overhead to maintain clock synchronization amongst large numbers of users. Moreover, clock drift occurring at the BAN smart meter has the potential to paralyze all the HAN smart meters reporting to it.

With reference to the protocol, the following scenarios describe how the protocol is resistant against the replay attack. For instance, let us assume that an adversary is eavesdropping between a HAN and BAN smart meter. As the adversary does not know the private key of the BAN smart meter, it cannot decrypt the value of A. Message 3 is replayed by the adversary, and it also receives a reply from the BAN smart meter. The reply contains values B and W in plaintext. W is value generated by combining various pieces of information and their hashes. Owing to the one-directional nature of hash functions, it is extremely difficult and time-consuming to feed the hash function with random inputs until the known output is achieved. Also, the adversary has no

previous knowledge about the password verifier and random value A. The adversary is only able to prohibit HAN smart meter's message from reaching the BAN smart meter. It cannot gain access to sensitive information or secret keys. On the other hand, when the HAN smart meter receives no reply from the BAN smart meter, it simply retransmits its message again.

The next instance is when Message 4 is replayed by an adversary. The adversary does not have much use for B and W alone. Upon receiving a reply from the HAN smart meter, the adversary will once again be able to read the sequence number and valid period. To verify the signature, the adversary requires the public key of the HAN smart meter, the message and its signature. Despite the public key of a HAN smart meter not being exchanged in any transaction in this protocol, let us assume that the adversary has both the signature and the public key of the HAN smart meter. Yet, it requires the message, which is the session key. Attaining the session key is extremely difficult because it has been produced cumulatively over the previous transactions.

## **b) Man-in-the-Middle Attack**

In the MITM attack, the attacker places itself between its two targets. It will establish individual connections with each target. The adversary will do active eavesdropping, where it relays and manipulates the information between the two targets. The two targets believe they are communicating with each other, but in reality the traffic exchange will be controlled by the adversary alone. One of the most common methods of avoiding MITM attack is to ensure mutual authentication between two entities prior to any message exchange. For example, TLS ensures mutual authentication between two entities by trusting one certificate authority.

Let us assume that user A sends a message to user B asking for his encryption key. User C is an active eavesdropper between users A and B. User C intercepts the message and sends it to user B. User B, upon receiving this request, genuinely believes it is from user A and unknowingly

replies to user C with its encryption key. Now, user C once again intercepts this message received from user B, and replaces user B's encryption key with its own key. It then sends this message to user A. Similarly, user A encrypts a message containing sensitive data using what she believes is user B's encryption key, and sends it to user B. As expected, user C is able to decrypt this message using its decryption key. It can then alter the contents of the message, re-encrypt it using user B's encryption key and send it forward to user B. User B will believe this is a message from user A and retrieves the contents of the encrypted message using its decryption key.

MITM attacks can be prevented by adopting public key infrastructures, secure DNS connections, certificate pinning, additional mutual authentication practices such as secret keys, passwords and verifiers. Timestamps also help defend against MITM attacks. For example, if the message between two entities usually take 10 seconds to reach the other end, now takes 40 seconds, then involvement of an adversary is definite.

As mentioned before, a mutual authentication protocol is needed to prevent MITM attack. In our protocol, we use a combination of random values, passwords, hashed messages and keys to create a strong mutual protocol. Furthermore, the HAN smart meter's private key is exchanged through a secure channel. For instance, an adversary intercepts the messages exchanged between the two smart meters. The attacker intends to prevent the messages actually sent by the HAN SM from reaching the BAN SM. So, it receives a message from the HAN SM and replaces the message. The MITM attacker will not be able to decrypt or verify any signed messages and will not know what to replace its contents with. The maximum harm that may occur is that the BAN SM may accept these messages. Even if the BAN SM replies to the message, it is of no use to the attacker because the attacker needs to have knowledge of other entities as well. The advantage of the proposed protocol lies in the fact that if a message contains a sensitive value, then that is encrypted

or signed. Otherwise, the message contains a hash of a value, which can only be verified by being recomputed at the sender's end.

### c) **Known Session Key Attack**

In the known session key attack, the adversary maintains parallel sessions with the two target nodes. The best way to explain this attack would be by means of a scenario. For instance, user C is the adversary between users A and B. User C establishes a parallel session with users A and B. Firstly, it receives a set of credentials from user A, such as user A's certificate details. Let us assume this to be Message 1. Similarly, user C receives a Message 2 from user B containing its certificate details. After this, user C replicates the contents of Message 1 into Message 3, and sends it across to user B. Similarly, it replicates the contents of Message 2 into Message 4 and forwards that to user A. The key generation mechanism involves using both these certificate values. Hence, upon request, user C will be provided with the session keys from both users A and B. This will represent a successful attack by user C as an active adversary.

To eliminate the known session key attack, it is important to remove the symmetry property of the key generation method. This can be achieved by using a key derivation function. In such a scenario, the key derivation function is a hash function  $H$ . For instance, the session key  $K$  is a result of the hash function being applied on a set of variables  $x$  and  $y$ .  $K$  is created by the sender. Similarly, the receiver creates session key  $K'$  using  $x$  and  $y$  as well. It is to be noted that the variables  $x$  and  $y$  should be complex in some nature. For example, they may be a modular exponentiation applied on a few operands. The sender and the receiver have to vary the manner in which the constituent variables are concatenated or associated together. Moreover, if the hash function is a random oracle then there will be no way for the adversary to compute the value of



one session key from the other session key, or vice versa. Hence, the session key should be generated in such a way that the two nodes compute the same key using different sets of variables. A good example of this is bilinear mapping.

The proposed protocol is well-guarded from the known session key attack owing to the incorporation of bilinear mapping techniques. After an exchange of verifier, random values and keys, the sender and receiver use a different combination of variables and apply a hash function on it. These constituent variables are complex and difficult to guess. This acts as a barrier to the adversary. Next, the HAN smart meter uses a combination of its private key and the BAN smart meter's public key, along with random value B. The BAN smart meter uses its private key, the HAN smart meter's public key, and random variables A and  $\beta$ . Both of these values give the same session key but with different set of variables.

