A QUANTUM-BASED SIGNCRYPTION FOR SUPERVISORY CONTROL AND DATA ACQUISITION (SCADA) NETWORKS

by

Sagarika Ghosh

Submitted in partial fulfillment of the requirements
for the degree of Master of Computer Science

at

Dalhousie University
Halifax, Nova Scotia
August 2019

This thesis is dedicated to two of my first idols, my father, Mr. Samiran Ghosh and, my mother, Mrs. Lina Ghosh. And my beloved little sister, Miss.Seemarekha Ghosh.

Contents

List of Tables

List of Figures

Abstract

Supervisory Control and Data Acquisition (SCADA) systems are used for monitoring industrial processes such as power grids, water supply systems, traffic control, oil and natural gas mining, space stations and nuclear plants. However, their security faces the threat of being compromised due to the increasing use of open access networks. Furthermore, the emergence of quantum computing has exposed a new type of threat to SCADA systems. Failure to secure SCADA systems can lead to catastrophic consequences. For example, a malicious attack can take control of the power supply to a city, shut down the water supply system, or cause malfunction of a nuclear reactor.

The primary goal of this thesis is to classify attacks on SCADA systems, identify the new type of attack based on quantum computing, and design a novel security scheme to defend against traditional attacks as well as the quantum attack. The proposed 'Signcryption' scheme provides both encryption and intrusion detection. In particular, it detects the man-in-the-middle attack as this intrusion can lead to others. The signcryption scheme is built on the foundation of the fundamental BB84 cryptographic scheme and does not involve computationally expensive third party validation. We simulate the proposed scheme using the Quantum Information Toolkit in Python. Furthermore, we validate and analyze the proposed scheme using security verification tools, namely, Scyther and Prism.

Glossary

API	American Petroleum Institute
ASKMA	Adavanced Key-Management Architecture
C-S	Controller -Subordinate
CAPK	Cryptographic Authority Public Key
CK	Common Key
DNP3	Distributed Network Protocol 3.0
DoS	Denial of Service
ECC	Elliptic Curve Cryptography
ECP	Error Correction Protocol
GK	General Key
GSK	General Seed Key
HKMA	Hybrid Key Management Architecture
HMI	Human-Machine Interface
IDS	Intrusion Detection System
IEC	International Electrotechnical Commission
IED	Intelligent End Device
IEEE	Institute of Electrical and Electronics Engineers
ISO	International Organization for Standardizationr
KDC	Key Distribution Center

KNN	K-nearest neighbor
LCPC	Low Complexity Parity Check
LDPC	Low Density Parity Check (LDPC
LKH	Logical Key Hierarchy
LTK	Long-Term Key
MiM	Man-in-the-Middle
MSU	Master Station Unit
MTU	Master Terminal Unit
NASA	National Aeronautics and Space Administration
NERC	North American Electric Reliability Council
NSA	National Security Agency
NTRU	Nth Degree Truncated Polynomial Ring
OCSVM	One-Class Support Vector Machine
OTS	One-time Digital Signature
P-P	Peer to Peer
PK	Private Key
PKSK	Public key Signature Key
PLC	Programmable Logic Controller
QBER	Quantum Bit Error Rate
QFT	Quantum Fourier Transform
QK	Quantum Key
QKD	Quantum Key Distribution

R-S	Reed Solomon Protocol
RTU	Remote Terminal Unit
SCADA	Supervisory Control and Data Acquisition
SK	Session Key
SKE	Scada Key Establishment
TVP	Time-Varying Parameters

Acknowledgements

I want to thank my supervisor Prof. Srinivas Sampalli, for his knowledge, support, motivation, and immense patience. Due to his guidance, a significant part of our thesis as a manuscript named "A Survey of Security in SCADA Networks: Current Issues and Future Challenges" is accepted for publication in IEEE Access. It represents the Chapter 1 and Chapter 3 of the thesis. However, it is available in early access and therefore, we have cited our work in the thesis.

I would also like to thank my family and friends, who were always there for me.

Chapter 1

Introduction

SCADA systems are used as control systems for monitoring industrial processes such as oil mining, electric grids, traffic control systems, water treatment plants, space stations, and nuclear systems. Modern SCADA systems have been exposed to a range of cyberattacks since they use open-access networks to leverage efficiency. Failure to secure SCADA systems can be catastrophic [49]. For example, a malicious attack can take control of the power supply to a city, shut down the water supply system, or cause the malfunction of a nuclear reactor[37].

Current SCADA systems have a number of added features which increase the system complexities and are thus difficult to maintain. Some of the added features include control logic, communication protocols, user interfaces, and security. For example, many organizations do not tolerate data delay or data loss. Added features like firewall function and anti-virus software processes can lead to delayed delivery of data [92]. The systems must operate continuously and in tight timing [67]. Moreover, the communications are vulnerable to various threats. In the past few years, the number of cyber-attacks, in general, is rising and has been affecting the power station, water, gas, and nuclear control systems. The pattern of cyber-attacks has also evolved beyond the simple attacks such as Denial of Service or Man-in-the-Middle[67][37].

In December 2015, due to a successful cyber-attack on SCADA, 2,30,000 people were left without power for hours in Ukraine. After a year, another similar attack hit the country. This attack was launched by using spear-phishing emails and is still in practice against industrial organizations. According to the U.S. Department of Justice, there was an attack on a small dam in Rye Brook, New York in 2013. The hackers gained access to the core command-and-control system by using a cellular modem. Although the breach occurred in 2013, it remained unreported until 2016. Furthermore, according to FBI and Homeland Security last year's joint report [22], there have been cyber-attacks on nuclear power plants throughout the U.S., in which

the control systems were targeted. The main motive and severity of the attacks are not known, but the method used for the attack was spear phishing[37].

SCADA networks also comprise of resource-constrained devices such as Remote Terminal Units or Programming Logic Units, and these devices require lightweight ciphers. Traditional intrusion detection systems (IDSs) such as firewalls are now unable to protect from new threats [66]. Robust security schemes involving machine learning to detect intrusions and encryption algorithms are essential to ensure secure encrypted communication between nodes in SCADA networks. These threats and attacks have motivated researchers and organizations to develop new robust and secure techniques for SCADA networks[37].

Although there are several survey papers on the security threats, key management schemes, and intrusion detection systems in SCADA networks [77][79][59], the reviews do not specify a comprehensive comparison of the various schemes., Sajid et al.[83] have provided an excellent survey on the security and challenges of the SCADA systems. However, the paper does not provide a comparison of all the security protocols and standards for SCADA systems. Motivated by this, the thesis includes an extension of the survey provided by Sajid et al. [83]. It gives a review of the SCADA communication structure and the recent threats faced by them. It then provides a classification and comparative study of the existing security protocols used and proposed to date. Based on the analysis, it also provides the limitations of each of the standards and protocols[37].

Furthermore, the emergence of the quantum computer is not only valuable but also a risk to cyber field. According to Shor's algorithm and Grover's algorithm, a quantum computer can crack classical encryption schemes, including Elliptic Curve Cryptography (ECC) [26][34]. The existing standards and protocols are not only vulnerable to traditional attacks but also quantum attack[37].

The microchip circuits developed at QETLabs can generate and distribute keys encoded in qubits by using the quantum properties of superposition and entanglement. This chip presents an opportunity to apply Quantum Key Distribution (QKD) to resource-constrained devices[89].

1.1 SCADA Communication Architecture

SCADA systems consist of several entities organized in a hierarchical structure [66]. They are used in monitoring various kinds of infrastructure and industries. They comprise the integration of data acquisition systems, data transmission systems and Human-Machine Interface (HMI) [66]. The HMI is a user interface that connects a person to a device. It is mainly used to visualize data, and monitor production time, machine inputs and outputs. Figure1.1 illustrates a generic SCADA network communication architecture [35][84][26]. The HMI is a software interface while the hardware components are as follows[84][26][37].

- Master Station Unit or Master Terminal Unit (MSU/MTU) is the control center of a SCADA network.
- Sub-MSU or Sub-MTU acts as a sub-control center. However, it is not needed in some cases. The MSU can connect to the remote station units directly.
- Remote Station Units are Remote Terminal Unit (RTU), Intelligent End Device (IED) and Programmable Logic Controller (PLC). They are used to monitor sensors and actuators to collect data values.

A communication link is shared between the MSU and Remote Station Units. Various types of communication links may be used, such as Ethernet, optical fiber line, satellite, and wireless.

SCADA system architectures have four typical architectural styles as follows [9][37]:

- Monolithic: In 1970s, controlled units or MTUs were hardwired to RTUs.
- Distributed: In 1980s to 1990s, MTUs and RTUs communicated using communication protocols and servers. However, they did not allow Internet connection.
- Networked: In 2000s, SCADA architecture started using external networks like the Internet.
- Web-based SCADA: Currently, users can access SCADA systems using web browsers and mobile devices.

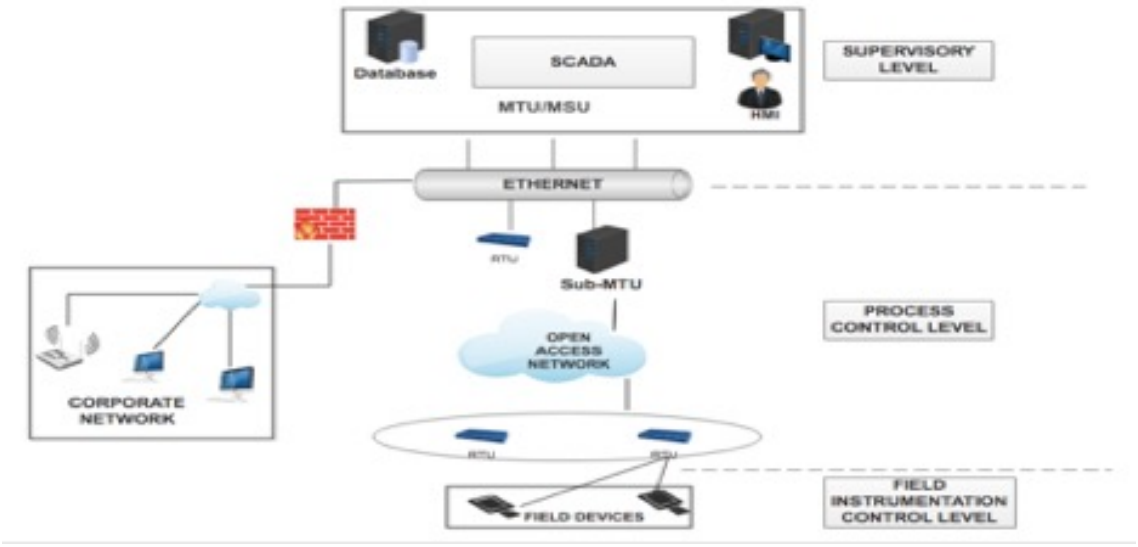

Figure 1.1: SCADA network communication architecture

The evolution of SCADA has led to increased complexities. Some of the features responsible for this are the following[37] [92][9].

- Addition of new components such as computers, operating stations, communication servers and other types of resources.
- Increase in amount of data exchange between units with the increase in the number of components.
- Increase in the amount of interactions between the system components.
- Usage of firewalls and anti-virus software that consequently slows down the processing power of the system and leads to delay in data transfer to other units.

Thus, as the size of the SCADA architecture and added features increase, the complexity of the SCADA architecture also increases. This makes managing a large amount of data more difficult, leading to loss of data availability. Furthermore, it makes the SCADA architecture susceptible to cyber-threats[92][9][37].

1.2 Security threats faced by SCADA networks

Like any other system or network, a SCADA network faces the following threats[49][26][37].

- Loss of availability can cause power outages and can have a negative impact on the efficiency of power supply chains. The operation, if completed after a deadline, may have a cascading effect in the physical domain. Thus, achieving availability as a security goal should be one of the primary objectives of a SCADA network.

- Loss of integrity is a scenario when the attacker modifies the data, and thus, the receiver receives the changed data. This type of scenario is achievable by launching a Man-in-the-Middle(MiM) attack, which can further result in malware injection and IP spoofing.

- Loss of confidentiality can be achieved by eavesdropping on a channel. It leads to the loss of privacy and stealing of data as private data is exposed.

- Repudiation is where the sender denies they have sent the data at that time.

- Lack of authentication in the Distributed Network Protocol 3.0 (DNP3) used in SCADA systems which can lead to an impersonation attack [42].

1.3 Attacks on SCADA networks

The usage of Internet connectivity, cloud computing, wireless communications, and social engineering on SCADA networks have made its architecture vulnerable [49]. One of the main reasons for the vulnerabilities in SCADA is the lack of strong encryption and real-time monitoring[37].

Attacks can occur at all layers from the supervisory level to the field instrumentation level [100]. The most common attacks are described as follows [100][94][44][40][38][37].

Eavesdrop: It can be of two types: Passive eavesdropping and Active eavesdropping[64]. The communication network can be wired or wireless. By accessing the network between the MTU and sub-MTUs or RTUs, the invader can install eavesdropping equipment in the network [99]. The tools that can be used to launch this type of attack are Wireshark, tcpdump and, dsniff [93].

Man-in-the-Middle (MiM): This occurs when the attacker is in between two units and fetches the private information. The most common MiM attacks are the following

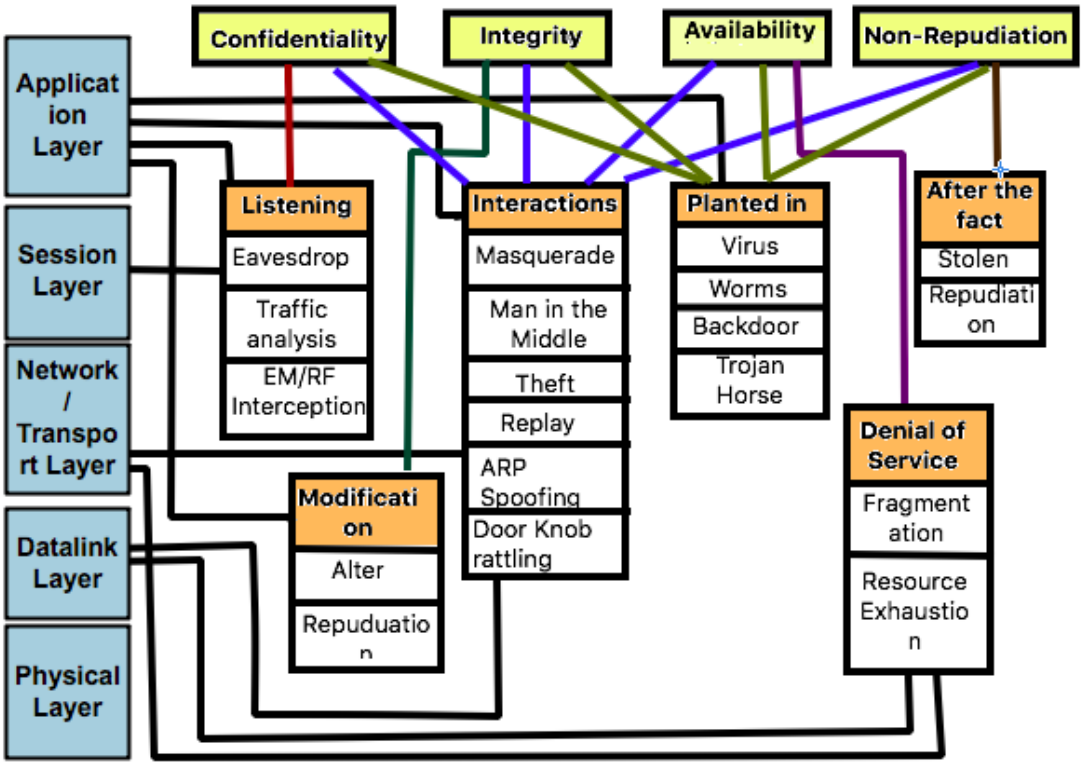

Figure 1.2: Classification of SCADA attacks in terms of security requirements and OSI layers.

[64]: Session Hijacking network server, IP Spoofing and Replay attack. In MiM attack, the intruder monitors the traffic and injects abnormal data during the transmission and sends it to the receiver [99]. In case of a successful session hijacking and IP spoofing, it takes over the session and maintains the connection. The spoofing helps the attacker to go undetected [6]. A few tools that can be used are Ettercap, SSLStrip and,Evilgrade [1][37].

Masquerade: The attacker uses a fake identity to pretend to be a legitimate user and steals information from the system or the network. By launching IP Spoofing and a brute force password attack, they can use stolen passwords and logins to gain unauthorized access [99]. A few examples of tools are Ettercap, Arpspoof and Brutus[1][37].

Virus and worms A malware is a malicious software or a program that corrupts the data stored in the computer. They can also lead to Distributed Denial of Service attack. Virus and worms are types of malware [51]. The intruder can send a file

containing malicious code to the MTU after launching MiM or masquerade attack. For example, the virus or worm can spread through sending e-mail attachments, web link and peer-to-peer file sharing networks [51].Any malicious code which is self-replicable and attached to .exe file in the device [51].

Trojan Horse: This is a type of malicious program disguised as a harmless file. However, unlike a virus, a trojan horse is not self-replicable. Therefore, hackers use social engineering tactics to transfer this type of virus [80]. After launching IP Spoofing or social engineering, the intruder can inject innocent looking malicious and executable code and send it as a web link or a free download to the target system. Thus, the hacker can gain access and hack the control systems[80]. Social engineering. For example, web links offering free software download.

Denial of Service (DoS): This is a type of attack where a legitimate user is denied access to a resource. It attacks the availability requirement of a network [48]. An infected RTU by virus or worm can send random IP packets to the MTU and thus consume network bandwidth. It further leads to resource starvation. For example, Slowloris, GoldenEye for operating system Kali Linux. And, another tool named Low Orbit Ion Cannon (LOIC)name=LOIC,description= Low Orbit Ion Cannon[88].

Fragmentation: Fragmentation attack is a type of DoS leading to unavailability of resource [2][85]. It involves sending of over-sized datagrams. In this type of attack, the sizes of the sent datagrams are greater than network's maximum transmission unit[85]. Tools used to launch DoS attack can be used for Fragmentation attack.