Known session key attack is also possible if the previous session keys are at a risk of being derived from the present session key. The proposed protocol is safe from the Known Session Key attack owing to Forward Secrecy. As discussed previously, forward secrecy ensures that an attacker, upon successful occupation of a smart meter, will be unable to gain access to previous session keys and hence, previous messages. In the proposed protocol, this is ensured by complete removal of the previous session keys once a new session key has been generated. Moreover, at the time of expiration of a session key, its hash is used for creating the new session key. Hence, session keys are partially comprised of hash values. It is difficult for the attacker to reverse the hash operation to find out the original session key. Hence, the proposed protocol is successful in protecting the previous session keys and providing forward secrecy.

#### **d) Impersonation Attack**

An impersonation attack is where the adversary assumes the identity of a legitimate node in a system. In other words, an adversary has to be successful in bypassing authentication mechanisms. For example, an attacker may attempt to extort money from people by creating a phony website. They will send phishing emails to the victims and lead them to a website similar to that of the original bank. Once the victim logs in, then his password is easily available to the attacker. Other forms of attack include a dictionary attack applied on passwords, or network sniffing. Keyboard loggers are also widely used to retrieve passwords. In other words, impersonation attack is possible if the adversary is able to gain access to any piece of information that authenticates the identity of that node to the remaining nodes.

In the smart grid environment, impersonation attack occurs when the attacker pretends to be a smart meter. For instance, an attacker may impersonate a HAN SM. To attain the identity of a HAN smart meter, the adversary has to have the identifier, the password, and the corresponding verifier. During the authentication process, the adversary attempting such an attack will always be unsuccessful owing to several factors. Firstly, the adversary might have access to the HAN smart meter's identifier and public key. But owing to ZKPP, the impersonating attacker will not know the password or the corresponding verifier of the actual HAN SM. Since the mutual authentication depends on the password and its related verifier, impersonation attack will definitely not succeed.

#### **e) Key Control Attack**

Key control attack is one where the protocol is designed in such a way that one of the entities has the power to pre-select the shared session key. So, if this entity is attacked by the attacker, then the shared session key and all the following communication is in full control of the attacker. In other words, even though the entities mutually authenticate each other. Yet, one entity alone plays

a role in choosing the session key. In future, if this entity is attacked, then key control lies in its hands alone.

Ways of avoiding a key control attack is by ensuring that both entities actively contribute to the generation of the session key. It should also be kept in mind that the key renewal process depends equally on both entities. For instance, values selected randomly and independently by both entities can be incorporated in the generation as well as the key renewal process for session keys. These values may be random numbers, nonce, password, identifiers or intermediate keys.

In the proposed protocol, the session key and mutual authentication process depends on the verifier and its corresponding password, and also between two random numbers exchanged between the two smart meters. The key is calculated individually at both ends and then verified. Hence, it does not depend on one end alone. If an attacker successfully compromises the security of one entity, even then it does not have absolute authority to be able to manipulate session keys maintained by that entity. Furthermore, bilinear mapping involves using the public and private keys of the entities. The HAN smart meter uses its private key and the BAN smart meter's public key. On the other hand, the BAN smart meter uses its private key and the public key belonging to the HAN smart meter. Owing to the property of bilinearity, both of these values should be the same. During the key renewal phase, upon expiration of the session key the HAN smart meter sends the valid period to the BAN smart meter. An alternative to this would be using a timestamp. But the disadvantage of using timestamps is the need to maintain clock synchronization between two smart meters. As mentioned before, maintaining clock synchronization for the smart grid network connecting the HAN and BAN smart meters may prove to be computation and time intensive, and also risky.

## Chapter 4

### Performance Analysis

The following section compares the performance of the proposed protocol with ECDSA. As the proposed protocol uses AES-128 and SHA-256, its equivalent algorithm ECDSA-256 has been chosen. The reason for choosing ECDSA is that it is a standardized algorithm, already in practice in the smart grid environment. There are two common modes of encryption/authentication being used in the smart grid industry. The first is a combination of AES-128 and SHA-256, with the public/private keys being provided by RSA-2048. The second industry protocol for authentication is ECDSA-256. MATLAB [23] and OpenSSL [22] have been used to generate results which depict that the proposed protocol is a better alternative. OpenSSL provides the processing times for SHA-256, AES-128, ECDSA-256 and RSA-2048 on a 2.2GHz processor. The following table shows the processing times for the various encryption and hash operations using OpenSSL, and utilized in the performance comparison of the proposed protocol.

```

First we calculate the approximate speed ...
Doing sha256 41943040 times on 16 size blocks: 41943040 sha256's in 27.40s
Doing sha256 -25165824 times on 64 size blocks: 0 sha256's in 0.00s
Doing sha256 -6291456 times on 256 size blocks: 0 sha256's in 0.00s
Doing sha256 -1572864 times on 1024 size blocks: 0 sha256's in 0.00s
Doing sha256 -196608 times on 8192 size blocks: 0 sha256's in 0.00s
Doing aes-128 cbc 41943040 times on 16 size blocks: 41943040 aes-128 cbc's in 4.02s
Doing aes-128 cbc 10485760 times on 64 size blocks: 10485760 aes-128 cbc's in 3.73s
Doing aes-128 cbc 2621440 times on 256 size blocks: 2621440 aes-128 cbc's in 3.72s
Doing aes-128 cbc 655360 times on 1024 size blocks: 655360 aes-128 cbc's in 3.84s
Doing aes-128 cbc 81920 times on 8192 size blocks: 81920 aes-128 cbc's in 3.88s
Doing 327 2048 bit private rsa's: 327 2048 bit private RSA's in 1.05s
Doing 6553 2048 bit public rsa's: 6553 2048 bit public RSA's in 0.55s
Doing 5242 256 bit sign ecDSA's: 5242 256 bit ECDSA signs in 1.34s
Doing 2621 256 bit verify ecDSA's: 2621 256 bit ECDSA verify in 2.91s
OpenSSL 0.9.8k 25 Mar 2009
built on: Thu Jul 23 09:35:27 2009
options:bn(64,64) md2(int) rc4(ptr,int) des(idx,cisc,4,long) aes(partial) idea(int) blowfish(idx)
compiler: cl /MD /Ox /W3 /Gs0 /GF /Gy /nologo -DWIN32_LEAN_AND_MEAN -DL_ENDIAN
-DDSO_WIN32 -DOPENSSL_SYSNAME_WIN32 -DOPENSSL_SYSNAME_WINNT -DUNICODE -D_UNICODE
-D_CRT_SECURE_NO_DEPRECATED -D_CRT_NONSTDC_NO_DEPRECATED -DOPENSSL_USE_APPLINK -I
./Fdout32d11 -DOPENSSL_NO_CAMELLIA -DOPENSSL_NO_SEED -DOPENSSL_NO_RC5 -DOPENSSL
_NO_MDC2 -DOPENSSL_NO_CMS -DOPENSSL_NO_JPAKE -DOPENSSL_NO_CAPIENG -DOPENSSL_NO_K
RB5 -DOPENSSL_NO_DYNAMIC_ENGINE
available timing options: TIMEB HZ=1000
timing function used: ftime
The 'numbers' are in 1000s of bytes per second processed.
type           16 bytes      64 bytes      256 bytes      1024 bytes      8192 bytes
aes-128 cbc     166979.01k    179675.67k    180206.40k    174762.67k    173139.48k
sha256          24493.18k         0.00         0.00         0.00         0.00
               sign    verify    sign/s    verify/s
rsa 2048 bits 0.003202s 0.000084s    312.3    11914.5
               sign    verify    sign/s    verify/s
256 bit ecDSA (nistp256) 0.0003s 0.0011s    3900.3    901.0
OpenSSL>

```

Figure 4.1 OpenSSL Algorithm Processing Delay

Table 4.6  
Simulation Parameters

<b>Simulation Parameter</b>	<b>Value</b>
Interval of message generation	Once every hour
Simulation time	24 hours
TCP Header	20 bytes
IPv4 Header	20 bytes
Ethernet Header	26 bytes
Payload	32 bytes
SHA-256 header	32 bytes
ECDSA signature size	64 bytes
ECDSA certificate size	125 bytes
Number of HAN smart meters	Maximum of 250 per BAN gateway

#### 4.2.1 Communication Overhead

Communication overhead refers to all other information that accompanies the payload to ensure it reliable delivery to the destination. The meter readings are the payload. The payload is transmitted from the customer's home to the utility control center. This transmission requires incorporation of additional information with the payload to ensure its reliable delivery to the utility control center. This additional information includes source and destination IP addresses, source and destination MAC addresses, source and destination port numbers, checksum, flags, sequence numbers and acknowledgements as well. In other words, the payload has to depend on various protocols to be transmitted to its destination. With regards to the OSI model, the payload is accompanied by 3 headers – Ethernet header (26 bytes), IPv4 header (20 bytes) and TCP header (20 bytes). The TCP

header will provide the source and destination port numbers, the IPv4 header will furnish the source and destination IP addresses, and lastly, the Ethernet header will contain the source and destination MAC addresses.

The following graph compares the communication overhead experienced by the BAN gateway of the proposed protocol and the industry favored ECDSA-256 algorithm. The HAN smart meter takes meter readings at regular intervals. These intervals may vary between one in every 15 seconds to once every hour. But the frequency of reporting these meter readings is less frequent (average is 6- 10 times in a 24-hour period for a residential smart meter). In this evaluation, to generate more traffic for more accurate analytical results, the message generation interval of the HAN smart meter is considered as once in every hour. Also, the entire simulation time is 24 hours (1 day). Furthermore, every BAN gateway is a gateway for a maximum of 250 smart meters.

In the proposed protocol, communication overhead has been calculated by taking into account the sizes of the TCP, IP, Ethernet and SHA-256 hash headers, which are 20 bytes, 20 bytes, 26 bytes and 32 bytes respectively. Therefore, raw payload, which is 32 bytes, is accompanied by an additional 98 bytes. In the ECDSA algorithm, the communication overhead includes the TCP, IP and Ethernet headers, as well as the ECDSA signature and certificate (64 bytes and 125 bytes respectively). So, the communication overhead generated by the ECDSA algorithm using the same payload as the proposed algorithm is 255 bytes. Assuming that all the smart meters send their meter reports simultaneously, Figure 4.2 describes the communication overhead incurred by both protocols.

When the number of HAN smart meters is 50, the communication overhead experienced by the BAN smart meter is around 100 KB for the proposed protocol, and around 300 KB for the

ECDSA algorithm. As the number of HAN smart meters increase to 125, the disparity between these two methods increase further. ECDSA algorithm has a communication overhead close to 775 KB. On the other hand, the proposed protocol displays a communication overhead of less than 300 KB. In the last scenario, when the number of HAN smart meters is 250, the communication overhead for ECDSA algorithm is almost 1550 KB, whereas the proposed protocol consumes 600 KB. This value is less than half of the communication overhead experienced in the ECDSA algorithm. Hence, with an increasing number of smart meters at the HAN, the communication overhead will increase greatly for the ECDSA algorithm. This may act as a barrier in further expansion of the smart meter network.