Cinderella: The objective of this type of attack is to expire the security software license. The hacker disguises their ID as a legitimate user and gains access to system by using a brute-force attack. Then the internal network clock is changed to expire the security software prematurely, thus increasing the network vulnerability[56]. Attackers can use the tools that are used to launch masquerade and brute-force attack. For example, Ettercap for masquerade [43]. Ncrack, Hydra and Hashcat for brute-force attack [43].

Doorknob rattling: The type of attack when the failed attempts of a brute-force remains hidden from the detection system of the network[100]. At first the attacker will launch masquerade attack. Then, they try to attempt a random combination of username and passwords repeatedly on different devices to gain access. So, this

leads to a few failed attempts. If the failed attempts are going undetected, this kind of attack can be successful[100]. Few tools that can be uses are: Ettercap for masquerade [43] and Ncrack, Hydra and Medusa for brute-force attack [43].

They can also be categorized based on attacks on hardware, software, and network connection [100].

- Attack on hardware: This is a scenario where the hacker gets unauthenticated access to the units and tampers with them or their functions. The primary challenge in securing hardware is access control. For example, the doorknob-rattling attack [100].

- Attack on software: The SCADA system utilizes a variety of software to enhance its efficiency by fulfilling the functional demands. However, due to poor implementation, it is vulnerable to SQL injection, trojan horse and buffer overflow. These are a few examples of attack on software [100].

- Attack on network connection: The attack on communication stack can be on the network layer, transport layer, and the application layer. Figure 1.2 mentions a few attacks on the layers of the Open Systems Interconnection (OSI) model[100]. The Application and session layers perform network processes to applications. The Transport layer manages transportation concerns between host and confirms data transport reliability. The Network layer routes data packets and decides the best path for data delivery. The Datalink layer defines the format of data to be transmitted. The Physical layer is responsible for binary transmission.

1.4 Possible Attacks Using Quantum Computer

In recent years, the rapid development of quantum computers has poised a threat to cybersecurity. A quantum computer can solve and crack mathematical operations. For example, the problem of factoring enormous numbers which is the core of any encryption scheme. The primary two types of attack a quantum computer can launch are:

- Man-in-the-Middle(MiM) attack between two victims to fetch the information passing over the communication channel.
- Brute-force attack launched to decrypt the fetched cipher over the channel.

Currently, a quantum algorithm can be used to launch only these two main attacks on the existing security structure of SCADA networks.

1.5 Defense against Attacks

Modern SCADA networks relies on internet connectivity, cloud computing and wireless communications. These has made its infrastructure susceptible to various attacks. Mostly, the Man-in-the-Middle (MiM) attack is the source of every other attack. Thus the proposed quantum resistant security scheme mainly focuses on the prevention of the MiM attack. It also provides security against the following attacks discussed in the Table 1.1.

1.6 Research Problem

- Existing standards do not provide strong confidentiality,integrity and availability.
- Existing intrusion detection systems do not provide confidentiality.
- Existing key management protocols fails to provide confidentiality and availability.
- Current security scheme does not provide resistance against an attack launched by a quantum computer.

1.7 Contributions of the thesis

The primary contributions of the thesis are the following:

Table 1.1: Attacks defended

CLASSICAL ATTACKS		Description
Attack against Confidentiality	Packet Sniffing	The intruder intercepts the two-way traffic and fetch sensitive data. By using Wireshark and Tcpdump, sniffing can be attained.
	Eavesdrop	The intruder can install an eavesdropping equipment in the wired or wireless network between the RTU and MTU. Tools that can be used are Wireshark and dnsiff.
Attack against Integrity	Man-in-the-Middle attack (MiM)	In MiM attack, the intruder monitors the traffic between the nodes. The data packets traded between two victim nodes are captured. The intruder then injects abnormal data and sends it to the receiver. It can launch IP spoofing and Session Hijacking attack. Few tools used to launch MiM attack: Ettercap, SSLStrip and Evilgrade.
	Session Hijacking	After a successful MiM attack, the intruder accesses the information and services in the MTU an RTU. It accesses the session ID and launches replay attack. A few examples of tools are Ettercap an Evilgrade.
	Data Injection	The intruder can successfully alter the data after launching MiM attack. Few tools that can be used are Wireshark and Ettercap.
	Replay Attack	The attacker can launch replay attack by performing session hijacking and IP spoofing. By imitating as a friendly unit and using the session id, it stores the old data and send it to other units later. Tools that can be used are Ettercap, Evilgrade.
Attack against Authentication	Masquerade	By using IP spoofing, the attacker uses a fake identity to pretend as a original unit and, steals essential data from the system.Tools that can be used for launching this type of attack are Ettercap, Arpspoof and Brutus.
Attack against Availability	Denial of Service (DoS)	This kind of attack occurs when a compromised unit is used to target a system by sending huge traffic or large amount of junk data. A unit can be compromised in several ways after a successful MiM attack. The examples of DoS attack tools are Slowloris and GoldenEye.
QUANTUM ATTACK		
Quantum Attack	Brute Force Attack by a Quantum Computer	The emergence of quantum computer brings with it benefits as well as risks to the cyber field. A quantum computer is way faster and efficient than traditional computers. Using Shor's and Grover's algorithm, a quantum computer can launch brute force attack and crack the traditional encryption schemes in a brief time. One such problem is Elliptic curve cryptography (ECC).

- It provides researchers and organizations with a report that discusses and analyzes the schemes and efforts proposed to secure the SCADA networks. It also gives a comparative study of the existing standards and schemes.
- It identifies a new threat based on quantum computing faced by SCADA.
- A new security scheme been proposed for SCADA networks has to protect against traditional as well as the quantum attack. Furthermore, the proposed scheme acts as both encryption and intrusion detection system. The scheme generates a signcrypted message by using BB84 protocol and one-time digital signature. Unlike other signcryption schemes, this scheme does not depend on a third-party.

1.8 Thesis Outline

The rest of the thesis is organized as follows; Chapter 2 describes the background work that was beneficial to understand the proposed scheme. Chapter 3 provides the existing standards and protocols developed and used for SCADA systems. It also gives a comparative study of the current security schemes. Chapter 4 describes the proposed scheme for SCADA networks. Chapter 5 displays the formal analysis and the experimental results of the proposed scheme. Lastly, Chapter 6 provides the conclusion and future work.

Chapter 2

Background

This thesis proposes a novel signcryption scheme which involves BB84 protocol, error correction protocol, and one-time digital signature. This chapter provides a background survey on existing procedures used in existing quantum key exchange protocols as well as contributed to developing the proposed scheme.

2.1 Existing Error Correction Protocols used in Quantum Key Exchange Systems

This section provides a survey on the existing error correction protocols used in quantum key exchange protocols. A quantum channel in the presence of an intruder or the cause of any environmental factor can create noise. This disrupts the key exchange via the channel. To resolve these errors, error reconciliation protocols are used. After the quantum key exchange, the system follows a process called Key Sifting. The sender and receiver discuss their choice of basis via the public channel with dubious security. After the discussion, they estimate the errors present in their respective keys. To reconcile the errors while preserving security in the exchanged key, the following protocols have been proposed and used[46].

Parity bits: All of the protocols use parity bits or check bits. They are adhered to a binary stream to denote the total number of 1-bits in that stream is even or odd. There are two types of parity bits: Even parity bits and Odd parity bits[41].

- Even Parity Bits: The number of 1s are counted. If the count is odd, parity bit is 1. Else, it is 0.

- Odd Parity Bits: The number of 1s are counted. If the count is odd, parity bit is 0. Else, it is 1.

2.1.1 Cascade

The Cascade protocol is the most famous error reconciliation protocol developed and used by Bennett and Brassard[46]. In this scheme, the sender and the receiver agree on a block size and seed based on the error estimation. Then, both of them segments their keys into agreed sized blocks. They exchange the 2-bit parity of each of their blocks. When there is a parity mismatch, they perform a binary search in the corresponding block to find a single bit error. After the search, the error is detected and resolved. This is the first complete pass of the protocol. In each succeeding pass, the block size is doubled, and the same process is repeated[46].

It is a simple protocol which is computationally efficient but with large amount of interaction between sender and receiver.

2.1.2 Winnow

In 2003, Butler et al. [46][96] proposed a protocol named Winnow, which significantly reduces the interaction between the sender and receiver. Instead of using binary search, it uses hamming code to identify and correct single-bit errors. Furthermore, this protocol introduces errors in the key during the process in case of non-uniform error distribution[46][96].

In the Winnow protocol, the initial key is segmented into blocks. Prior to segmentation, the sender and receivers perform error estimation. Based on the error rate, they agree on size of the blocks. The block size is determined in increasing powers of 2. For example, 8,16,32,64,128. They exchange and compares the parity of each block. In case the parity does not match, they compare the syndrome deduced by using Hamming hash function. The number of passes depends on the error rate[46][96].

Hamming Code Structure[46]: A Hamming Code follows even parity bits. They are used to detect two-bit errors and resolve single bit errors. In a Hamming codeword, all bit positions that are power of 2 are denoted as parity bits. The rest bits is data. For example, when Hamming function is applied on a message with four binary digits, three parity bits are added to the digits. It gives a (7,4) codeword as shown in the figure **??**. D7, D6, D5 and D3 are the data bits. P4, P2 and P1 are the parity bits.

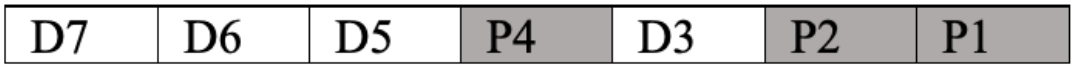

Figure 2.1: Hamming Code Structure

Hamming Code Algorithm[46][96][90]:

Step 1: To determine the value of parity bits, the sequence of bits is alternatively checked and skipped. For example, a sender sends data of binary digits 1101 to the receiver.

For P1, check 1 bit and skip 1 bit. That is, check for these positions: (1, 3, 5, 7, 9 . . .). For P2, check 2 bits and skip 2bits. That is check for these positions: (2,3; 6,7; 10,11; . . .). For P4, check and skip 4 bits which gives the follow bit positions: (4,5,6,7; 12,13,14,15; . . .).

For P1, the parity for D3, D5, D7 or 101 is 0. This is because it follows even parity. Since the number of 1 is even, the parity is 0. Else, it will be 1. Therefore, P1 is 0.

For P2, the parity for D3, D6, D7 or 111 is 1. Thus, P2 is 1. Similarly, the P4 for D5, D6, D7 or 011 is 0. Therefore, the obtained Hamming codeword is 1100110 for the message 1101.

Step 2: At the receiver's end, the bits on positions (1,3,5,7), (2,3,6,7) and (4,5,6,7) are checked using even parity. If P1, P2 and P4 is 0, then no error is present in the received codeword.

Step 3: If there is error present in any of the parities, the parity value will be 1. For example, if P1 and P4 is 1, then the received codeword is wrong. The error word will be P4, P2, P1 which gives 101. The decimal value of 101 is obtained which is 5. This depicts that fifth bit of the codeword is incorrect and thus, it is flipped.

As per the property of Hamming code, it can detect and correct one error per block. If that block contains a large number of errors, it introduces a new error to the block. To avoid this situation, the smaller block size can be used, but it results in a large number of blocks. This consumes more parity bits to exchange and increases the amount of information leaked. However, if larger block size is used, a new error is introduced[46].

2.1.3 Low Density Parity Check (LDPC)

The LDPC is defined by parity check matrix H and a generator matrix G. The dimensions of both the matrices is $m * n$ such that$n * k = m * j$. The 'j' is the number of 1s in each row and the 'k' is the number of 1s in each column. The 'n' is the block length. The Generator matrix is denoted as G and the Parity matrix is denoted as H. The j and k are small compared to the number of rows and code length. Therefore, H has a low density of 1s. The H is called a low-density parity check matrix. And, the code defined by H is called low-density parity check code[46].

In quantum transmission, the Generator matrix is not required, and the Parity matrix can be perceived as a Tanner graph[46]. In H matrix, each row is Check node and each column is Variable node. The check node signifies the parity check based on syndrome calculation. The variable node represents the single bits of the message[46].

In a single information exchange, the LDPC can resolve all the errors in the transmitted key. Unlike Cascade and Winnow, no parities or no segmentation of the key is followed. The sender calculates the syndrome for the key and sends it to the receiver. The receiver calculates the syndrome based on his obtained key. It then uses the sender's syndrome to detect and resolve the errors in the key. The receiver uses

a decoding algorithm to detect the location of errors in the key. The most common algorithm used in LDPC is the Sum-Product algorithm[46].

In Winnow, the hamming code is implied on small message segments and it can only correct single-bit errors. In LDPC, the syndrome is larger. Thus, it can correct multiple errors with less communication overhead as compared to Winnow and Cascade. However, the computational complexity is significantly higher than that of the other two protocols.

2.2 Error Correction Protocol proposed for future wireless networks application

2.2.1 Low Complexity Parity Check (LCPC)

In 2018, Salah A. Alabady et al.[46][11] proposed a low complexity parity check code for wireless network applications. The LCPC has less complexity and requires less memory as compared to that of LDPC. The LDPC performs better when larger codeword and a low-density parity matrix is used. A larger codework consumes means memory and computational requirements. It also leads to complex decoding. Furthermore, the LDPC performance depends on the characteristic of a parity matrix[46][11].

In LCPC, the message in the binary form is segmented into equal bits. Then, LCPC is applied on that segmented source data and a codeword is obtained. This codeword is transmitted through public channel. The receiver upon receiving the codeword checks for errors and resolve them in the codeword by using syndrome. Then, the codeword is decoded[11].

The LCPC has the following methods to encode and decode the codeword[11].

Step 1, Segmentation: The source data is segmented into equal bits. In this thesis, LCPC (9,4) is discussed implied of the source data of 56-bits. The source data will be segmented into 4-bit length blocks. The segmented data will contain fourteen 4-bit blocks.

Step 2, LCPC encoding: : LCPC (9,4) is imposed on each 4-bit block that is x_i. In this step, the codeword is generated by using parity-check matrix and generator matrix. For example, source data (SD) = x_i is 1010.

For x_i = 1010, each bit is represented as ß1 = 1, ß2= 0, ß3 =1, ß4= 0. The LCPC follows even parity. The parities are obtained using the following equations.

P5 = ß1⊕ß2⊕ ß3⊕ß4
P6 = ß1⊕ ß2⊕ß3
P7 = ß1⊕ ß2⊕ß4
P8 = ß1⊕ ß3⊕ ß4
P9 = ß2⊕ ß3⊕ ß4
⊕ is the xor operator symbol.

Thus, deduced parity bits are P5 =0, P6 = 0, P7 = 1, P8 =0 and P9 = 1. When parity bits are added to the x_i, it gives the codeword CD = 101000101. This codeword is sent to the receiver. Furthermore, the Generator matrix G and the Parity check matrix H obtained are as shown in Figure2.2, 2.3 [11]:

$$H = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Figure 2.2: Parity Check matrix

$$G = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \end{bmatrix}$$

Figure 2.3: Generator matrix

Step 3, LCPC Decoding: The receiver receives the codeword with or

without error. It performs three sub-steps. *Sub-step1:* It detects any error present in the codeword. *Sub-step2:* If error is detected, it checks and determines the error pattern. *Sub-step3:* Finally, it corrects the error and decode the codeword. Let the codeword 'c' is transmitted and vector 'r' is the received codeword such that it satisfies the following equation.

$$r = c + e \tag{2.1}$$

where 'e' is the error vector.

For detecting and correcting errors, it uses syndrome vector which indicates whether the equation is satisfied for that particular codeword. If the value of syndrome 's' is zero, it denotes that no error has occurred, and r is the correct codeword c. Otherwise, it detects that the codeword is received with an error.

If H is the parity-check matrix of c, then,

$$HrT = H(c + e)T = HcT + HrT \tag{2.2}$$

Since, HcT for any codeword is 0. Thus,

$$HrT = HeT = s \tag{2.3}$$

HrT is the syndrome of r.

When the syndrome has a non-zero value, the column of H matrix which is a scalar multiple of the 's' is searched. If no such column is found, the code contains more than one error and it fails to correct multiple errors. Else if the syndrome is 'α' times that particular column 'j', then the vector is added with '$-\alpha$'on the jth bit-position and with 0 on the rest bit-positions.

When the error is corrected, the codeword is decoded using a masking process for the last left four-bits of codeword. The logic operator AND is used between the corrected codeword and 111100000.

Although LCPC is way more efficient in respective of computational cost and memory requirement as compared to other protocols, it fails to correct burst

errors. In quantum transmission, there are chances of burst errors in the quantum key transmitted and sifted. Cascade and Winnow protocols can correct single-bit errors and LDPC can correct multiple bit errors.

Error Correction Protocol used in the Proposed Security Scheme: In the proposed security scheme, Reed-Solomon(R-S) protocol is used for error reconciliation in the key[27]. Due to high computational and complexity cost, LDPC requires and exhausts more hardware resource which hampers the system applications. Reed-Solomon is a multi-bit error correcting protocol with low computational overhead as compared to LDPC[27].

2.3 Existing research that uses Quantum computing to solve security problems

Researchers have proposed and developed various quantum cryptography protocols — for example, B92, Six-State, and SARG04 protocol[71]. In our thesis, we are deploying BB84 in our security scheme since it is a feasible option for resource-constrained devices[81]. Various researchers and organizations are practicing and evolving quantum key distribution. The following points are a few examples[71][50].

- In 2017, M. Karpinski et al.[50] proposed a cryptographic scheme for computer-aided voting system using quantum bit commitment and quantum secret sharing protocol.It also uses BB84 protocol where Alice generates a random series of qubits to Bob based on voter's decision.

- BBN Technologies, funded by US Defense Advanced Research Projects Agency(DARPA), deployed the DARPA Quantum network in Massachusetts, USA. Researchers practice Quantum Key Distribution(QKD) protocol for traffic processing on the internet. They have deployed a Virtual Private Network (VPN) based on quantum properties[71].

- A group of industrial and research organizations, collaboratively, developed

a project named The SEcure COmmunication based on Quantum Cryptography(SECOQC) in Europe. It is a quantum network which utilizes the QKD protocols as well as key distillation authentication schemescite[71].

- In 2014, The U.S. National Security Agency (NSA) reported that they are developing a quantum computer skilled in breaking the widely used encryption algorithms[71].