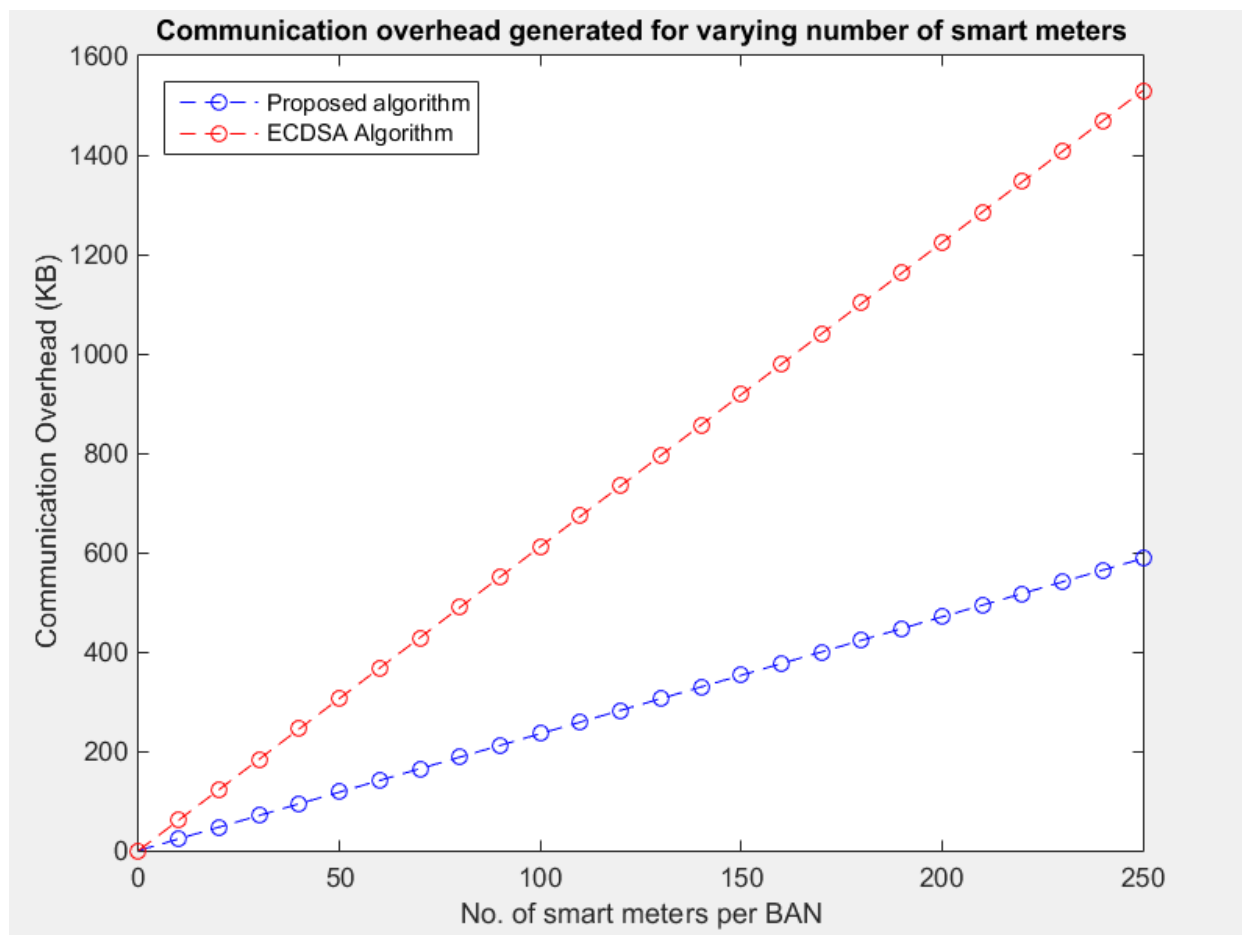


Figure 4.2 Communication Overhead



## 4.2.2 Average Delay

Average delay refers to the mean time taken to perform decryption/signature verification for the cipher text. Figure 4.3 compares the proposed protocol with ECDSA-256 algorithm and RSA-2048 asymmetric protocol. In the proposed protocol, after the session key has been established, the message is first encrypted using AES-128 algorithm using this session key, and then signed using SHA-256 hash function. This ensures source authentication, data integrity as well as message confidentiality.

The processing times were generated using OpenSSL on a 2.2GHz processor. Symmetric algorithm AES-128 processes 41943040 16-bytes blocks in 4.02 seconds. Similarly, SHA-256 processes 41943040 16-bytes blocks in 27.40 seconds. On the other hand, each RSA-2048 signature verification takes 0.000084 seconds. Similarly, a single ECDSA-256 signature takes 0.0003 seconds to verify. These processing times have been obtained using the OpenSSL tool [22]. The average delay is calculated by dividing the total delay by the number of smart meters. The total delay is generated by applying the rate of decryption/verification to the traffic generated by varying number of smart meters over a span of 24 hours. This traffic refers to the size of the meter reports generated by the smart meters. As mentioned earlier, each meter report is a payload of 32 bytes. In the proposed protocol, AES-128 is applied on 16-byte blocks. Hence, for each meter reading generated, AES-128 works on 3 16-byte blocks of plaintext. And it produces a cipher text of size 48 bytes. It is important to mention that in AES, when the plaintext is in exact multiples of 16 bytes, an additional 16 bytes is concatenated with the plaintext for padding purposes. The padding acts as a delimiter and hence, is mandatory. If the plaintext is not an exact multiple of 16 bytes, then the remaining payload is padded until it is an exact multiple of 16 bytes. After the encryption, the cipher text is then inputted to SHA-256. As the name suggests, SHA-256 produces

a hash of 256 bits or 32 bytes. Hence, after AES-128 encryption and SHA-256 hashing, the payload is 80 bytes (48 bytes + 32 bytes). ECDSA signature scheme doesn't encrypt, therefore it can work on arbitrarily sized plaintext. So, the signature verification delay calculation for ECDSA takes into account the number of messages instead of the size of each message. Lastly, RSA-2048 accepts plaintext of up to 256 bits. As the raw payload of meter reading is 256 bits, RSA-2048 is sufficient for this evaluation. Like ECDSA, the encryption delay for RSA depends on the number of meter readings generated by varying number of smart meters over a period of 1 day. The reason why AES-128 encryption delay requires a specific calculation of each payload size is that AES-128 is a block cipher.

In Figure 4.3, it is clearly depicted that RSA-2048 is very expensive in terms of delay. The BAN gateway experiences an average delay of 0.075 seconds for 250 smart meters, each generating meter reports once every hour over 24 hours. Now, if the graph is analyzed closely, then the graph plot for the proposed algorithm lies almost along the x-axis. The reason for the high disparity between the graph plots for the ECDSA algorithm and the proposed protocol, which forces the plotting of the proposed protocol to become almost parallel to the x-axis. To get better understanding of the delay experienced by the proposed protocol, Figure 4.3 has also been provided. Considering the highest number of smart meters in this simulation, which is 250, the proposed protocol experiences an average delay of 0.0085ms for 50 smart meters. Similarly, for 175 and 250 smart meters, the average delay generated is 0.095 ms and about 0.01 ms respectively. On the other hand, the ECDSA algorithm experiences an average delay of 0.05 seconds for 250 smart meters. This comparison shows the wide difference between the ECDSA algorithm and the proposed protocol. Furthermore, RSA displays a very high delay of 0.075 seconds for the same number of smart meters.

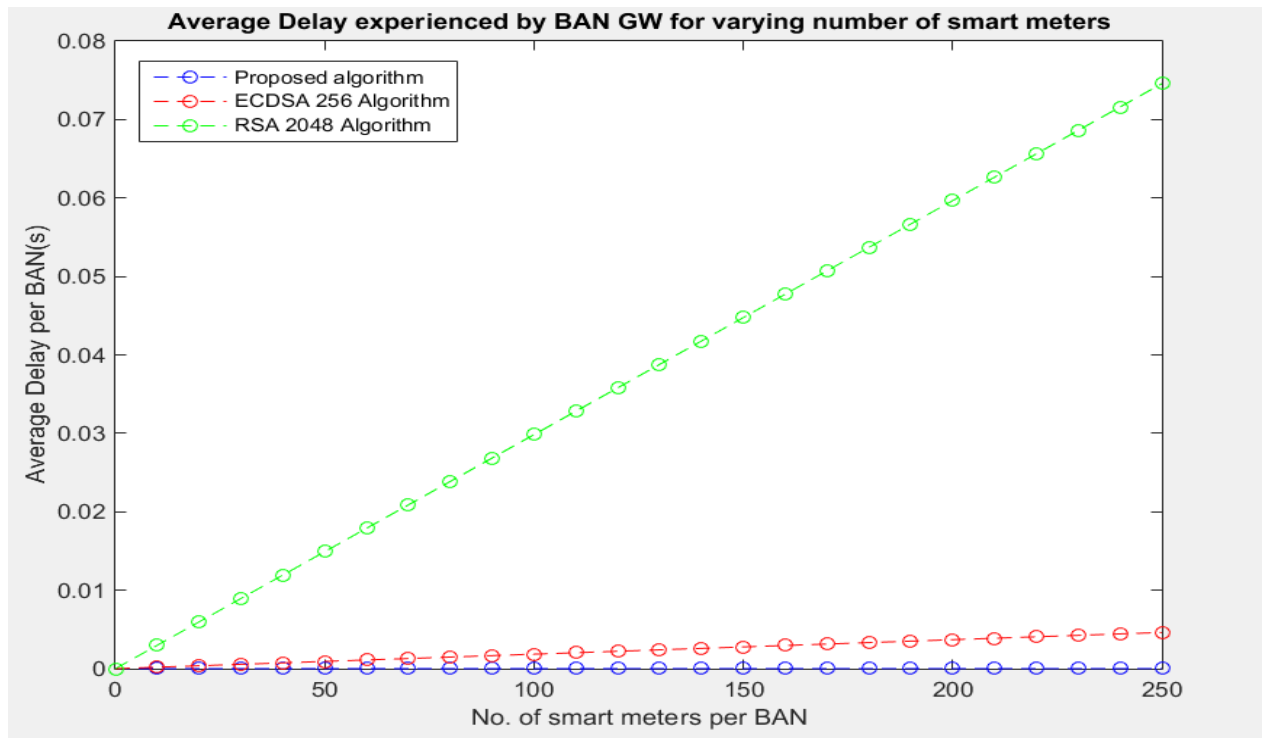


Figure 4.3 Average decryption/verification delay

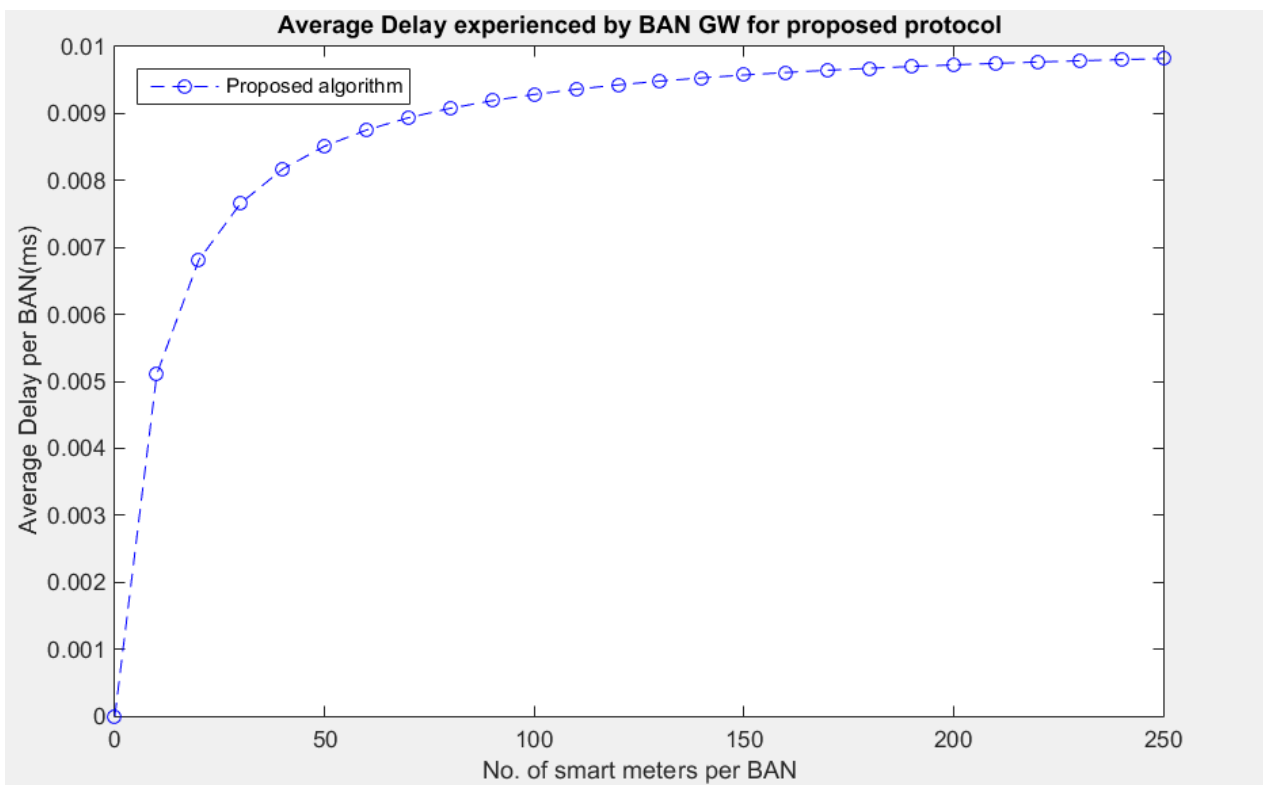


Figure 4.4 Average delay experienced for proposed protocol

### 4.2.3 Buffer Occupancy

The BAN smart meter has a RAM of 128 KB and a flash memory of 1 MB. In total, the BAN has 1152 KB of memory size. The RAM and flash memory represents the buffer size of the BAN gateway. While transmitting meter reports, the BAN receives messages from the HAN smart meters and then conveys them to either the NAN gateway or the utility control center. The rate of incoming messages is faster than the rate of outgoing messages, at the BAN gateway. As a result, the BAN utilizes its buffer space of 1152 KB to store the incoming messages. On average, as we know, the frequency of incoming meter readings at the BAN is once every hour. Being a device of higher capabilities than the HAN smart meter, the BAN gateway (160 MHz processing power [3]) can store the received meter readings and process a bulk of messages before sending them out. The average rate of meter reading transmission is considered to be once every 8 hours. In other words, the rate at which the BAN gateway is receiving messages is once every hour, but the rate at which it processes these messages is once every 8 hours.