2.3.1 Post-quantum Digital Signature: One-time Digital Signature

One-time Digital Signature (OTS) scheme is based on hash-based signatures. One of the well-known OTS schemes is the Lamport-Diffle scheme where the key used for signing (sk) is randomly generated and the verification key (vk) is generated by applying hash function on the sk[60][10][28]. This feature of Lamport Signature scheme has been used in the proposed scheme to generate signcrypted message.

Chapter 3

Existing SCADA Security Schemes

An attack on a SCADA system may have many adverse effects. Due to this reason, organizations and researchers have been putting much effort into developing standards, protocols, and security schemes. The existing security schemes can be categorized based on: current standards, detection of SCADA attacks, and prevention of SCADA attacks[37].

Classification 1: Current standards can be divided into two categories: Standard Providing Guidelines and Standards acting as crypto-suites. These standards are used in practice depending on the particular industry's requirements. However, the mechanisms of thwarting attacks in the standards are either not clearly discussed or, are not strongly secure.

Thus, to add more security in the existing standards for SCADA, many researchers have proposed novel schemes. In this thesis, the academic effort has been further classified into two following categories.

Classification 2: Detection of SCADA attacks consists of all the proposed intrusion detection systems for SCADA networks. The main objective is to overcome the lack of availability that is one of the security requirements.

Classification 3: Prevention of SCADA attacks consists of all the key management protocols proposed to secure the communication between the units.

3.1 Current Standards

Throughout the world, over 10 countries have proposed more than 40 standards and protocols. The available standards are described as follows[73][82]. Few of the standards provide guidelines to secure an infrastructure from physical and cyber-attacks. Furthermore, the remaining standards include a major part that acts as a crypto-suite. In this thesis, they are categorized into two [37]: 1) Security guidelines-based Standards and 2) Crypto-suites based Standards.

3.1.1 Security Guidelines based Standards

IEEE 1402

Institute of Electrical and Electronics Engineers (IEEE) 1402-2000 is an IEEE Guide for Electric Power Substation Physical and Electronic Security. The Power Engineering Society/Substations of IEEE sponsors the standard. It discusses security issues caused by human intrusion at power supply substations along with methods and schemes to mitigate physical and electronic intrusions[7].

In the guide, the intrusions are classified into four main categories: pedestrian, vehicular, projectile, and electronic intrusion[82][7]. The thesis also categorizes the security methods used at power control substations[82][7].

The computer security systems include using passwords, dial-back verification, selective access, virus scans, and encryption. The guide also explains the substation security plan and categorizes it into three questions: Why is the plan required? Who may monitor the plan? What security methods are needed? According to the guide, these are the main criteria on which the security plan should be executed [82][7].

IEEE 1402 does not solely focus on the information security. Rather, it gives a broad and general guideline for physical as well as cyber security.

ISO 17799 – "Information Teachnology- Code of paratice for Information Security Management"

The International Organization for Standardization (ISO) published ISO 17799 in December 2000. The ISO 17799 is an international guideline for monitoring information security management of any organization [82]. The standard refers to information as an asset that is valuable to industry. The main objective of the standard is to protect the asset by preserving confidentiality, integrity and availability[20]. ISO 17799 provides a structured guideline to control security and perform security risk assessment. It provides the following benefits[20].

- Organizational Security
- Asset Classification
- Personnel security
- Physical and environmental security
- Network management that involves media handling, backup schedules and logging.
- Access control
- Maintenance of cryptographic controls and system integrity.

ISO 17799 is the one standard that is dedicated to Information Security Management. However, ISO 17799 does not provide any evaluation methodology of a security scheme. It also does not deal with the requirements of functional and security components in an organization. ISO 15408 was developed in 2004 to alleviate some of these issues.

ISO 15408 – "Common criteria for Information Technology Security Evaluation"

ISO developed the "Common Criteria got Information Technology Security Evaluation" in January 2004 [45]. The criteria are used to evaluate various functional classes as listed as follows [33].

- Audit
- Communication
- Cryptographic support
- User data protection
- Identification and authentication
- Security Management
- Privacy
- Security functions protection
- Resource Utilization
- Access
- Trusted path/channels

It has three sections. ISO 15408-1 provides the introduction and general model. ISO 15408-2 provides the functional security components, and ISO 15408-3 discusses the security assurance components [82].

However, the report does not focus on the utilization of cryptographic designs in communication and control applications[82]. Furthermore, it does not uniquely focus on the need of physical security in SCADA structure.

NERC Security Guidelines – "Security guidelines for the electricity sector: PHYSICAL SECURITY"

On June 14, 2002, North American Electric Reliability Council (NERC) releases a version 1.0 of NERC Security Guidelines discusses physical and cyber security along with the general practices for protecting the power supply infrastructure systems [82].

The general guideline focuses on the need of the physical security to maintain the integrity and availability of electric power systems, for example, promoting and deploying the security standards and procedures, periodic evaluation of the security measures, monitoring and reporting threats to the operating section, and quick recovery of the delivery services if damaged[68].

The report also guides to follow a strategy 'Protection in Depth'. The objective of this strategy is to delay the progress of an attacker. This buys time to the organization to defend and recover against the attack[68].

However, the security guidelines focus mainly on physical security. In 2003, NERC produced a report that deals with cyber security parameters.

NERC 1200 – "Urgent action standard 1200 – cyber security" and NERC 1300 – "CYBER SECURITY"

NERC developed a temporary standard named "Urgent Action Standard 1200" for setting a set of security requirements for the energy industry infrastructure. NERC adopted this standard on August 13th, 2003 for a one-year period and later, it extended the standard till August 2006[82].

NERC developed NERC 1300 to replace NERC 1200 by addressing the security requirements and recommendations mentioned in NERC 1200 [82][8]. NERC 1300 focuses on both physical and cyber security. The report has a section that implies that a responsible industry should follow the System Security Management to prevent any malicious cyber activity. The Management section mainly involves the following security measures [8] :

- Account and Strong Password management.
- Using anti-virus monthly.
- Performing vulnerability assessment at least annually.
- Preserving and auditing system logs quarterly.
- Using operating status monitoring tools.
- Back-up of information on computer systems.
- Disabling unused ports.

NERC 1200 and NERC 1300 are security guidelines for the energy industry infrastructure. They do not provide security features for the oil and pipeline infrastructure. Therefore, the American Petroleum Institute developed a standard that

Table 3.1: Concerns addressed in API 1164

API 1162	Concerns/Areas Addressed
First edition	Access control Secure communication Classification of data distributed Physical complications for example disaster recovery Operating systems Network Designs Management systems Field devices configuration and local access

provides security guidelines for control systems of oil and pipeline systems.

API 1164 – "Scada Security"

API 1164 has three editions. The first edition was released in September 2004. It specifies guidance to secure the SCADA system used in the oil and pipeline infrastructures[82][69]. It addresses the following issues mentioned in the Table 3.1[82].

The second edition is the API – "Security Guidance for the Petroleum Industry." Oil and gas infrastructures utilize this standard to prevent terrorist attacks [82].

The American Petroleum Institute(API) and the National Petrochemical and Refiners Association mutually developed the third edition named API- "Security Vulnerability Assessment Methodology for the Petroleum and Petrochemical Industries". It is utilized for evaluating various kinds of threats, vulnerabilities, and aftereffects of terrorist attacks [82].

The above-discussed standards are general guidelines to protect the infrastructure of an organization. They do not involve any in-depth discussion of cryptographic algorithms or any technical methodology to detect or protect from any attack.

However, the following standards use crypto-suites.

3.1.2 Crypto-suite Standards

IEC 62210 – "Data and Communication Security"

In 1999, IEC 62210 was developed by the International Electrotechnical Commission (IEC) as the report of IEC TC 57 AHWG06. Later, AHGW06 was systemized into Working Group (WG) 15 upon Data and Communications Security. Later, it was published in 2003. The IEC TC57 WG15 developed the cybersecurity standards for power control system communications [82].

The working group report describes the security process for the power control systems which involves the corporate security policy, network security protocol, and the end to end application security. The security scheme was also utilized for encrypting communication in the network[82].

AHWG06 issued the report recommending establishing the following tasks[82]:

- Consequence analysis combined with ISO 15408
- Attention to the application layer
- Address key management
- Address end-to-end security

However, the above recommended tasks were challenging to resolve at that time[82]. Therefore, the following standard was developed as an extension of IEC 62210.

IEC 62351 – "Data and Communication Security"

International Electrotechnical Commission (IEC) developed IEC 62351 to address the deficiency in IEC 62210. The standard is classified into as shown in Table 3.2[86].

Table 3.2: Classification of IEC 62351

Sections	Schemes used
Security for profiles including TCP/IP	It uses Transport Layer Security (TLS) for secure transactions over the internet. It provides confidentiality, integrity, and authentication.
Security for profiles including MMS	For Transport Layer which includes layer 1 to layer 4 of the OSI Reference Model, Transport Layer Security is used.,The report describes a set of protocols, how to use them, and the requirements for Application Layer which includes layer 5 to layer 7 of the OSI Reference Model.
Security for derivatives (DNP 3.0)	For network versions which run over TCP/IP, the standard uses TLS encryption. For the serial version, it uses an authentication mechanism named Hashed Message Authentication Code (HMAC).
Security for IEC 61850 peer-to-peer profiles	For client/server, the standard utilizes TLS and MMS. For Generic Object-Oriented Substation Events (GOOSE), it uses analog and digital multicast.

Using TLS security, IEC 62351 provides defense mechanisms against various attacks including spoofing, message replay attack and to some extent Denial-of-Service (DoS) attacks. However, it involves simple encryption schemes.

Immediately after the 9/11 attack, the American Gas Association (AGA) decided to improve the security mechanism which can protect SCADA communication from malicious users. The primary purpose of the standard was to develop a security scheme which can provide security as well as save time and computation cost [73].

AGA-12 – "Cryptographic Protection for SCADA communications and general recommendations"

Traditional security protocols used in SCADA systems such as IEC 60870, DNP3, IEC 61850 and Modbus lack proper security services [42]. However, the new protocol AGA-12 provides security features to the SCADA systems. It uses cryptographic suites to secure the wireless communication between field devices and the MTUs [42][82]. The steps in AGA-12 is described in Table 3.3[73].

AGA-12 provides confidentiality, integrity and authentication. However, it fails to provide availability. It does not defend against DoS attacks. Furthermore, AGA-12 uses RSA as the key management protocol which has been cracked recently [95].

Furthermore, the current standards including IEC 62210, IEC 62351 and AGA-12 fail to provide two main security requirements, namely, defense against DoS attacks and a strong key exchange protocol.

The aforesaid studies have research gaps that fail to address availability and secured communication channel. Therefore, researchers have proposed schemes to overcome these limitations in SCADA networks.

In this thesis, the proposed schemes are categorized based on limitations addressed.

Table 3.3: Steps in AGA-12 standard

Steps	Sub – Step(s)	Description
Perform system security audit	• System-wide network audit must be done • Following the audit, risk assessment is required. • Security goals must be set.	During risk assessment, cost-benefit analysis is done. When benefits outweigh the cost, the AGA-12 is implemented in the SCADA network.
Agreement of Hardware and Software Modules to be used	• Guidelines are provided for testing of hardware and software modules. • The guidelines must also provide the cryptographic process agreement.	The algorithms which are accepted and permitted by National Institute of Standards and Technology (NIST), AGA 12 are as follows. • Advanced Encryption System (AES) Encryption with a key length of minimum 124 bits. • Rivest-Shamir-Adleman (RSA) with a key length of minimum 1024 • Elliptic Curve Digital Signature Algorithm (ECDSA) with a key length of minimum 160 bits. • Secure Hash Algorithm (SHA-1).
Performing a post-deployment security audit)	• Implement AGA-12 • Post Implementation audit	After implementation, it involves a detailed audit throughout the network. If any security threat is detected, the necessary compliance level should be approached.

- Detection of SCADA attacks: It involves the security schemes addressing the availability issue in the SCADA networks. Most of the schemes are based on machine learning algorithms.
- Prevention of SCADA attacks: The discussed schemes address the key exchange and management issue in SCADA networks.

3.2 Detection of SCADA attacks

Traditional standards and Intrusion Detection Systems (IDSs) such as firewalls used in SCADA are not strong enough to cope up with emerging attacks [66]. To increase the immunity in SCADA, machine learning algorithms, such as Naïve Bayes, Random Forest, C4.5 decision tree algorithm, Support Vector Machine, etc. are used to detect intrusion in the network[16][37].

3.2.1 Rule-Based Intrusion Detection System for SCADA networks

The proposed IDS uses a rule-based in-depth protocol analysis along with a Deep Packet Inspection (DPI) method. The model establishes a new set of intrusion recognition rules. The rule-based scheme contains two sub-schemes; namely, signature-based detection and model-based detection [97]. Signature-based detection utilizes a blacklist approach and is used for detecting a more significant amount of false spontaneous messages, unauthorized commands between nodes, and buffer-overflow. The model-based detection builds a model based on an in-depth analysis of the protocol. The created models portray the expected behavior of the protocol. It uses protocols and traffic pattern to generate the expected behavior[97]. It can detect known attacks as well as its source. Using the proposed IDS along with IEC/104 protocol, unknown attacks may be diagnosed in the SCADA network [97]. However, the proposed rule-based IDSs do not ensure the detection of novel or unidentified intrusions that pass through traditional IDS in open access networks.

3.2.2 Network Anomaly Detection for m-connected SCADA networks

Usually, IDSs and security schemes are for SCADA systems using open access networks. However, there is no intrusion detection mechanism for closed and isolated SCADA networks. This kind of SCADA architecture is referred to as an 'm-connected' SCADA network [52].

The model uses a dynamic detection for detecting intrusions with a packet logger and packet sniffer followed by a pattern matching algorithm. It generates new rules and stores them in a database. It further uses new rules for the next round[52]. The proposed scheme is based on rule-based intrusion detection and further research is needed for accurate implementation [52]. Furthermore, the scheme does not guarantee detection of unidentified attacks.

3.2.3 lp - norms in one-class classifications for intrusion detcetion in SCADA systems

In 2014, an intrusion detection system was proposed to detect abnormal activity in the network that is not detected by the traditional IDS or firewalls. It uses a machine learning based on the one-class classification algorithm for live detection of unnoticed cyberattacks [66].

The thesis analyses two approaches: the support vector data description (SVDD), and the kernel method[66]. It uses kernel principle as non-linear methods to detect patterns, and interdependencies within the real-world data. SVDD maps the data to the subspace which is optimized for one-class classification. The thesis concludes that the proposed method showed the highest error detection and the lowest false alarm rates after conducting tests on a real dataset with several cyber-attacks [66].

3.2.4 One-Class Support Vector Machine (OCSVM)

In 2014, Leandros et al. [62] developed a One-Class Support Vector Machine(OCSVM) model for detecting new attacks in the SCADA network. The proposed model addresses the following issues:

- The research community has developed many IDS algorithms for SCADA networks. Most of them are rule-based algorithms which make them incapable of detecting any new intrusions. In a real-time application, when any new anomaly is present, it fails to predict the behavior of the system [62].
- Other algorithms such as K-nearest neighbor (KNN), Hidden Markov models, and Support Vector Machines are used for detecting intrusion. However, they require learning of expected anomaly. Thus, these schemes may be sensitive to noise present in the training dataset [62].
- Negative selection algorithms can fail in the case of real-time application because of enormous diversity in real time data[62].

The proposed IDS is an algorithm to detect anomaly without any labeled data for training. Network traces train the OCSVM model without the use of open access networks. These features help the proposed IDS to perform in real time. Table 3.4 outlines the steps in the detection process[62].

However, the OCSVM model does not manage false positive results.

3.2.5 OCSVM model combines with k-means recursive clustering for intrusion detection in SCADA systems

One-class classifiers suffer from false positives and overfitting. False positive is a scenario when the IDS detects abnormal behavior but there is no intrusion in real. Overfitting is a case when the model begins to learn the details and errors in the training data. These two factors decline the performance of the model on the new data [63].

Table 3.4: Steps to detect intrusion using OCSVM

Step	Description
Step 1	Data analysis.
Step 2	Attributes in the network traces are extracted. The attributes, rate and packet size, are used to train the model.
Step 3	Integration of OCSVM Module.
Step 3.1	The network traces data, and the extracted attributes are used to train and generate the model.
Step 3.2	The model is tested for real-time anomaly detection.
Step 3.3	The detected anomalies are classified based on the severities.
Step 3.4	The main correlator is alarmed regarding the detected anomalies.

To address these two issues, Leandros et al.[63] developed an intrusion detection model to detect the malicious network traffic in SCADA. The model includes the One-Class Support Vector Machine (OCSVM) with Radial Basis Function (RBF) kernel and recursive k-means clustering [63]. OCSVM is an extension of support vector machines and is used to detect the outliers in the data. The k-means clustering algorithm is used to cluster the outliers and sort them with two clusters. OCSVM obtains two values, namely, maximum and minimum negative value [63]. The cluster which is near to minimum negative value represents severe alerts, and therefore, the cluster is used as input when there is recalling of k-means clustering. This step is repeated till the after-k means clustered are in a single cluster. After the completion of K-OCSVM phase, the model distributes the severe alerts among the nodes in the SCADA structure [63].

3.2.6 A Hybrid Model for anomaly-based intrusion detetion in SCADA networks

Usually, intrusion detection systems when deployed in real time lead to high computational and time costs. These two factors affect the performance of a SCADA

network[94].

In 2017, anomaly-based intrusion detection was developed using a feature selection model after removing redundant data. Irrelevant data can affect the efficiency of SCADA systems. This proposed scheme is time-saving, has low computational complexity and has 99.5% accuracy of detecting specific-attack labeled[83]. At first, the J48 classifier is used to train the dataset and then, to develop the model, Bayes Net classifier is utilized. The proposed model is tested on a database with three types of labeling as follows[83].

- Case 1: binary-labeled
- Case 2: categorized-labeled
- Case 3: specific attack labeled

The above-mentioned scheme focuses on the availability limitation in the SCADA networks. The schemes propose novel IDSs that detect any abnormal network behavior, which can lead to DoS attacks. However, the scheme fails to secure the communication channel. The following section on the prevention of SCADA attacks focuses on securing the communication channel with novel key exchange and management schemes in SCADA networks.

3.3 Prevention of SCADA attacks

The existing standards use vulnerable key management protocols that do provide a strong secure communication channel.