To depict buffer occupancy, the x-axis represents a span of time (in hours) from 0 to 8 hours with increments every 30 minutes, and, the y-axis plots the amount of memory consumed in KB. This graph representation will assist in visualizing the buffer occupancy over an 8 hour period. To depict memory consumption, three different rates of meter report delivery are considered. They are as follows: 10 messages every 15 minutes, 50 messages every 15 minutes and 100 messages every 15 minutes. For the proposed protocol, the size of each message is 120 bytes. This includes the 66 bytes of headers, 48 bytes of encrypted data, and 32 bytes of hash. For the ECDSA scheme, the total message size is 287 bytes. Headers attribute for 66 bytes, while ECDSA signature and certificate attribute for 64 bytes and 125 bytes respectively. As is known, the raw payload is 32 bytes. The buffer occupancy is calculated by cumulatively adding the size of total messages every

30 minutes. This representation of memory occupancy helps gauge whether the proposed and ECDSA protocol are able to handle incoming message rates of 10, 50 and 100 messages every 15 minutes without clearing the buffer for 8 hours.

With reference to the graph depicted below, the ECDSA algorithm and the proposed protocol are both successful in managing the BAN smart meter's available memory of 1152 KB when the message generation interval is 10 messages every 15 minutes. The proposed protocol occupies less than 100 KB, whereas ECDSA occupies almost 200 KB. With respect to the message generation rate being 50 messages, the ECDSA algorithm exceeds the memory space of the BAN by consuming approximately 900KB. The proposed protocol, on the other hand, occupies just 400 KB of buffer space. Lastly, with a message generation rate of 100 messages, the ECDSA exceeds the buffer space of the BAN vastly. It consumes more than 1200 KB of memory. But the proposed protocol consumes approximately 775 KB of memory, which is well within the 1152 KB offered by the BAN. This proves that the proposed protocol is able to handle incoming message rates that are greater than 100 messages for every 15 minutes. Hence, this proves the proposed protocol to be more scalable and lightweight, without compromising on security.

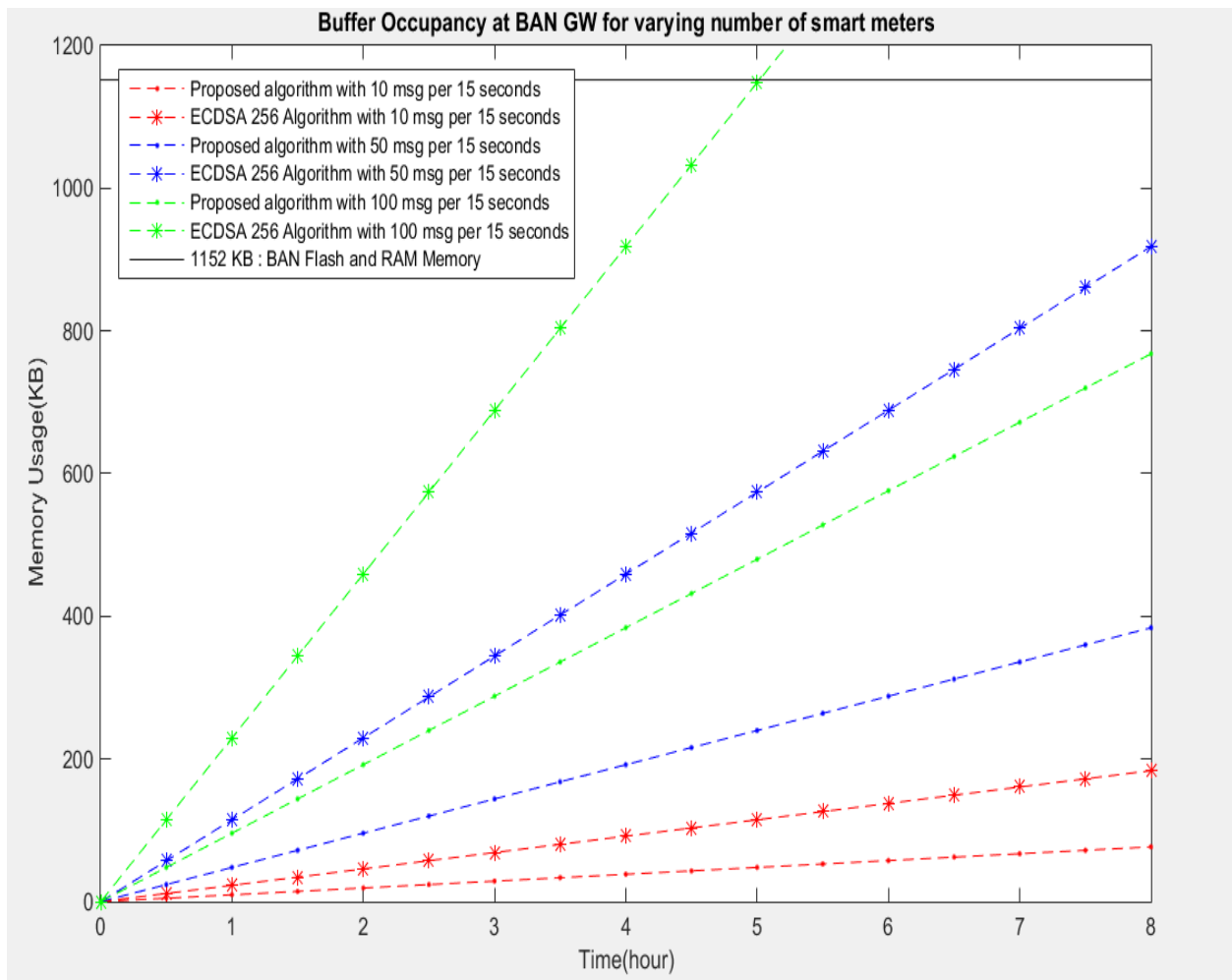


Figure 4.5 Buffer occupancy at BAN gateway

## Chapter 5

### Conclusion

The proposed mutual authentication protocol is both lightweight and scalable. Furthermore, it provides source authentication, data integrity, message confidentiality and non-repudiation. By using signatures with private keys, the sender cannot deny having sent a message. Encryption operations using public keys ensure decryption using only the corresponding private keys. Furthermore, the key renewal scheme is also lightweight. It does not use timestamps because clock synchronization incurs overhead as well. Instead, clock time, with reference to the clock at the smart meter, is used. This eliminates overhead and clock drift issues as well.

The proposed protocol is secure against Replay Attack, Man-in-the-Middle Attack, Known Session Key Attack, Impersonation Attack, and Key Control Attack. Replay attacks are averted because each message in the authentication protocol, is connected to the succeeding and preceding messages. Man-in-the-Middle attacks are prevented by encrypting messages using the public key of the recipient. Known Session Key attacks are averted by removing the symmetry property from the key generation algorithm. This is made possible in the proposed paper by using ID-based Cryptography and Bilinear Pairing. The constituents of each bilinear map at both ends were different, hence key generation method was different. Impersonators would likely impersonate the smart meters to have direct control over the meter readings. Each smart meter has a password associated with it, which never leaves this smart meter. Only its corresponding verifier is conveyed through a secure channel to later confirm its identity with the server. On account of Zero Knowledge Password Proof, the password never leaves the smart meter. Hence, an impersonator can never get the password from the authentication protocol exchange. Lastly, the

Key Control attack is prevented by ensuring that one node is not given the sole responsibility of calculating the session key. In the proposed protocol, both nodes choose a random number and contribute equally to key generation. Also, when checking signatures or value of  $W$  or  $W'$ , the nodes use their local copy of variables.

The comparison between the proposed protocol and ECDSA reveals that the proposed protocol is more lightweight. The proposed protocol incurs a computation overhead of 98 bytes, whereas ECDSA incurs 255 bytes. The proposed protocol has an average delay of 0.01ms. ECDSA displays an average delay of 0.05s, whereas RSA displays an even higher average delay. With an incoming rate of 100 messages every 15 minutes over a simulation period of 8 hours, the proposed protocol uses 775 KB whereas ECDSA suffers buffer overflow.