Encryption and key management are crucial in communication between nodes in a SCADA architecture. Key management schemes developed for SCADA can be categorized into two, namely, centralized key distribution and decentralized key distribution[77]. They can also be categorized into symmetric key cryptography, asymmetric key cryptography, and hybrid key cryptography[79]. In this thesis, another classification concerning self-healing property is added[37].

3.3.1 Symmetric Key Cryptography

Scada Key Establishment(SKE)

SKE categorizes SCADA communication into Controller -Subordinate (C-S) which uses symmetric key cryptography and Peer to Peer (P-P) which uses public key cryptography. The controller is the sub-MTU or sub-MSU, and the subordinate is the RTU. Peer-to-peer communication is between two sub-MTUSs or two RTUs[79].

For C-S communication, SKE uses four kinds of keys: Long-Term key, General Seed Key (GSK), General Key (GK) and Session Key (SK). The Long-Term Key (LTK) is manually distributed between the controller and subordinate [32]. The controller stores the GSK and is used by Cryptographic Authority (CA) to produce GK. By using two keys, GSK and LTK, the GK is generated and is then shared between the controller and the subordinate. While transmitting GK, it is encrypted by LTK. The session key is generated by using GK, sender's identity and TVP (Time-Varying Parameters). TVP field involves timestamp and a sequence number[77][32].

For peer-to-peer communication, SKE uses four different keys: Cryptographic Authority Public Key (CAPK), Public key Signature Key (PKSK), Common Key (CK) and Session Key (SK). The CAPK is shared among sub-MTUs while the PKSK is shared among the sub-MTUS, MTU and Cryptographic Authority (CA). The common key is generated by following a key exchange algorithm. The methodology to generate session key is the same as that of C-S communication. The session key is used to encrypt the messages transmitted[77][32].

However, the RTU to RTU communication is not directly allowed. Since the communications are treated differently in different conditions, it increases the overall overhead and complexity. Furthermore, the long-term keys are managed manually[77][32].

SCADA Key Management Architecture (SKMA)

In comparison to SKE, the implementation of SKMA name=SKMA,description=CADA Key Management Architecturearchitecture is more simplified. The architecture establishes the key exchange protocol among the Key Distribution Center (KDC), and any two nodes. The long-term keys are accumulated only on the required nodes and on the KDC which is the third party. The design uses three main keys[26][38]:

- Long-Term Node-KDC key is used to yield keys for communication and is manually shared between a node and the KDC.
- Long-Term Node- Node key is distributed between the nodes that require to communicate with each other.
- Session Key is used for encrypting the information transmitted from one node to another. Once the key establishment is completed, the session key is generated by using pseudo random number function, nonce-key and a time stamp[26].

The SKMA scheme does not use GSK. The key exchange in SKMA only happens when a new node joins the SCADA network[26].

Nevertheless, the SKMA does not provide the following security features[32].

- SCADA systems mostly use broadcast communication. However, the SKMA cannot provide such a mechanism.
- This protocol does not provide any confidentiality and integrity.

Logical Key Hierarchy (LKH)

To address one of the issues, the LKH protocol was developed. LKH protocol provides secure broadcast communication [77][79]. It is based on an architecture of the logical tree of keys [26]. It maps all the nodes of the SCADA network as the leaves of a

structure tree. Each node stocks the entire symmetric keys from the root to its leaf. When a node leaves or joins, the node keys from its leaf to the root is updated so that the security of the network is preserved[26]. For example, Figure 3.1[26] explains the mechanism when a node joins the network.

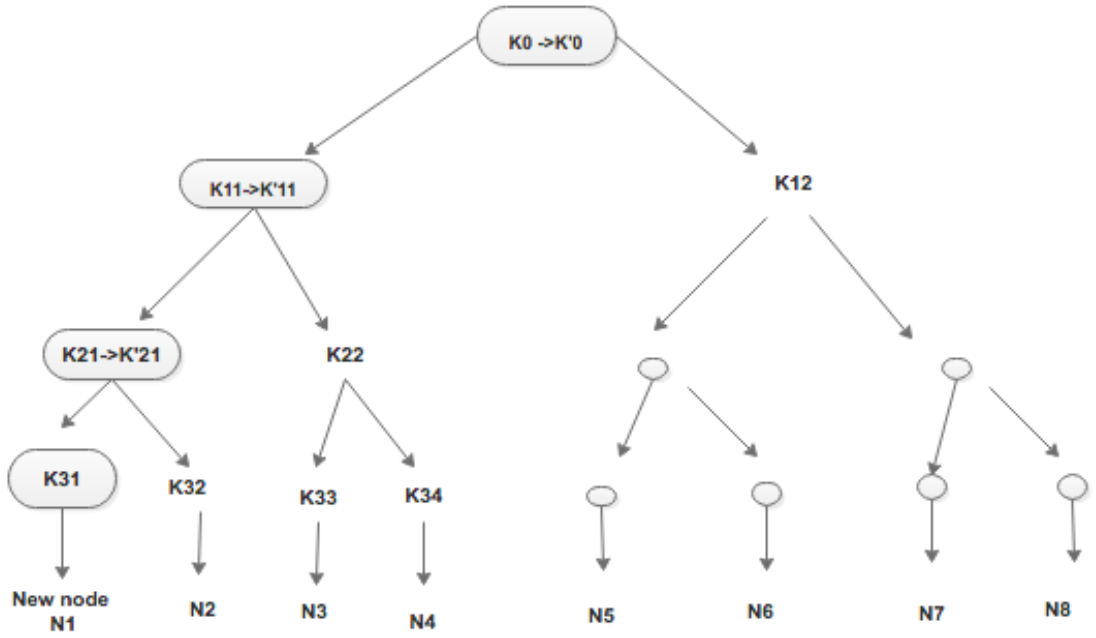

Figure 3.1: Update Mechanism of LKH protocol when a new node joins

Adavanced Key-Management Architecture(ASKMA)

To enhance the efficiency of SKMA and LKH, the ASKMA was proposed [77]. It provides both message broadcasting and secure communication. It also keeps a minimal load on the resource-constrained nodes [77][26].

In ASKMA , the LKH protocol is used by Choi et al. [26] for message broadcasting in 2009. The nodes of the SCADA networks such as RTUs, sub-MTUs, and the MTU are organized in two tree structure: binary tree and n-ary tree. The MTU to sub-MTU follows a binary tree structure whereas the sub-MTUs to RTUs follows n-ary tree structure [77].

The ASKMA protocol evenly spreads the computations to the sub-MTUs

and MTUs which are high power nodes and keeps a minimal load on the low power nodes like RTUs. Therefore, the nodes are arranged logically in a tree structure, n-ary or binary tree, depending on their computational power[26].

When a new node is added to the SCADA network, the ASKMA follows a Join Protocol. Any key received by a new RTU must be independent of any existing keys in the nodes of the tree. It preserves backward confidentiality. When a new node joins the tree, the KDC updates all the keys from its leaf to the root on the freshly joined RTU's path. It uses a hash function for renewing the keys. The Join Protocol has the following steps[26].

Step 1: The KDC renews all Ki,j to K/i,j where K/i,j = H(Ki,j). Step 2: In case the RTUs have keys belonging to Ki,j, each RTU updates their key Ki,j to K/i,j. Step 3: With Km, the KDC encrypts all K/i,j and transmits the encrypted information to the newly joined RTU which is Nm.

When a node leaves the SCADA network, the ASKMA follows a Leave Protocol. Similar to the Join Protocol, all the keys throughout the key path updated with new keys [12]. However, the leaving node Nm should not be able to use the updated keys. This makes the Leave Protocol a little more complicated than Join Protocol. The following are the steps of Leave Protocol[26].

Step 1: The KDC removes the RTU which is parting. Step 2: It then updates the remaining keys by executing a key generation algorithm such that the leaving RTU does not know the updated key. Consequently, the departed RTU is unable to compute the new keys. Step 3: Each RTU updates its keys by using the hash function. Step 4: If the RTUs are unaware of their sibling keys, KDC encrypts the new keys and sends them to those RTUs. Step 5: The departed node knows all the ancestor keys of the sibling RTUs. Therefore, the KDC encrypts all the updated keys with sub-MTU's private key and transmits to the sub-MTU. The sub-MTU encrypts the received keys with the child RTU's key and then sends it to each child RTU.

ASKMA supports broadcast and multicast communication. However, it does not offer efficient multicast communication. To solve this issue, ASKMA+ was

proposed[77]. By reducing the number of stored keys, it provides efficient multicast and broadcast mechanism [77]. However, ASKMA and ASKMA+ do not address the availability issue in SCADA[77].

3.3.2 Hybrid Key Cryptography

Hybrid Key Management Architecture(HKMA)

To satisfy the availability requirement, Choi et al.[25] proposed a Hybrid Key Management Architecture (HKMA) which supports a replace scheme [25]. The scheme includes an operation of the replace protocol in case of compromised or broken main device. It uses a public key cryptosystem in MTU to sub-MTU communication which has high performance, and symmetric key cryptosystem in sub-MTU to RTU which has low performance. Thus, it reduces the number of keys to be stored in the MTU [25].

Adavnce Hybrid Key Management Architecture(AHSKMA)

Rezai et al. [77][76] proposed a scheme based on hybrid key management architecture to tackle the availability issue in SCADA networks and to increase the performance and security of HKMA. It follows ECC for MTU to sub-MTU communication. Since RTUs have limited computational resources, symmetric cryptography is used for sub-MTU to its RTUs communication. This scheme makes the architecture suitable for the environments with resource constrained devices and supports unicast, multicast and broadcast communications[76]. Figure 4 shows the mechanism of the protocol.

The Iolus Framework[65] is used while connecting the MTU and RTUs. The MTUs act as the Group Security Control (GSC) and the sub-MTUs act as the group security intermediary (GSI). The architecture consists of four phases: Setup phase, Join Phase, Leave Phase and Replace phase[65][76].

- Setup Phase: In the first phase, the group key is generated by the MTU and is

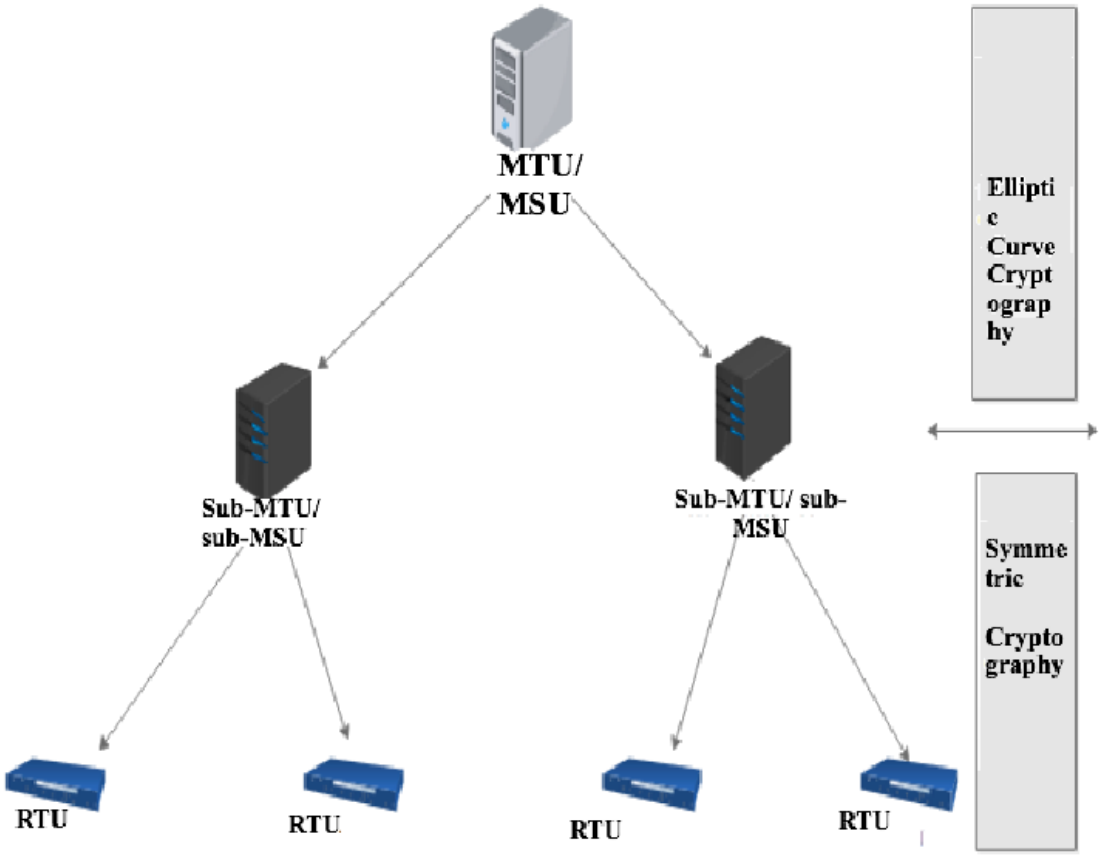

Figure 3.2: Mechanism of AHSKMA.

shared with RTUs and IEDs.

- Join Phase: Similar to LKH and AHSKMA, the MTU updates all the keys of the remaining nodes in the SCADA network as soon as a new node joins.
- Leave Phase: This phase is also similar to the leave protocol of the AHSKMA.
- Replace Phase: In case the MTU is damaged, it is replaced by its backup device. Each MTU and sub-MTU has a backup device. While backing up the broken device, the Join phase and the Leave phase are performed concurrently.

The Replace Phase resolves the availability issue in SCADA networks. In this scheme, the session is produced using a hash function, a key, and TVP with a sequence number and timestamp [76]. So, HSKMA also guarantees the freshness of key along with availability.

Both HKMA and AHSKMA provides replace scheme to satisfy the availability requirement, but the affected devices stop working during the replacement. To solve this issue, LiSH+ was proposed[77][45].

3.3.3 Self-Healing Key Distribution

Lmited Self-Healing Key Distribution(LiSH+)

LiSH+ is an efficient group key management scheme which utilizes a self-healing procedure having collusion resistance capability and effective revocation[45]. The scheme involves five phases: initialization, rekeying, self-healing mechanism, the addition of new nodes, and reinitialization. It uses a bivariate polynomial to lower the storage burden from RTUs[45]. It also uses intrusion detection system to detect compromised and eliminate users. These features provided helps LiSH+ to enhance the security of SCADA networks[45].

However, the LiSH+ focuses on only two requirements: availability, and efficiency[45]. It does not focus on the authentication mechanism.

3.3.4 Asymmetric Key Cryptography

ID-based Key Management Architecture

Lim[57] proposes an ID-based key management architecture (ID-KMA) based on pairing algorithm based on elliptic curves. The architecture addresses the issues of the public key cryptography with a digital signature. It involves fast and efficient session key establishment along with session key recovery protocol. It removes the concept of the digital certificate which minimizes the overhead.

The architecture involves the role of three units of SCADA: Key Management System (KMS), MTU and RTUs. The KMS is linked with the MTU, and the MTU is connected to the RTUs. The KMS communicates with RTUs through

MTU[57].

The ID-based Key Management architecture uses four main keys[57] as described in Figure 3.3.

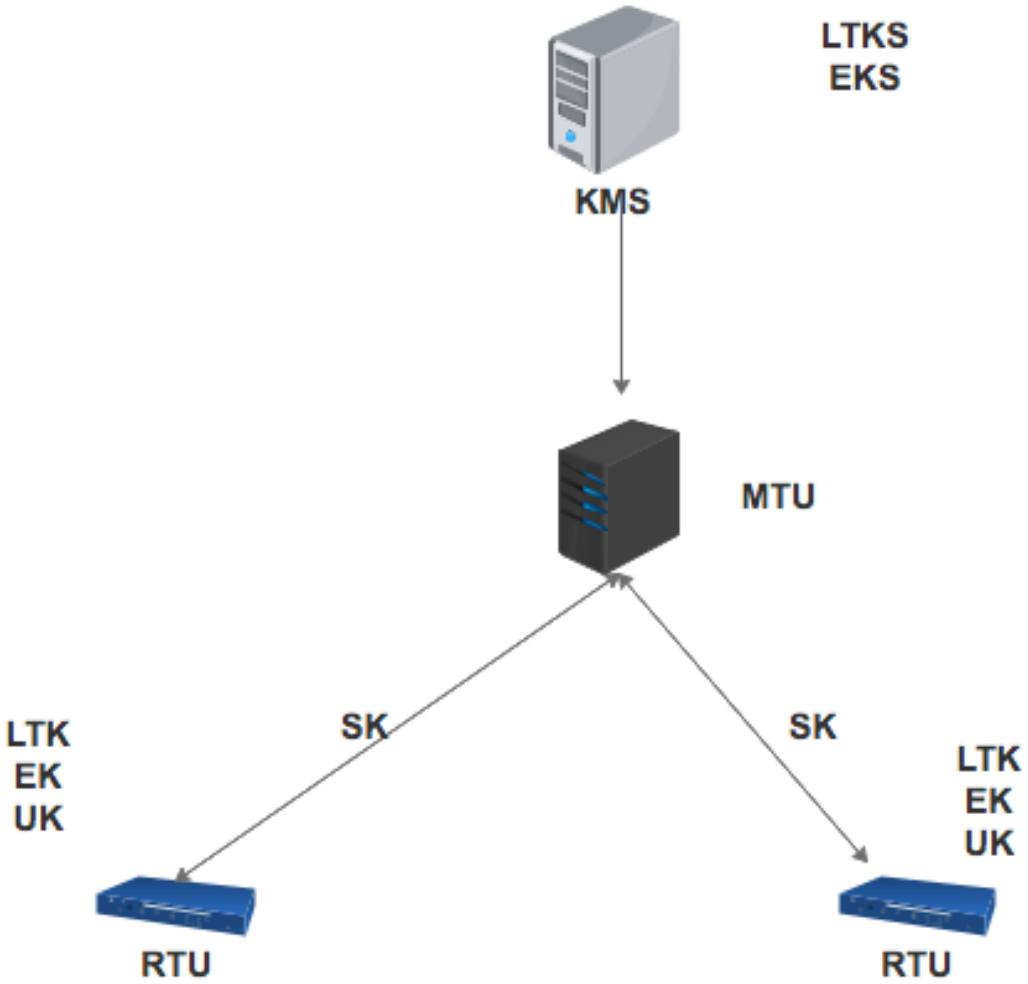

Figure 3.3: Architecture of ID-KMA

The ID-based key management protocol is composed of four phases[57].

- EK setup: The EK is stored in each component of the architecture in advance.
- Initialization: The initialization has two stages. In the first stage, the KMS produces system parameters (SP) which are public and generates MTU's and RTU's LTK. The SK and LTK are encrypted with EK. The KMS shares the encrypted SP and LTK with the MTU. In the second stage, the KMS distributes

a UK and LTK to each component so that the MTU can share an SK with RTUs. The first stage is LTK distribution and the second stage is the UK distribution.

- RTU-RTU session key establishment: This phase focuses on the secure communication between the RTUs with the usage of session key (SK) and initially shared update key (UK).
- MTU-RTU session key establishment: The session key is distributed among MTU and RTU to have a secure communication. The MTU sends a session key request to the RTU.
- RTU-MTU session key establishment: Similarly, the session key is established between MTU and RTU. The RTU sends a session key request to the MTU.

All the afore-mentioned key management protocols are based on traditional cryptography schemes which are vulnerable to quantum attacks[21]. Furthermore, public key algorithms tend to increase the computational and time cost[74].

Therefore, the following scheme named as Nth Degree Truncated Polynomial Ring (NTRU) is proposed to defend against quantum attacks.

NTRU cryptographic Algorithm for SCADA networks

The key management scheme is based on a faster and light-weight public key algorithm named NTRU cryptography[74]. The cryptographic algorithms in IEC62351 and AGA-12 have performance issues when apples to SCADA network security. They are time and power consuming[74][14].

Due to various security and performance complexities of SCADA systems [77][14][29], NTRU was developed. It is a public key scheme based on lattice-based cryptography[39][3]. The security of the cryptography depends on a hard problem known as Short Vector Problem [74][39]. The encryption and decryption use polynomial operations which makes the system faster[74]. Therefore, it has better processing speed than traditional schemes and is suitable for real-time requirements of SCADA security[23].

The NTRU algorithm is also known as post-quantum cryptography and has been resistant to quantum attacks [74]. The scheme has two sub-algorithms, namely, NTRU Encrypt which is used for encryption, and NTRU Sign which is used for generating a digital signature. The scheme comprises of three phases [74]:

- Key Generation and Certificate Creation phase: In this phase, public and the private key of the RTU and its digital certificate is generated. For this, it uses a public key infrastructure. In this phase, other than RTU, two components play their roles. One, Local Key Store (LKS) and another, Certificate Authority (CA). The phase has the following steps[74].

  Step 1: The RTU generates a public key and private key using key generation algorithm. The algorithm uses algebraic structures of certain polynomial rings and is based on the Short Vector Problem. It then stores the generated key pair in the LKS.

  Step 2: The RTU then sends a request containing its public key to CA for generating a digital certificate. The CA in return creates a digital certificate and directs it to the RTU.

- NTRU Encryption: In this phase, the RTU uses the receiver's public key, generated by the CA, to encrypt the messages. The messages are converted to a ring of truncated polynomials modulo. The receiver then decrypts the cipher using its private key.

- NTRU based authentication: In this phase, it is verified that the encoded message, which is in the state of a truncated polynomial ring, is validated. This phase uses a procedure built on a non-keyed hash function to ensure the integrity and authenticity of the message. The scheme creates a message digest by using the hash function. The message digest is then digitally signed by using the RTU's private key. Thus, it generates the digital signature. Therefore, the RTU sends the encrypted message and its digital signature to the receiver. The receiver uses its own NTRU private key to decode the message and generates the message digest (MD1) following the same procedure. The digital signature is then decrypted using the RTU's public key. The receiver gets the expected

message digest (MD2). It then verifies whether MD1 and MD2 are equal or not. If they match, the signature is verified[74].

Even though NTRU is not yet vulnerable to quantum threats, a quantum computer can crack the algorithm using brute-force[4]. There are further challenges in post-quantum cryptography as follows[18].

- Need to improve the efficiency of the algorithm.
- Need to build confidence in the scheme.
- Need to improve the usability of the algorithm.

The existing standards have research gaps that have been addressed by the above-discussed security schemes. However, all the schemes are based on arithmetic operations. The emergence of quantum computers is proven to be beneficial as well as precarious to the cyber world. By launching a brute-force attack using Shor's or Grover's algorithms, these schemes can be broken. Therefore, there is a research gap in securing the SCADA networks from quantum attacks.

3.4 Comparative study of current security schemes for SCADA

3.4.1 Primary Factors Used For Comparative Study

The comparative analysis in this thesis is based on the primary factors in each category. In case of current standards, the current standards are categorized into two classes as shown in Table 3.5. In this scenario, the primary factors used for comparison are as follows:

- Information Security Policy is a set of security rules governed by an industry that is imposed on the users of its system[53].
- Vulnerability and risk assessment are the processes where the weaknesses in a system are detected, analyzed and prioritized by the organization. The analyzed results are used to recommend security requirements in the system[47].

- Information security infrastructure is a set of security rules to protect only critical architecture such as airports, nuclear power plants and traffic control systems. It is similar to the Information Security Policy[53]].

- Third Party access or Outsourcing is giving access to service providers, vendors and contractors that can lead to credential theft and data risk management. To overcome these security concerns, the organizations extend the security policy. For example, the third party can be given access to a separate domain from the internal network, by using firewalls [15].

Table 3.5: Classification of current standards

Guideline based Standards	Crypto-suites based standards
IEEE 1402	IEC 62210
ISO 17799	IEC 62210IEC 62351
ISO 15408	AGA-12
NERC Security Guidelines	
NERC 1200	
NERC 1300	
API 1164	

Furthermore, the crypto-suites standards are compared based on the following factors.

- Presence of Key Management Protocol in the standard and the strength of the protocol.
- Presence of Strong Encryption and strength of the encryption algorithm used in the standard.
- Sustaining security requirements refers to the existence of confidentiality, integrity and availability in the security scheme of the standard.

The strength of the key management and encryption scheme depends on the resources and time utilized to crack the scheme.

In case of detection of SCADA attacks, the primary factors are as follows:

- Known Attack Detection is the scenario where any traffic is categorized as an attack if the features of that particular traffic fall under the domain of attacks stored in the IDS database.
- New Attack Detection is the scenario where any traffic with unique behavior will be detected.
- Open Access networks are the public networks where the connected devices are exposed to each other. The public networks are vulnerable to various cyber threats. The private networks that provides access to the legitimate user.
- False positive is the situation where the IDS can detect the false alarms. False positives are the consequences where an activity is classified as abnormal even if its behavior is normal.

In case of prevention of SCADA attacks, the primary factors used for critical analysis are as following:

- The efficiency of the encryption scheme depends on the amount of computation resources utilized by the algorithm. Therefore, an algorithm with high overhead or cost is less efficient and vice versa.
- Confidentiality is the secured privacy of the data.
- Integrity is when the data remains intact and unmodified.
- Authentication is a security property focusing on verifying and validating the identity of the user in the network.
- Availability is the scenario where the server is always accessible to the client.
- Non-repudiation is when the sender cannot deny that the data has not been sent by him at a particular time.
- Broadcast communication is the one-to-many communication case in a network.
- Self-healing is the case the users of an attacked network can recover their lost session keys to secure the data communication.
- Vulnerability to quantum attack refers to the absence of security measures to protect a system from quantum attack.

3.4.2 Comparison of various security schemes

We now present a critical analysis of the schemes developed for SCADA network security. The schemes are classified into three categories: current standards, detection, and prevention of SCADA attacks. The thesis analyzes the schemes in each category. Moreover, all the schemes are compared with each other. The tables below show the comparison between the protocols.

Table 3.6 shows that AGA-12 is the best among all the standards providing cryptographic protection to the SCADA systems. However, AES relies on ECDSA, AES, RSA, and SHA which leads to high computational and time cost. It also does not involve an intrusion detection system and a strong key management protocol.

Table 3.7 provides the comparison of all the crypto-suite based standards and AGA-12 is by far the best standard. However, unlike IEC 62351, it does not provide defense against DoS attacks. Thus, the scheme has lack of availability property.

In all the standards, the key management protocols and encryption scheme used are weak and vulnerable to quantum attacks.

Table 3.8 compares all the intrusion detection system proposed for SCADA network security. In this category, OCSVM with K-means emerged as the best detection scheme for SCADA systems using open access networks. However, it is unclear whether it is efficient when used for closed access networks.

Table 3.9 compares all the proposed key management protocols for SCADA networks. NTRU is the best scheme among the proposed schemes. It satisfies the main security requirements: confidentiality, integrity, and authentication. The scheme is not yet vulnerable to attacks from quantum computers. However, a quantum computer may able to crack the NTRU algorithm in the future.

Table 3.6: Comparative analysis of current standards used in SCADA systems.

CURRENT STANDARDS			Organizational Security	
Standards	Information security policy	Vulnerability and risk assessment	Information security Infrastructure	Security of third-party access
AGA 12	Yes	Yes	Yes	Yes
API 1164	Yes	No	No	Yes
ISO 17799	Yes	No	Yes	Yes
NERC Security Guideline	Yes	Yes	Yes	Yes
NERC 1200	Yes	No	Yes	No
NERC 1300	Yes	Yes	Yes	Yes
IEC 62210	Yes	No	Yes	Yes
IEC 62351	Yes	No	No	No
IEEE 1402	Yes	No	Yes	No
ISO 15408	Yes	Yes	Yes	No

Table 3.7: Comparative analysis of crypto-suite based SCADA standards.

Crypto-suite based Standards	Presence of Key Management scheme	Strength of Encryption	Sustaining Security Requirements		
	Strong/Weak	Strong/Weak	Confidentiality	Integrity	Availability
IEC 62210	No	Yes, weak	Yes	Yes	No
IEC 62351	Yes, weaker than AGA-12	Yes, weaker than AGA-12	Yes	Yes	Yes
AGA-12	Yes, weak	Yes, weak	Yes	Yes	No

Table 3.8: Comparative analysis of detection schemes of SCADA attacks.

Detection of SCADA attacks	Known Attack Detection	New Attack Detection	For Open access networks	Distinguish false positives
Rule-based	Yes	No	Yes	No
IDS for m-connected SCADA networks	Yes	No	No	No
lp – norms in One-Class Classification	Yes	Yes	Yes	No
OCSVM	Yes	Yes	Yes	No
OCSVM with K-means	Yes	Yes	Yes	Yes
Hybrid model	Yes	No	Yes	No

Table 3.9: Comparative analysis of prevention schemes of SCADA attacks.

Prevent SCADA attack	Cost	Confidentiality	Integrity	Non-repudiation	Availability	Authenticate	Broadcast Interaction	Self-Heal	Prone to QC attack
SKE	High	Yes	No	No	No	No	No	No	Yes
SKMA	Low	Yes	No	No	No	No	No	No	Yes
LKH	High	Yes	No	No	No	No	Yes	No	Yes
ASKMA	Low	Yes	No	No	No	No	Yes	No	Yes
HKMA	Low	Yes	No	No	Yes	No	Yes	No	Yes
AHSKMA	Low	Yes	No	No	Yes	No	Yes	No	Yes
LiSH+	Low	Yes	No	No	Yes	No	Yes	Yes	Yes
ID-based KMP	Low	Yes	No	Yes	No	Yes	Yes	No	Yes
NTRU	Low	Yes	Yes	Yes	No	Yes	Yes	No	No

Chapter 4

Quantum Attacks on SCADA systems

4.0.1 Quantum Computer

Traditional computers are the digital electronic computers which encode information in bits, where each bit can be 0 or 1. They execute algorithms on bits using simple digital logic operations such as AND, OR, and NOT[24]. Instead, quantum computers encode information in qubits which are generated using atoms as digital bits[12]. The value of qubits is based on the rules of modern physics: superposition and entanglement principle. According to the superposition principle, each qubit can represent 0 or 1 or both at the same time. Entanglement occurs when two superposed qubits are allied with each other [98][12]. Therefore, the number of qubits is directly proportional to the number of states held by the set of qubits[98][81]. These two principles make quantum computing way faster than traditional computing[37].

A quantum algorithm was proposed to solve a binary maze problem [54]. Each line has one input and two outputs. The quantum algorithm attempted all the paths at the same time, and therefore, it solved the problem at extreme speed. Whereas, solving the maze problem was hard for a traditional computer since the size of the problem was doubling each time. For example, a 1000 step binary maze has 2 1000 outcomes, and this took more time in the case of traditional approach [54].

D-wave, a quantum computing company, launched its first commercial quantum computer named D-Wave One in 2011, which is being used by National Aeronautics and Space Administration (NASA) for in-depth space exploration. By 2013, they increased the number of qubits and released the D-Wave Two system. Google is also planning to use a quantum computer for big data mining [98].

4.0.2 Man-in-the-Middle attack by a Quantum Computer

The SCADA uses a hierarchical infrastructure. The RTUs and MTUs communicate with each other continuously and in tight timing. A quantum computer can intercept the communication channel between two units and fetch the information without being detected. Even if the fetched data is in encrypted form, it can easily crack the cipher by brute force[21].

4.0.3 Brute force attack by a Quantum Computer

The capacity and speed of quantum computer for solving mathematical problems make them a threat to traditional security schemes. All the encryption schemes are derived from mathematical logic. Cracking these schemes may be possible for quantum computers [21][34]. One such problem is Elliptic curve cryptography (ECC or ECDSA). Using Shor's algorithm, a quantum computer can launch a brute force attack and crack ECC in a brief time[34].

Section 1: CLASSICAL PART

Step 1: Select a random positive integer m such that m<n.Then, calculate gcd(m,b) using the Euclidean algorithm. If gcd is not equal to 1, a non-trivial factor is obtained. Thus, the algorithm ends. Otherwise, go to Step 2.

Section 2: QUANTUM PART

Step 2: Calculate the period P of the sequence:

$$xmodn, x^2modn, x^3modn, ..t... \tag{4.1}$$

Step 3: If p is odd, return to step 1. If p is even, go to step 4.

Step 4:

$$m^{p/2} - 1 = m^p - 1 = 0mod(n), \tag{4.2}$$

since p is even. If

$$m^{p/2} + 1 = 0 mod(n), \tag{4.3}$$

then return to step 1. Else, go to step 5.

Step 5: Calculate

$$result = gcd(m^{p/2} - 1, n) \tag{4.4}$$

using the Euclidean algorithm.

Shor's algorithm is a quantum algorithm for factorizing a number [75]. It implies that any public key cryptography can be easily cracked. The algorithm has two sections as follows [19]. The steps are explained as following.

- The classical computer can compute Section 1. It reduces the factoring problem to an order finding problem using the Euclidean algorithm. The Euclidean algorithm is a fast scheme to calculate the greatest common divisor (gcd) of two integers[61].
- Section 2 is the quantum part which used order finding algorithm. It finds the period of the function using the Quantum Fourier Transform (QFT).

In step 2, to calculate the period of the function based on the series, Quantum Fourier Transform (QFT) is used. Using QFT increases the speed of the algorithm by evaluating the function at all points simultaneously[19]. The QFT is a linear operator when applied to any state of qubit transforms it into another state. In other words, it is applied to the vector of amplitudes of a quantum state[58]. For example, if QFT operates on a quantum state X, then it transforms it into a quantum state Y as shown in the following equations.

$$X : |x >= \sum_{i=0}^{N-1} x_i |i > \tag{4.5}$$

$$Y : |y >= \sum_{i=0}^{N-1} y_i |i > \tag{4.6}$$

The QFT refers to the equation below.

$$y_k = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} x_j \omega_n^{jk}, k = 0, 1, 2, 3...., N-1 \tag{4.7}$$

where,

$$\omega_n = e^{2\pi i/N} \tag{4.8}$$

And, is a primary Nth root of unity, N is the length of vectors such that $N := 2^n$ [58].

Existing security standards and schemes are based on traditional cryptography such as Advanced Encryption System (AES), Elliptic-curve cryptography (ECC), and Secure Hash Algorithm (SHA). Therefore, they are vulnerable to quantum attacks. The transformation of quantum computing from theory to practice in the recent past has not only brought with its potential advantages but also increasing threats[21][34].

Chapter 5

Proposed Security Scheme

Based on the research gap in existing security schemes, we infer that they practice key management and authentication protocol, which is weak against the quantum algorithms. Accordingly, we propose a new scheme to guard the communication channel between RTU and MTU from a quantum along with classical attack. Moreover, Fröhlich et al. [36] demonstrated that BB84 protocol can be conducted to 200 km with multiplexing.

In our proposed scheme, we assume the following factors:

- MTU has the identities and hashed IDs of all RTUs.
- The id of MTU is embedded in each and every RTU.
- The RTU and MTU is aware of hash functions used to generate the private key.
- The data stored in the legitimate units are secure.
- We assume that our proposed scheme is conducted 200 km.

Sibson et al[89]. have developed a chip-based QKD in 2015. This evolution of QKD has motivated us to propose a quantum-based signcryption scheme for SCADA networks since they can be deployed in RTUs as well. However, both RTU and MTU need a few hardware changes. There is a need to integrate a monolithically integrated transmitter and a receiver with a photonic circuit using thermo-optic phase shifters in the RTU as well as in the MTU.