## Bibliography

- [1] J. Menezes, O. P. C. Van, S. A. Vanstone, *Handbook of applied cryptography*. Boca Raton: CRC Press, 1997.
- [2] "NIST Special Publication 1108R2, NIST Framework and Roadmap for Smart Grid Interoperability Standards, Release 2.0." [http://www.nist.gov/smartgrid/upload/NIST\\_Framework\\_Release\\_2-0\\_corr.pdf](http://www.nist.gov/smartgrid/upload/NIST_Framework_Release_2-0_corr.pdf), 2012. last accessed August 2015.
- [3] M. M. Fouda, Z. M. Fadlullah, N. Kato, L. Rongxing, S. Xuemin, "A Lightweight Message Authentication Scheme for Smart Grid Communications," in *IEEE Transactions on Smart Grid*, vol. 2, no. 4, pp. 675-685, December 2011.
- [4] X. Fang, S. Misra, G. Xue, D. Yang, "Smart Grid — The New and Improved Power Grid: A Survey," in *Communications Surveys & Tutorials, IEEE* , vol. 14, no. 4, pp. 944-980, Fourth Quarter 2012.
- [5] M. Balakrishnan, M. Mienkina, "Designing Smart Meters for the Smart Grid." [http://www.freescale.com/files/training\\_pdf/WBNR\\_SMARTMETER.pdf](http://www.freescale.com/files/training_pdf/WBNR_SMARTMETER.pdf). last accessed August 2015.
- [6] H. Nicanfar, P. Jokar, K. Beznosov, V. C. M. Leung, "Efficient authentication and key management mechanisms for smart grid communications," in *IEEE Systems Journal*, vol. 8, no. 2, pp. 629-640, July 2014.
- [7] C. Chou, K. Tsai, C. Lu, "Two ID-based authenticated schemes with key agreement for mobile environments," in *The Journal of Supercomputing*, vol. 66, no. 2, pp. 973-988, November 2013.

- [8] H. Li , X. Lin , H. Yang , X. Liang ,R. Lu ,X. Shen, “EPPDR: An efficient privacy-preserving demand response scheme with adaptive key evolution in smart grid” in *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 8, pp. 2053-2064, April 2013.
- [9] Z. Fan, P. Kulkarni, S. Gormus, C. Efthymiou, G. Kalogridis, M. Sooriyabandara, Z. Zhu; S. Lambotharan, W. H. Chin, "Smart Grid Communications: Overview of Research Challenges, Solutions, and Standardization Activities," in *Communications Surveys & Tutorials, IEEE* , vol. 15, no. 1, pp. 21,38, First Quarter 2013.
- [10] N. Liu , J. Chen , L. Zhu , J. Zhang, Y. He, “A key management scheme for secure communications of advanced metering infrastructure in smart grid,” in *IEEE Transactions on Industrial Electronics*, vol. 60, no. 10, pp. 4746-4756, August 2013.
- [11] “Zigbee-based Home Area Networks Enable Smarter Energy Management.” <https://www.silabs.com/Support%20Documents/TechnicalDocs/ZigBee-based-HANs-for-Energy-Management.pdf>, 2013. last accessed August 2015.
- [12] C. March and C. Youngblood, “An Introduction to Identity-based Cryptography.” [https://courses.cs.washington.edu/courses/csep590/06wi/finalprojects/youngblood\\_csep590tu\\_final\\_paper.pdf](https://courses.cs.washington.edu/courses/csep590/06wi/finalprojects/youngblood_csep590tu_final_paper.pdf). last accessed August 2015.
- [13] Y. Yacobi, “A note on the bilinear Diffie-hellman assumption”, in *Cryptology ePrint Archive*, Report 2002/113, August 2002.
- [14] “Energy Efficiency in the Power Grid.” <https://www.nema.org/Products/Documents/TDEnergyEff.pdf>, 2007. last accessed August 2015.
- [15] R. Herold, C. Hertzog, “Data Privacy for the Smart Grid,” Auerbach Publications, January 2015.

- [16] J. Benoit, "An Introduction to Cryptography as Applied to the Smart Grid." [http://www.cooperindustries.com/content/dam/public/powersystems/products/grid\\_automation/resources/Cryptography\\_and\\_the\\_Smart\\_Grid.pdf](http://www.cooperindustries.com/content/dam/public/powersystems/products/grid_automation/resources/Cryptography_and_the_Smart_Grid.pdf). last accessed August 2015.
- [17] H. Li, "Enabling Secure and Privacy Preserving Communications in Smart Grids." Retrieved from [http://reader.ebilib.com.ezproxy.lib.ryerson.ca/\(S\(51ao0xeuylxekuko5hzvda5ql\)\)/Reader.aspx?p=1697934&o=2160&u=Bc2k5l%2f3UwE%3d&t=1440006304&h=683DB7C24A334DE0215C712FA358E8ADD728077B&s=37685006&ut=7370&pg=13&r=img&c=-1&pat=n&cms=-1&sd=2](http://reader.ebilib.com.ezproxy.lib.ryerson.ca/(S(51ao0xeuylxekuko5hzvda5ql))/Reader.aspx?p=1697934&o=2160&u=Bc2k5l%2f3UwE%3d&t=1440006304&h=683DB7C24A334DE0215C712FA358E8ADD728077B&s=37685006&ut=7370&pg=13&r=img&c=-1&pat=n&cms=-1&sd=2). last accessed August 2015.
- [18] M. Balakrishnan, "Security in Smart Meters." [http://cache.freescale.com/files/industrial/doc/white\\_paper/SECSMTMTRWP.pdf](http://cache.freescale.com/files/industrial/doc/white_paper/SECSMTMTRWP.pdf), August 2012. Last accessed August 2015.
- [19] C. Bekara, T. Luckenbach, K. Bekara, "A Privacy Preserving and Secure Authentication Protocol for the Advanced Metering Infrastructure with Non-Repudiation Service," in *ENERGY 2012 : The Second International Conference on Smart Grids, Green Communications and IT Energy-aware Technologies*, March 2012.
- [20] H. K.-H. So, S. H. Kwok, E. Y. Lam, and K.-S. Lui, "Zero- configuration identity-based signcryption scheme for smart grid," in *Proceedings of First IEEE International Conference on Smart Grid Communications*, pp. 321-326, Oct. 2010.
- [21] T. W. Chim, S. M. Yiu, L. C. K. Hui, V. O. K. Li, "PASS: Privacy-preserving authentication scheme for smart grid network," *Smart Grid Communications (SmartGridComm), 2011 IEEE International Conference on*, vol. 196, no. 201, pp. 17-20, October 2011.
- [22] "OpenSSL." <https://www.openssl.org/>. last accessed August 2015.
- [23] "MATLAB and Statistics Toolbox Release 2014b." last accessed August 2015.