Quantum Cryptography is based on Heisenberg's Uncertainty Principle and Principle of Polarization of Photons[90]. Furthermore, the No Cloning Theorem

makes quantum cryptography a feasible scheme to resist the threats of both quantum and traditional computer. The most popular protocol of Quantum Key Distribution (QKD) is BB84 protocol and is the most suitable for IoT applications[81]. The proposed scheme has three main phases: Phase A: Quantum Key Distribution Phase B: Signcryption Phase C: Un-Signcryption

5.1 Quantum Key Distribution: Identification of and Defense against quantum attack

This phase uses BB84 protocol to generate a final quantum key[17]. The final quantum which is generated is used for signcryption. The BB84 protocol uses two Basis: Horizontal-vertical linear and Diagonal directions. Since, the key generation is based on the polarization of light, each photon is polarized using one of the two Basis randomly. The protocol uses two channels: Quantum Channel, which is used for key generation and distribution, and Classical Channel, which is used for information transmission and eavesdrop detection. This phase has further steps: Phase 1: Quantum Key Generation Phase 2: Key Sifting Phase 3: Error Correction Phase 4: Privacy Amplification

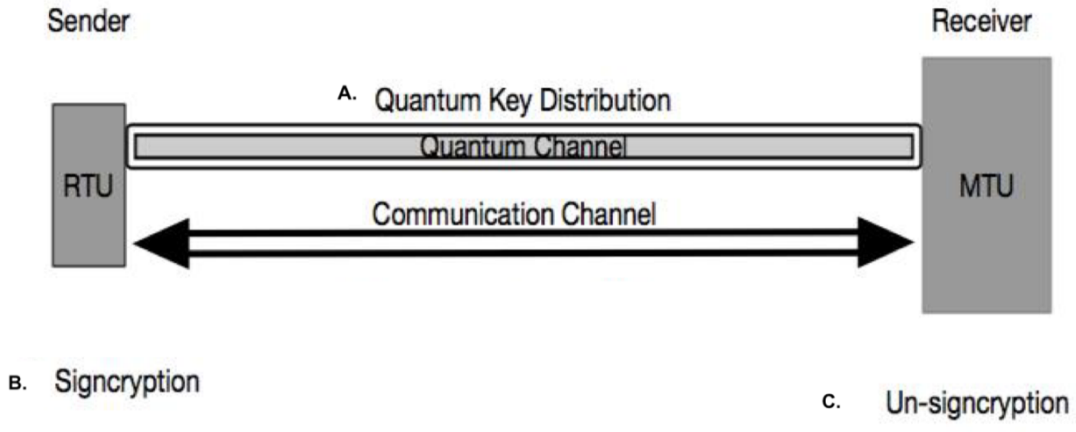

Figure 5.1: The Proposed Scheme Model

5.1.1 Quantum Key Generation

The sender generates the first qubits by randomly using one of the basis and sends it to the receiver via the quantum channel. For example, RTU acts as the sender and MTU acts as the receiver, as shown in Figure 5.1. The series of first qubits is called raw bits[81][98].

The MTU reads each qubit with either of the two bases randomly and independently. The series of qubits received by the MTU is called the raw key. There are two cases of measuring the raw bits as following.

- *Case1:* The receiver has a 50% success rate of choosing the right basis to measure the bits and thus getting the correct bits.
- *Case 2:* The receiver has a 50% failure rate where it selects the wrong basis. However, the outcome of using the wrong machine is random that is either 0 or 1. Thus, the probability of incorrect bits in the received bits is 25%, and that of correct bits is 75%. This ratio persists in the absence of any eavesdropper[81][98].

The qubits, when measured using any basis, their state changes randomly. Also, the states of the qubits cannot be cloned, which helps in the detection of an eavesdropper. When an eavesdropper tries to read the qubits in the quantum channel, it disrupts the state of the qubits. Thus, MTU receives the disrupted qubits. The MTU measures the tampered raw key, and the rate of incorrect qubits exceeds 25%[81][98].

5.1.2 Key Sifting

The MTU sends the randomly chosen basis to the RTU via the public or communication channel. The RTU verifies its chosen basis with that of MTU's. Then, the RTU sends the incorrect basis to the MTU. Both the units discard the bits measured by the incorrect basis and obtains the sifted key. In case of no noise in the quantum channel, the sifted key of both the units is the same. In case of any presence of noise, there is an error in the sifted key deduced by MTU[98].

5.1.3 Error Correction Protocol

In this phase, it has the following sub-steps[81][98].

Step 1: Determine Quantum Bit Error Rate (QBER).
The QBER is calculated. The RTU and MTU randomly extracts a part of its sifted key to RTU. The MTU discloses its extracted part to RTU. The RTU obtains the QBER by calculating the ratio of MTU's extracted key and its own extracted key. Both of the unit discards the exposed part and obtains the sub-sifted key.

- *Case 1:* If QBER is greater than 25%, the sifted key is discarded, and the raw key is again generated.
- *Case 2:* If QBER is less than 25%, error correction protocol and privacy amplification is followed.

Step 2: Error Correction Protocol (ECP) used: Reed Solomon Code.
In this phase, the sender and the receiver resolve the error in the sub-sifted key via the public channel. The error correction protocol phase is crucial after quantum key exchange for the following reasons.

- It gives both the units to check the confidentiality and integrity of the obtained sub-sifted key.

- The RTU sends its sub-sifted key encoding it with ECP protocol to MTU. The encoded key is called codeword. The encoding involves adding extra bits or parity bits to the original data. This helps the receiver to detect and resolve the errors. Therefore, the eavesdropper is unable to read the original key. The eavesdropper if modifies the codeword, it can be detected as well as corrected by the MTU.

- In this phase, based on the QBER, the sub-sifted key is polished as the errors are reconciled.

  There are various error correcting codes proposed by researches throughout the world. The most common are the Cascade protocol, Winnow protocol, Low

Density Parity Check (LDPC) protocol, Low Complexity Parity Check (LCPC) and, Reed Solomon protocol(R-S) [46][11][27].

The most feasible protocol for wireless networks is LCPC and Reed Solomon protocol. The purpose of Low Complexity and Parity Check is to detect and correct single- and double-bit errors. However, during the quantum key exchange, the errors can occur in a burst. In that scenario, the Reed-Solomon code is the most suitable protocol in the BB84 protocol. R-S code is an efficient algebraic code which can correct a large number of errors with low overhead and low complexity. It has the power to correct errors in a cluster. Various storage systems, broadcast systems, and wireless networks widely adopts R-S code.

Characteristic of Reed-Solomon(R-S) code: It is a subgroup of Bose-Chaudhuri-Hocquenghem(BCH) codes and linear codes which performs their arithmetic operations in a Galois field or finite field. BCH is cyclic error-correcting codes that involve using polynomials over data blocks. The code word generated in this algorithm consists of polynomials, which is divisible by another fixed short-length polynomial. The fixed polynomial is called Generator Polynomial[78].

A Reed-Solomon code is represented as R-S(n.k) with s-bit symbols. It implies that the encoder takes k data symbols with s bits each. Then, it adds parity symbols. Thus, obtaining a code word of n symbol. The parity symbols of s bits each are n-k. The R-S decoder can resolve up to t symbol errors in a codeword. It implies that it can automatically correct errors up to t bytes. The length of parity is calculated as following[27][78].

$$2t = n - k \tag{5.1}$$

The maximum codeword length (n) can be calculated as following

$$n = 2^s – 1 \tag{5.2}$$

R-S Encoder: In R-S encoding, the sub-sifted key is the message which is represented as a polynomial i(x). The polynomial is multiplied with the Generator

polynomial g(x)[27][78].

$$c(x) = g(x).i(x) \tag{5.3}$$

where,c(x) is the valid codeword.
i(x) is the information block.
g(x) is the generator polynomial.

Using Lagrange Interpolation, the polynomial is evaluated.

$$p(x) = i(x).x^{n-k} \mod g(x) \tag{5.4}$$

R-S Decoder: The Decoding algorithm follows algebraic procedure to correct up to t errors or up to 2t erasures. An error occurs when an incorrect bit is present in the codeword. An erasure occurs when the position of the incorrect bit is known[27][78].

The received codeword can be represented as following.

$$r(x) = c(x) + e(x) \tag{5.5}$$

where r(x) is the received codeword, c(x) is the recovered codeword and, e(x) is the error pattern present in the r(x).

The decoder follows the succeeding steps.

- Syndrome Calculator: It calculates the syndrome which is used to identify the symbol errors. One symbol error occurs when either 1 bit is incorrect, or all the bits are incorrect in a symbol.
- Error Locator: It then finds the symbol error locations by calculating the error locator polynomial. It uses Euclid's algorithm.
- Calculate Magnitude of error: Then, it finds the roots of the error locator polynomial.

- Error evaluation: To calculate the symbol error values, Forney algorithm is used.

Finally, a recovered codeword is received.

5.1.4 Privacy Amplification

From the received and recovered codeword, the sub-sifted key is extracted. To reduce any information leakage during error correction protocol and to increase the secrecy of the key, the sub-sifted key is hashed. Both MTU and RTU obtains the finalized key or Quantum Key (QK).

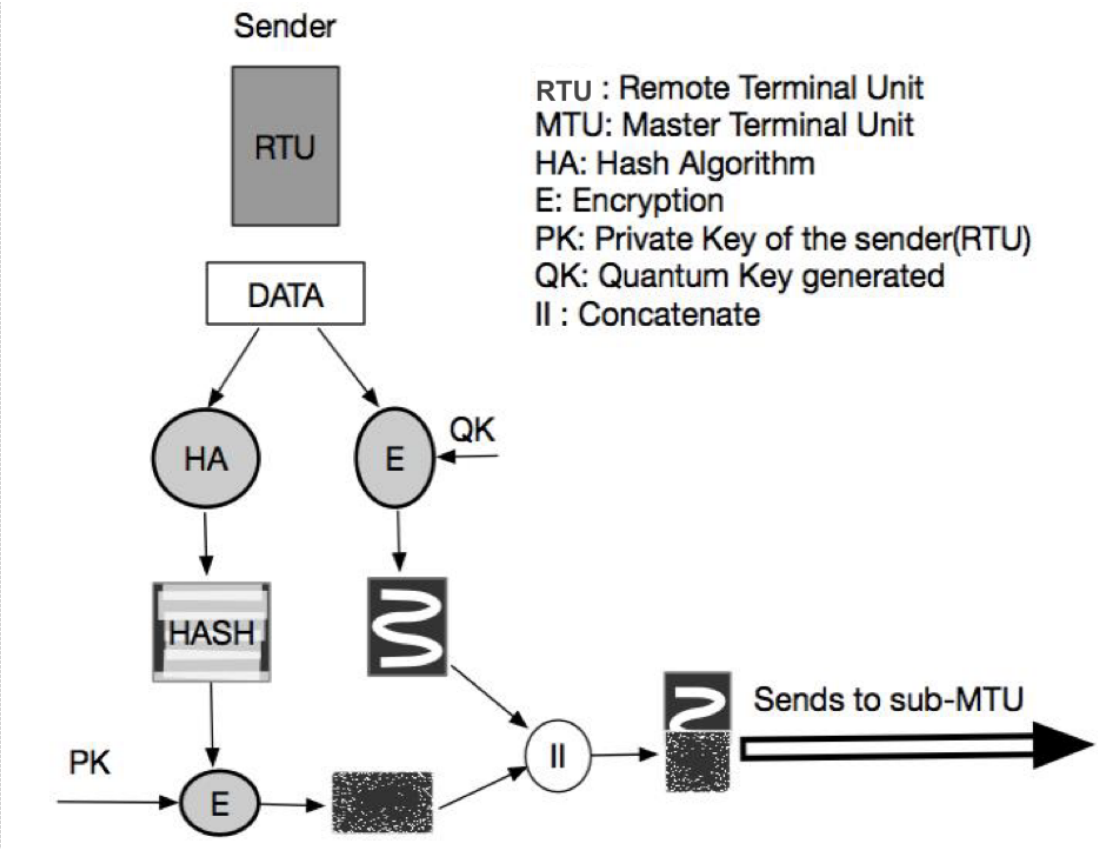

Figure 5.2: Operations of RTU (sender). The signcrypted message is sent to sub-MTU/MTU (receiver).

5.2 Signcryption

Both the components have the finalized quantum key (QK). The RTU executes the following steps[91] as shown in Figure 5.2.

- *Encryption:* The RTU makes a copy of the data, encrypts the data with the finalized quantum key.
- *One-Time Digital Signature:*The RTU hashes the copy of the data. It then encrypts the hash with its private key (PK). It segments the quantum key into equal chunks. It, then, generates a private key by applying a hash function on one of the segments of the QK. It concatenates the hashed message, hashed unique ID of the RTU and a timestamp. The PK is used to encrypt the concatenated data and thus, generating a one-time digital signature.

The RTU sends the signcrypted data to the MTU over the classical channel.

5.3 Un-Signcryption

The MTU receives the signcrypted data and executes the following steps. Furthermore, it is assumed that the MTU has the database which stores the information including IDs of all the RTUs and their hash values. The MTU is also aware of the signcryption algorithm used by the RTU.

- *Decryption:* The encrypted data is decrypted with the quantum key (QK).
- *Validation:* The MTU also decrypts The encrypted hashed value with the private key (PK). The MTU hashes the copied data by the same algorithm used by the RTU. Thus, the timestamp and the hashed ID is extracted and verified.

Chapter 6

Analysis and Experimental Results

6.1 Formal Analysis of Proposed Model

The proposed scheme has two major chunks: Quantum part and Classical part. The quantum part involves BB84 protocol and the classical part involves the Signcryption algorithm. Therefore, in this thesis, two tools have been used for the formal analysis of the proposed scheme.

- Modelling and Analysis of BB84 protocol in Prism.
- Modelling and Analysis of Signcryption in Scyther.

6.2 Modelling and Analysis BB84 protocol in Prism

Prism is a probabilistic model checker to model and analyze the systems based on probabilistic behavior. It automatically analyzes the systems to find out flaws and errors in the system specification. The following two types of inputs are fed to the model checker[72].

- The description of the system which is to be designed. It is mostly formulated in process algebras in such a way that can be used as input in model checker.
- A set of rules or properties that the system must follow.

There are two stages to build a model in a prism[72]:
Stage 1: Model the system where all the states and transitions of the system are

represented.
Stage 2: The properties of the system are expressed in temporal logic statements.

When the temporal logic statements are executed against the model, it verifies whether and with what probability the properties hold for the system.

In this thesis, Discrete-Time Markov Chain (DTMC) model has been used to design the BB84 protocol system. While building the system, the following two properties of the system is defined:

- Public channel handles transmission of messages in such a way that they can be monitored. However, the messages cannot be altered by the eavesdropper.
- Quantum channel handles message exchange in such a way that any attempt by eavesdropper to monitor the channel causes an alteration in the message and thus, creates a noise.

Thus, the system detects any eavesdropping attack as well as cloning attack.

The following two types of attack has been designed for this system[55].

- *Intercept-Resend Attack:* The eavesdropper uses the basis once to measure the qubit. It measures a qubit and the state of the qubit changes randomly.
- *Random-Substitute Attack:* The eavesdropper uses the basis twice. At first, it uses the basis to measure the qubit. After fetching the value of the qubit, it reads the same qubit again to replace its value. It is an attempt to clone the state of the qubit.

In this thesis, we are analyzing the following three factors of the BB84 protocol.

- Whether the protocol detects any intrusion?

- How much information is leaked processing the protocol?
- Can BB84 protocol discard or prevent the eavesdropping attack?

The following six variables have been calculated and used in the models[55]:

- P1 = Probability of detecting an eavesdropper (EVE).
- P2 = Probability that EVE measures more than half of the information correctly.
- N = no. of bits transferred
- Correct bits measured by Eve > = N/2.
- L = LUCKY = Probability of obtaining correct value with wrong basis.
- REPLACE = 0.5 = Probability to use to substitute with 0 or 1.

In this model, probability value ranges from 0 to 1.

Based on the type of attacks, there are two major models:

Model1: BB84 with intercept-resend eavesdropping attack.

Model2: BB84 with random-substitute eavesdropping attack.

6.2.1 Model1: BB84 with intercept-resend eavesdropping attack

Table 6.1: Probability of detecting of Intercept-Resend eavesdropping when Lucky is 0.5.

MODEL 1	P1	P2
N = 4	0.938	0.145
N = 5	0.969	0.155
N = 6	0.984	0.065
N = 7	0.992	0.067
N = 8	0.996	0.028
	LUCKY= 0.5	

Table 6.2: Probability of detecting of Intercept-Resend eavesdropping when N is 5.

MODEL 1	P1	P2
L= 0.5	0.968	0.155
L = 0.6	0.968	0.174
L = 0.7	0.968	0.193
L= 0.8	0.968	0.285
	N=5	

6.2.2 Model2: BB84 with random-substitute eavesdropping attack

Table 6.3: Probability of detecting of Random-Substitute eavesdropping when Lucky is 0.5.

MODEL 1	P1	P2
N = 4	0.938	0.145
N = 5	0.969	0.155
N = 6	0.984	0.065
N = 7	0.992	0.067
N = 8	0.996	0.028
	LUCKY= 0.5	

Table 6.4: Probability of detecting of Random-Substitute eavesdropping when N is 5.

MODEL 1	P1	P2
L= 0.5	0.969	0.155
L = 0.6	0.969	0.174
L = 0.7	0.969	0.193
L= 0.8	0.969	0.285
	N=5	

6.3 Modelling and Analysis of Signcryption in Scyther

In this section, formal analysis of the signcryption scheme is presented. Scyther is a tool which verifies traditional security and authentication protocols[31].

Using Scyther, two main properties is analyzed: Secrecy and Authentication[31].

Secrecy: The following two assumptions are considered for the system[31]

- The sender or the receiver is communicating a trusted party.
- The sender and the receiver are communicating over an untrusted channel.

*Authentication:*The four factors that are assumed for the system are as following[31].

- Aliveness: There is at least one communication partner in the network.
- Synchronization: The intended party is aware of the authenticity of the other party to which it is communicating with.
- The protocol is executing.
- Message Agreement: The message sent by the sender is intact and not tampered. Thus, it has been exchanged as expected.

Furthermore, in the proposed signcryption model, we have used two keys. One, the quantum key is denoted as qk. The qk is used for encryption of messages. Two, sk denotes private key in the model. It is used to generate the digital signature. It provides authentication to the scheme. Both qk and sk are secret and private. Figure 6.1 and Figure 6.2 exhibits the verification results of a simple authentication protocol and the proposed signcryption scheme respectively.

claim	ns3,A	Secret_A2	na	Ok	[proof of correctness]
claim	ns3,A	Secret_A3	qk	Fail	[at least 2 attacks]
claim	ns3,A	Alive_A4	-	Fail	[at least 2 attacks]
claim	ns3,A	Weakagree_A5	-	Fail	[at least 2 attacks]
claim	ns3,A	Commit_A6	(B,na)	Fail	[at least 2 attacks]
claim	ns3,A	Commit_A7	(B,nb)	Fail	[at least 2 attacks]
claim	ns3,A	Niagree_A8	-	Fail	[at least 2 attacks]
claim	ns3,A	Nisynch_A9	-	Fail	[at least 2 attacks]
claim	ns3,B	Secret_B2	na	Fail	[at least 1 attack]
claim	ns3,B	Secret_B3	nb	Ok	[proof of correctness]
claim	ns3,B	Alive_B4	-	Fail	[at least 1 attack]
claim	ns3,B	Weakagree_B5	-	Fail	[at least 1 attack]
claim	ns3,B	Commit_B6	(A,na,nb)	Fail	[at least 1 atta
claim	ns3,B	Niagree_B7	-	Fail	[at least 1 attack]
claim	ns3,B	Nisynch_B8	-	Fail	[at least 1 attack]

Figure 6.1: Verification results of a simple authentication protocol.

```
Sagarikas-MacBook-Air:scyther-mac-v1.1.3 sg$ ./scyther/scyther-mac --dot-
--output=ns3-attacks.dot ns3.spdl
claim   ns3,A   Secret_A2       na      Ok      [proof of correctness]
claim   ns3,A   Secret_A3       qk      Ok      [proof of correctness]
claim   ns3,A   Secret_A4       sk(A)   Ok      [proof of correctness]
claim   ns3,A   Alive_A5        -       Ok      [does not occur]
claim   ns3,A   Weakagree_A6    -       Ok      [does not occur]
claim   ns3,A   Commit_A7       (B,na)  Ok      [does not occur]
claim   ns3,A   Commit_A8       (B,nb)  Ok      [does not occur]
claim   ns3,A   Niagree_A9      -       Ok      [does not occur]
claim   ns3,A   Nisynch_A10     -       Ok      [does not occur]
claim   ns3,B   Secret_B2       na      Ok      [proof of correctness]
claim   ns3,B   Secret_B3       nb      Ok      [proof of correctness]
claim   ns3,B   Secret_B4       qk      Ok      [proof of correctness]
claim   ns3,B   Alive_B5        -       Ok      [does not occur]
claim   ns3,B   Weakagree_B6    -       Ok      [does not occur]
claim   ns3,B   Commit_B7       (A,na,nb)       Ok      [does not occur]
claim   ns3,B   Niagree_B8      -       Ok      [does not occur]
claim   ns3,B   Nisynch_B9      -       Ok      [does not occur]
```

Figure 6.2: Verification results of the proposed Signcryption scheme.

6.4 Evaluation

We executed the proposed scheme in python 3.6. We simulated the quantum channel with noise by designing a binary symmetric channel (BSC). In BSC, the sender sends a bit with value being either 0 or 1. The receiver receives that bit. However, there is a small probability that the bit is flipped in the channel[70].

To generate qubits and to measure their state, Quantum Information Toolkit (QIT) has been used[5]. To implement basis, two types of operators, Pauli X operator and Hadamard operator, has been used[30].

Pauli X operator: It acts on a single qubit and flips its sate. It maps $|0>$ to $|1>$ and $|1>$ to $|0>$.

Hadamard Operator: It provides the property of Hadamard quantum gate. When it applies on a qubit with state $|0>$ or $|1>$, there is an equal probability that the outcome state is either $|0>$ or $|1>$. Furthermore, if the Hadamard gate applies twice on the same qubit, the final state is always the same as the initial state.

In evaluation testing, the experiment splits into two groups:

Group1: It involves performing the proposed scheme on 128-bit initial or raw key.
Group2: It involves performing the proposed scheme on 256-bit initial or raw key.

The simulation parameters of each group are as follows:

- Error rate
- Sifted key size
- Final key size
- Execution time
- Digital Signature
- Time to generate a raw key

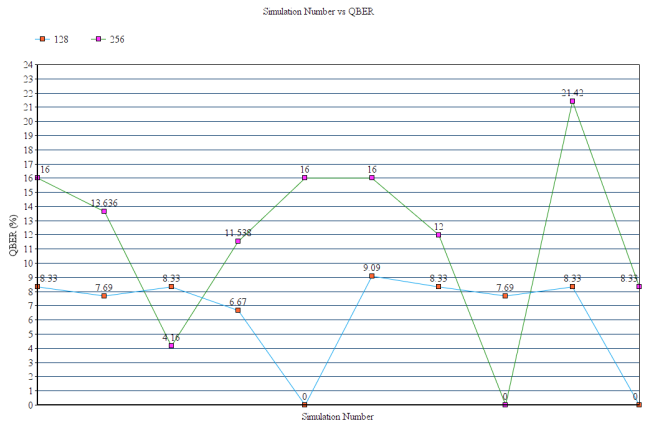

Figure 6.3: Simulation Number vs QBER

Figure 6.3 illustrates the behavior of the scheme with 128-bit and 256-bit raw key in terms of error rate. The coefficient of variation of this parameter for the 128-bit raw key is 0.507, and that of 256-bit is 0.502. Also, Figure6.10 illustrates that the QBER evidently increases as the size of raw key increases.

Figure 6.4 shows the behavior of the scheme with 128-bit and 256-bit raw key in terms of sifted key size. The coefficient of variation of this parameter for the 128-bit and 256-bit is 0.082 and 0.062. Furthermore, Figure6.10 illustrates that the sifted key size is directly proportional to the raw key size.

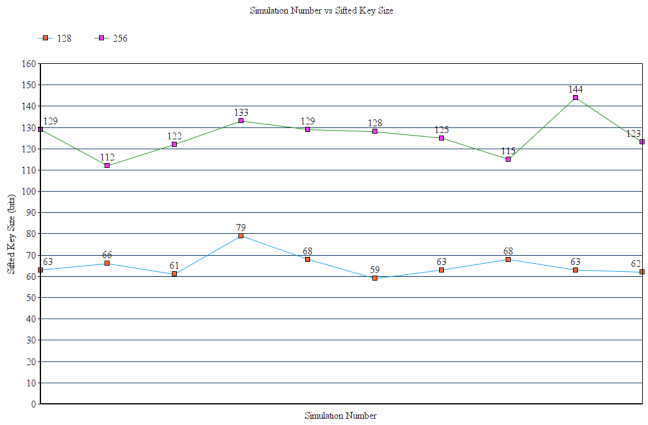

Figure 6.4: Simulation Number vs Sifted key size

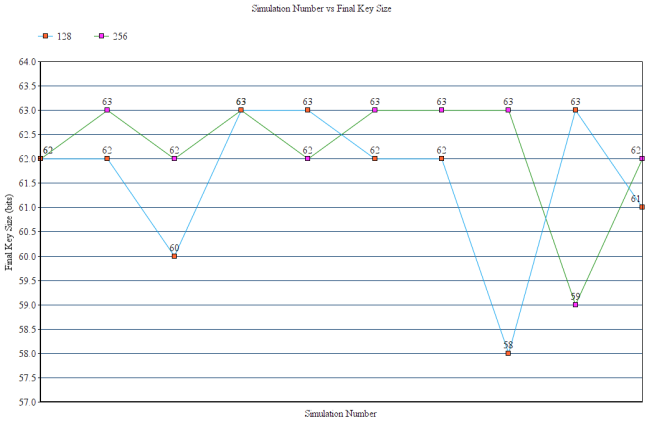

Figure 6.5: Simulation Number vs Final Key Size

Figure 6.5 displays the behavior of the scheme with 128-bit and 256-bit raw key in terms of final key size. The coefficient of variation of this parameter for the 128-bit and 256-bit is 0.024 and 0.018. Figure6.10 illustrates that the final key size does not vary when the raw key size varies.

Figure 6.6 illustrates the behavior of the scheme with 128-bit and 256-bit raw key in terms of digital signature size. The coefficient of variation of this parameter for the 128-bit and 256-bit is 0.0065 and 0.009. Moreover, Figure6.10 illustrates that the digital signature size does not vary when the raw key size is doubled.

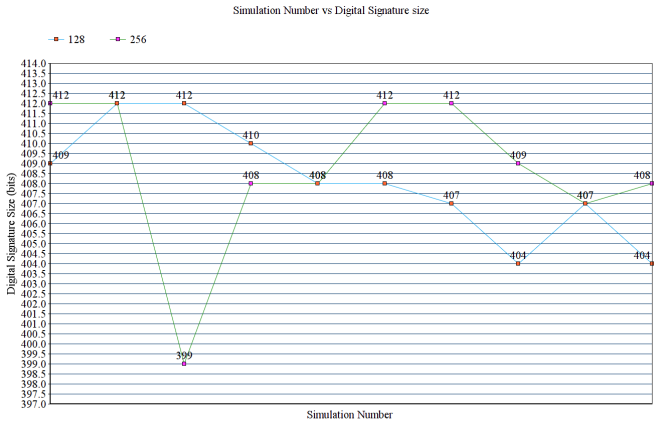

Figure 6.6: Simulation Number vs Digital Signature Size

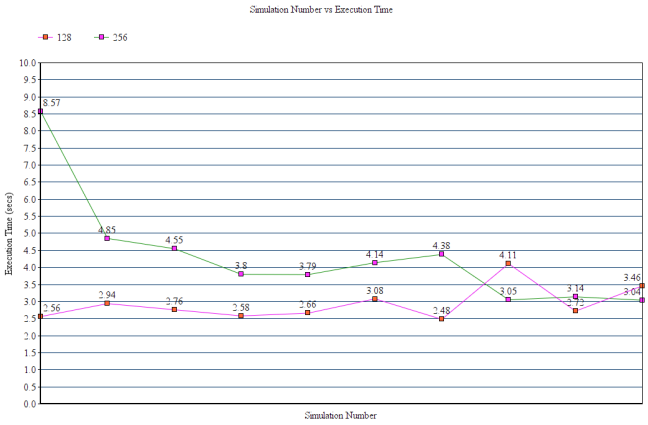

Figure 6.7: Simulation Number vs Execution Time

Figure 6.7 explains the behavior of the scheme with 128-bit and 256-bit raw key in terms of execution time. Furthermore, Figure 6.10 illustrates that the execution time significantly changes when the raw key is adjusted.

Figure 6.8 displays the time to generate the raw keys or the initial keys. We observe that the generation time is directly proportional to the generation time. It is more evident when we compare the mean values of the generation time of each group, as shown in Figure 6.9. For each data communication, Alice generates a new raw key and obtains the private and secret key. Therefore, the private key and secret key randomly varies in each communication.

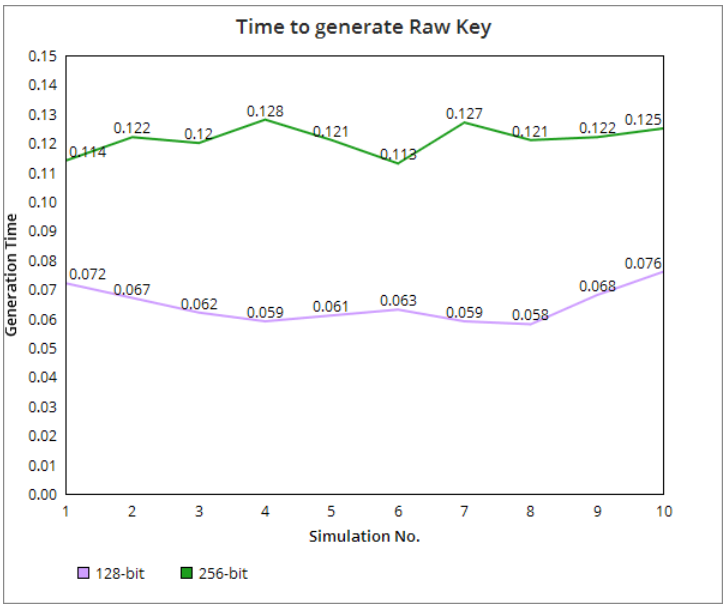

Figure 6.8: Simulation Number vs Time to generate Raw Key(Generation Time)

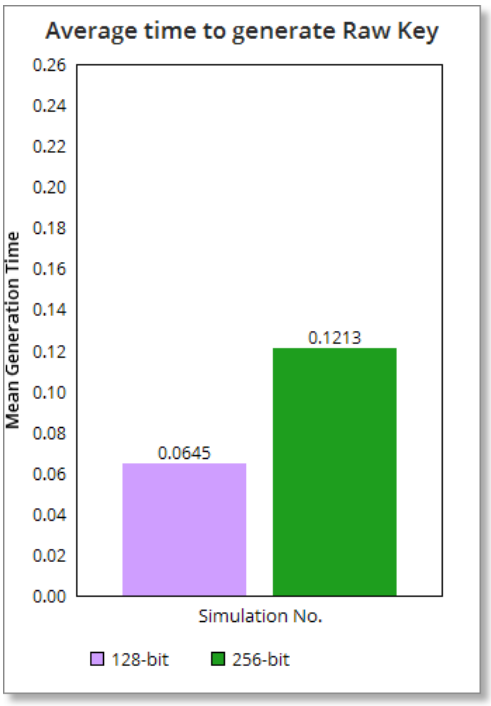

Figure 6.9: Comparison of Group1:128-bit raw key vs Group2:256-bit raw key, using the mean value of Generation Time of each group

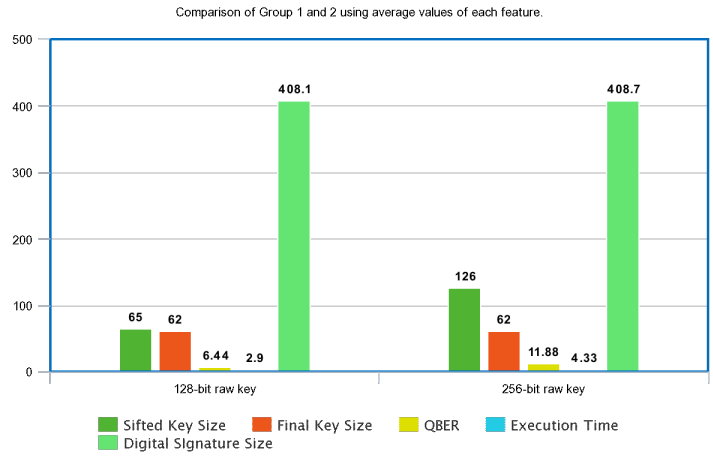

Figure 6.10: Comparison of Group1:128-bit raw key vs Group2:256-bit raw key, using the mean value of each feature.

6.5 Benefits of the Proposed Scheme

In this thesis, a novel security scheme is proposed, which uses the properties of quantum mechanics to provide the following benefits to SCADA networks.

- It resists not only the attacks of traditional computers but also quantum computers. It mainly defends against all kinds of Man-in-the-Middle attack.
- It is an encryption algorithm which also acts as an intrusion detection system.
- The scheme also adds authentication of every communication between components.
- It does not rely on any third-party for generating and validating keys and authentication.

Chapter 7

Conclusion and Future Work

7.1 Conclusion

In this thesis, we have proposed a scheme for securing communication in SCADA networks using BB84 protocol and one-time signcryption. This scheme does not use any trusted third party and yet attains all the security goals: integrity, confidentiality, availability, authentication, and non-repudiation. The randomness property of the key and its size enhances the security of the scheme. It uses uncertainty and superposition properties of quantum physics to detect any eavesdropping. As compared to other protocols, it acts as an encryption scheme as well as an IDS. Therefore, it reduces the computational cost.

7.2 Future Work

As a part of our future work, we intend to perform a comparative analysis between the current standard AGA-12 and our proposed scheme. Additionally, we will do a comparative study by implementing various hash functions in the signature scheme and find out which yields the most potent hash. Our current proposed schemes focus on two quantum attacks: MiM and Brute-force attack on existing SCADA networks. However, an eavesdropper can use the properties of quantum mechanics to launch probe attacks on Quantum Key Distribution[13][87].In our future work, we will focus on these types of attacks, mainly, entangling-probe and Fuchs-Peres-Brandt (FPB) probe attack.

Appendix A

Copyright Permissions

A.1 A Survey of Security in SCADA Networks: Current Issues and Future Challenges[37]

Bibliography

[1] 11 offensive security tools for sysadmins - hacking like a pro. https://hackinglikeapro.blogspot.com/2015/06/11-offensive-security-tools.html. (Accessed on 07/03/2019).

[2] An analysis of fragmentation attacks. http://www.ouah.org/fragma.html. (Accessed on 07/04/2019).

[3] Bergami.pdf. https://www.math.u-bordeaux.fr/~ybilu/algant/documents/theses/BERGAMI.pdf. (Accessed on 07/06/2019).

[4] Quantum computing kills encryption — hackaday. https://hackaday.com/2015/09/29/quantum-computing-kills-encryption/. (Accessed on 07/06/2019).

[5] Quantum information toolkit — quantum information toolkit 0.11.0 documentation. http://qit.sourceforge.net/docs/html/. (Accessed on 07/06/2019).

[6] Session hijacking process — ethical hacking. https://www.greycampus.com/opencampus/ethical-hacking/session-hijacking-process. (Accessed on 07/03/2019).

[7] Ieee guide for electric power substation physical and electronic security. *IEEE Std 1402-2000*, pages i–, 2000.

[8] Cyber security (permanent). https://www.nerc.com/pa/Stand/Pages/Cyber-Security-Permanent.aspx, 2005. (Accessed on 07/04/2019).

[9] Hosny Abbas and Samir Shaheen. Future scada challenges and the promising solution: the agent-based scada. volume 10, pages 307 – 333, 01 2014.

[10] Ghazi Muhammad Abdullah, Quzal Mehmood, and Chaudry Bilal Ahmad Khan. Adoption of lamport signature scheme to implement digital signatures in iot. In *2018 International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*, pages 1–4. IEEE, 2018.

[11] Salah Abdulghani Alabady and Fadi Al-Turjman. Low complexity parity check code for futuristic wireless networks applications. *IEEE Access*, 6:18398–18407, 2018.

[12] P. K. Amiri. Quantum computers. *IEEE Potentials*, 21(5):6–9, Dec 2003.

[13] Hiroo Azuma. An entangling-probe attack on shor's algorithm for factorization. *Journal of Modern Optics*, 65(4):415–422, 2018.

[14] Bijoy Babu, Thafasal Ijyas, P Muneer, and Justin Varghese. Security issues in scada based industrial control systems. In *2017 2nd International Conference on Anti-Cyber Crimes (ICACC)*, pages 47–51. IEEE, 2017.

[15] H. Barwick. Security threats explained: Third party access - computerworld. `https://www.computerworld.com.au/article/429271/`, 2012. (Accessed on 07/06/2019).

[16] Justin M Beaver, Raymond C Borges-Hink, and Mark A Buckner. An evaluation of machine learning methods to detect malicious scada communications. In *2013 12th International Conference on Machine Learning and Applications*, volume 2, pages 54–59. IEEE, 2013.

[17] Charles H Bennett and Gilles Brassard. Quantum cryptography: public key distribution and coin tossing. *Theor. Comput. Sci.*, 560(12):7–11, 2014.

[18] Daniel J Bernstein. Introduction to post-quantum cryptography. In *Post-quantum cryptography*, pages 1–14. Springer, 2009.

[19] Stephanie Blanda. Shor's algorithm – breaking rsa encryption — ams grad blog. `https://blogs.ams.org/mathgradblog/2014/04/30/shors-algorithm-breaking-rsa-encryption/`, April 2014. (Accessed on 07/03/2019).

[20] Tom Carlson. Information security management: understanding iso 17799. *Lucent Technologies*, 2001.

[21] S. Castellanos. Nascent quantum computing poses threat to cybersecurity - cio journal. - wsj. `https://blogs.wsj.com/cio/2017/09/13/nascent-quantum-computing-poses-threat-to-cybersecurity/`, September 2017. (Accessed on 07/03/2019).

[22] US CERT. Russian government cyber activity targeting energy and other critical infrastructure sectors — cisa. 2018. (Accessed on 07/03/2019).

[23] Narasimham Challa and Jayaram Pradhan. Performance analysis of public key cryptographic systems rsa and ntru. *International Journal of Computer Science and Network Security*, 7(8):87–96, 2007.

[24] L. Chang. How secure is today's encryption against quantum computers? `https://betanews.com/2017/10/13/current-encryption-vs-quantum-computers/`, 2017. (Accessed on 07/03/2019).

[25] Donghyun Choi, Hanjae Jeong, Dongho Won, and Seungjoo Kim. Hybrid key management architecture for robust scada systems. *Journal of information science and engineering*, 29(2):281–298, 2013.

[26] Donghyun Choi, Hakman Kim, Dongho Won, and Seungjoo Kim. "advanced key-management architecture for secure scada communications". volume 24, pages 1154–1163, 2009.

[27] Sanjana P Choudhari and Megha B Chakole. Reed solomon code for wimax network. In *2017 International Conference on Communication and Signal Processing (ICCSP)*, pages 0176–0179. IEEE, 2017.

[28] Frances Cleary and Massimo Felici. *Cyber Security and Privacy: 4th Cyber Security and Privacy Innovation Forum, CSP Innovation Forum 2015, Brussels, Belgium April 28-29, 2015, Revised Selected Papers*, volume 530. Springer, 2015.

[29] Gregory M Coates, Kenneth M Hopkinson, Scott R Graham, and Stuart H Kurkowski. A trust system architecture for scada network security. *IEEE Transactions on Power Delivery*, 25(1):158–169, 2009.

[30] Williams C.P. Quantum gates. In *Explorations in Quantum Computing.Texts in Computer Science.*, pages 1–5. Springer, London, 2011.

[31] Cas Cremers and Sjouke Mauw. Operational semantics of security protocols. In *Scenarios: Models, Transformations and Tools*, pages 66–89. Springer, 2005.

[32] Robert Dawson, Colin Boyd, Ed Dawson, and Juan Manuel González Nieto. Skma: a key management architecture for scada systems. In *Proceedings of the 2006 Australasian workshops on Grid computing and e-research-Volume 54*, pages 183–192. Australian Computer Society, Inc., 2006.

[33] IS Decisions. Iso 15408 compliance. `https://www.isdecisions.com/compliance/ISO-15408-compliance.htm`. (Accessed on 07/04/2019).

[34] R. Dennis. Quantum computers are the most powerful tech threat to cryptocurrency. `https://blog.icoalert.com/quantum-computers-are-the-most-powerful-tech-threat-cryptocurrency-will-face`, March 2018. (Accessed on 07/03/2019).

[35] Mohamed Endi, Yehia Z. Elhalwagy, and Attalla hashad. Three-layer plc/scada system architecture in process automation and data monitoring. volume 2, pages 774–779, 2010.

[36] Bernd Fröhlich, Marco Lucamarini, James F Dynes, Lucian C Comandar, Winci W-S Tam, Alan Plews, Andrew W Sharpe, Zhiliang Yuan, and Andrew J Shields. Long-distance quantum key distribution secure against coherent attacks. *Optica*, 4(1):163–167, 2017.

[37] S. Ghosh and S. Sampalli. A survey of security in scada networks: Current issues and future challenges. *IEEE Access*, pages 1–1, 2019.

[38] H. Hilal and A. Nangim. Network security analysis scada system automation on industrial process. In *2017 International Conference on Broadband Communication, Wireless Sensors and Powering (BCWSP)*, pages 1–6, Nov 2017.

[39] Jeffrey Hoffstein, Jill Pipher, and Joseph H Silverman. Ntru: A ring-based public key cryptosystem. In *International Algorithmic Number Theory Symposium*, pages 267–288. Springer, 1998.

[40] Kim Holl. Osi defense in depth to increase application security. 2003. (Accessed on 07/03/2019).

[41] H.Pandey. Computer network — hamming code - geeksforgeeks. `https://www.geeksforgeeks.org/computer-network-hamming-code/`. (Accessed on 07/06/2019).

[42] Control Microsystems Inc. Scadapack e configuration technical reference. 2013. (Accessed on 07/03/2019).

[43] Infosec. Popular tools for brute-force attacks [updated for 2019]. `https://resources.infosecinstitute.com/popular-tools-for-brute-force-attacks/#gref`, 2019. (Accessed on 07/03/2019).

[44] E. Irmak and İ. Erkek. An overview of cyber-attack vectors on scada systems. In *2018 6th International Symposium on Digital Forensic and Security (ISDFS)*, pages 1–5, March 2018.

[45] Rong Jiang, Rongxing Lu, Jun Luo, Chengzhe Lai, and Xuemin Shen. Efficient self-healing group key management with dynamic revocation and collusion resistance for scada in smart grid. *Security and communication networks*, 8(6):1026–1039, 2015.

[46] James S Johnson, Michael R Grimaila, Jeffrey W Humphries, and Gerald B Baumgartner. An analysis of error reconciliation protocols used in quantum key distribution systems. *The Journal of Defense Modeling and Simulation*, 12(3):217–227, 2015.

[47] Audun Jøsang, Bander AlFayyadh, Tyrone Grandison, Mohammed AlZomai, and Judith McNamara. Security usability principles for vulnerability analysis and risk assessment. In *Twenty-Third Annual Computer Security Applications Conference (ACSAC 2007)*, pages 269–278. IEEE, 2007.

[48] R. Kalluri, L. Mahendra, R. K. S. Kumar, and G. L. G. Prasad. Simulation and impact analysis of denial-of-service attacks on power scada. In *2016 National Power Systems Conference (NPSC)*, pages 1–5, Dec 2016.

[49] D. Kang, J. Lee, S. Kim, and J. Park. Analysis on cyber threats to scada systems. In *2009 Transmission Distribution Conference Exposition: Asia and Pacific*, pages 1–4, Oct 2009.

[50] Mikołaj Karpinski, Tomasz Gancarczyk, Aleksandra Klos-Witkowska, Igor Limar, and Yevhen Vasiliu. Security amplification of the computer-aided voting system using quantum cryptography protocols. In *2017 9th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*, volume 1, pages 91–94. IEEE, 2017.

[51] Kaspersky. Computer viruses vs. network worms — kaspersky. `https://usa.kaspersky.com/resource-center/threats/computer-viruses-vs-worms`. (Accessed on 07/03/2019).

[52] Si-Jung Kim, Bong-Han Kim, Sang-Soo Yeo, and Do-Eun Cho. Network anomaly detection for m-connected scada networks. In *2013 Eighth International Conference on Broadband and Wireless Computing, Communication and Applications*, pages 351–354. IEEE, 2013.

[53] D. Kostadinov. Key elements of an information security policy. `https://resources.infosecinstitute.com/key-elements-information-security-policy/#gref`, 2018. (Accessed on 07/06/2019).

[54] Niraj Kumar and Debabrata Goswami. Quantum algorithm to solve a maze: converting the maze problem into a search problem. *arXiv preprint arXiv:1312.4116*, 2013.

[55] S. Kuppam. 68345.pdf. `https://www.scitepress.org/papers/2018/68345/68345.pdf`, 2018. (Accessed on 07/06/2019).

[56] R. Fernanado D. Pundick L. Ayala, K. Burton-Weisman and S. McDermott. Cybersecurity lexicon. In *Eds. California: Apress Media LLC,*, pages 29–50, 2016.

[57] Yong-Hun Lim. Ikms—an id-based key management architecture for scada system. In *7th International Conference on Networked Computing*, pages 139–144. IEEE, 2011.

[58] Fang Xi Lin. Shor's algorithm and the quantum fourier transform. *McGill University*, 2014.

[59] R. Lopez Perez, F. Adamsky, R. Soua, and T. Engel. Machine learning for reliable network attack detection in scada systems. In *2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*, pages 633–638, Aug 2018.

[60] Xin Lu and Dengguo Feng. Quantum digital signature based on quantum one-way functions. In *The 7th International Conference on Advanced Communication Technology, 2005, ICACT 2005.*, volume 1, pages 514–517. IEEE, 2005.

[61] Ben Lynn. Number theory - euclid's algorithm. `https://crypto.stanford.edu/pbc/notes/numbertheory/euclid.html`. (Accessed on 07/03/2019).

[62] Leandros A Maglaras and Jianmin Jiang. Intrusion detection in scada systems using machine learning techniques. In *2014 Science and Information Conference*, pages 626–631. IEEE, 2014.

[63] Leandros A Maglaras and Jianmin Jiang. Ocsvm model combined with k-means recursive clustering for intrusion detection in scada systems. In *10th International conference on heterogeneous networking for quality, reliability, security and robustness*, pages 133–134. IEEE, 2014.

[64] Jeff Melnick. Day: May 15, 2018 - netwrix blog. 2018. (Accessed on 07/03/2019).

[65] Suvo Mittra. Iolus: A framework for scalable secure multicasting. In *ACM SIGCOMM Computer Communication Review*, volume 27, pages 277–288. ACM, 1997.

[66] P. Nader, P. Honeine, and P. Beauseroy. l_p-norms in one-class classification for intrusion detection in scada systems. *IEEE Transactions on Industrial Informatics*, 10(4):2308–2317, Nov 2014.

[67] S. Nazir, S. Patel, and D. Patel. Autonomic computing meets scada security. In *2017 IEEE 16th International Conference on Cognitive Informatics Cognitive Computing (ICCI*CC)*, pages 498–502, July 2017.

[68] NERC. Microsoft word - physical security guideline 2012-05-18-final.docx. `https://www.nerc.com/comm/CIPC/Security%20Guidelines%20DL/Physical%20Security%20Guideline%202012-05-18-Final.pdf`. (Accessed on 07/04/2019).

[69] D. of S. Department. api 1164 pipeline scada security vol.1 - google search. `https://www.google.com/search?rlz=1C5CHFA_enCA843CA843&ei=m_0fXcnjBdKp_Qa71Z24DQ&q=api+1164+pipeline+scada+security+vol.1&oq=api+1164+pipeline+scada+security+vol.&gs_l=psy-ab.3.0.33i160l5.6378.10218..11912...1.0..0.175.714.1j5......0....1..gws-wiz.....0i71j0i22i30j33i21.olHuZ_kAwgY`. (Accessed on 07/05/2019).

[70] University of Victoria. coding515.pdf - ece 515 information theory channel capacity and coding 1 information theory problems how to transmit or store information as efficiently. `https://www.coursehero.com/file/35896396/coding515pdf/`, 2016. (Accessed on 07/06/2019).

[71] V Padamvathi, B Vishnu Vardhan, and AVN Krishna. Quantum cryptography and quantum key distribution protocols: A survey. In *2016 IEEE 6th International Conference on Advanced Computing (IACC)*, pages 556–562. IEEE, 2016.

[72] Nikolaos K Papanikolaou. Techniques for design and validation of quantum protocols. 2005.

[73] Bilal Parvez, Junaid Ali, Usman Ahmed, and Muhammad Farhan. Framework for implementation of aga 12 for secured scada operation in oil and gas industry. In *2015 2nd International Conference on Computing for Sustainable Global Development (INDIACom)*, pages 1281–1284. IEEE, 2015.

[74] Amritha Puliadi Premnath, Ju-Yeon Jo, and Yoohwan Kim. Application of ntru cryptographic algorithm for scada security. In *2014 11th International Conference on Information Technology: New Generations*, pages 341–346. IEEE, 2014.

[75] Quantiki. Shor's factoring algorithm — quantiki. `https://www.quantiki.org/wiki/shors-factoring-algorithm`, October 2015. (Accessed on 07/03/2019).

[76] Abdalhossein Rezai, Parviz Keshavarzi, and Zahra Moravej. Advance hybrid key management architecture for scada network security. *Security and communication networks*, 9(17):4358–4368, 2016.

[77] Abdalhossein Rezai, Parviz Keshavarzi, and Zahra Moravej. Key management issue in scada networks: A review. volume 20, pages 354 – 363, 2017.

[78] Martyn Riley.

[79] R. J. Robles, M. Balitanas, R. Caytiles, Y. Gelogo, and T. Kim. Comparison of encryption schemes as used in communication between scada components. In *2011 International Conference on Ubiquitous Computing and Multimedia Applications*, pages 115–118, April 2011.

[80] Margaret Rouse. What is trojan horse (computing)? - definition from whatis.com. `https://searchsecurity.techtarget.com/definition/Trojan-horse`, January 2018. (Accessed on 07/03/2019).

[81] Sudhir K Routray, Mahesh K Jha, Laxmi Sharma, Rahul Nyamangoudar, Abhishek Javali, and Sutapa Sarkar. Quantum cryptography for iot: Aperspective. In *2017 International Conference on IoT and Application (ICIOT)*, pages 1–4. IEEE, 2017.

[82] A. N. Laboratory S. N. Laboratories, I. N. Laboratory and N. N. L. Pacific. "a summary of control system security standards activities in the energy sector". `https://scholar.google.ca/scholar?hl=en&as_sdt=0%2C5&q=S.+N.+Laboratories%2C+I.+N.+Laboratory%2C+A.+N.+Laboratory%2C+and+N.+N.+L.+Pacific%2C+%E2%80%9CA+Summary+of+Control+System+`

Security+Standards+Activities+in+the+Energy+Sector%2C%E2%80%9D+2005.&btnG=, 2005. (Accessed on 07/03/2019).

[83] A. Sajid, H. Abbas, and K. Saleem. Cloud-assisted iot-based scada systems security: A review of the state of the art and future challenges. volume 4, pages 1375–1384, 2016.

[84] H. Saputra and Z. Zhao. Long term key management architecture for scada systems. In *2018 IEEE 4th World Forum on Internet of Things (WF-IoT)*, pages 314–319, Feb 2018.

[85] N. Sayegh, A. Chehab, I. H. Elhajj, and A. Kayssi. Internal security attacks on scada systems. In *2013 Third International Conference on Communications and Information Technology (ICCIT)*, pages 22–27, June 2013.

[86] Roman Schlegel, Sebastian Obermeier, and Johannes Schneider. Assessing the security of iec 62351. In *Proceedings of the 3rd International Symposium for ICS & SCADA Cyber Security Research*, pages 11–19. BCS Learning & Development Ltd., 2015.

[87] Jeffrey H Shapiro and Franco NC Wong. Attacking quantum key distribution with single-photon two-qubit quantum logic. *Physical Review A*, 73(1):012315, 2006.

[88] Akshay Sharma. Top10 powerfull dos/ddos attacking tools for linux,windows & android - thehackerstuff. https://thehackerstuff.com/top10-powerfull-ddos-tools-linux-windows/, August 2017. (Accessed on 07/03/2019).

[89] Philip Sibson, Chris Erven, Mark Godfrey, Shigehito Miki, Taro Yamashita, Mikio Fujiwara, Masahide Sasaki, Hirotaka Terai, Michael G Tanner, Chandra M Natarajan, et al. Chip-based quantum key distribution. *Nature communications*, 8:13984, 2017.

[90] Varinder Singh and Narinder Sharma. A review on various error detection and correction methods used in communication. *American International Journal of Research in Science, technology, Engineering and Mathematics*, 15:252–257, 2015.

[91] Elif Ustundag Soykan, Seda Demirag Ersoz, and Gurkan Soykan. Identity based signcryption for advanced metering infrastructure. In *2015 3rd International Istanbul Smart Grid Congress and Fair (ICSG)*, pages 1–5. IEEE, 2015.

[92] Trihedral. Vts/vtscada. 2016. (Accessed on 07/03/2019).

[93] TutorialsPoint. Ethical hacking tutorial. https://www.tutorialspoint.com/ethical_hacking/. (Accessed on 07/03/2019).

[94] Ullah and Q. H. Mahmoud. A hybrid model for anomaly-based intrusion detection in scada networks. In *2017 IEEE International Conference on Big Data (Big Data)*, pages 2160–2167, Dec 2017.

[95] A. Williams. Rsa encryption cracked easily (sometimes) — hackaday. `https://hackaday.com/2019/01/16/rsa-encryption-cracked-easily-sometimes/`, 2019. (Accessed on 07/06/2019).

[96] Hao Yan, Xiang Peng, Xiaxiang Lin, Wei Jiang, Tian Liu, and Hong Guo. Efficiency of winnow protocol in secret key reconciliation. In *2009 WRI World Congress on Computer Science and Information Engineering*, volume 3, pages 238–242. IEEE, 2009.

[97] Yi Yang, Keiran McLaughlin, Tim Littler, Sakir Sezer, and HF Wang. Rule-based intrusion detection system for scada networks. 2013.

[98] Xin Zhang, Zhao Yang Dong, Zeya Wang, Chixin Xiao, and Fengji Luo. Quantum cryptography based cyber-physical security technology for smart grids. 2015.

[99] Y. Zhang, Y. Xiang, and L. Wang. Reliability analysis of power grids with cyber vulnerability in scada system. In *2014 IEEE PES General Meeting — Conference Exposition*, pages 1–5, July 2014.

[100] B. Zhu, A. Joseph, and S. Sastry. A taxonomy of cyber attacks on scada systems. In *2011 International Conference on Internet of Things and 4th International Conference on Cyber, Physical and Social Computing*, pages 380–388, Oct 2011.