INFORMATION THEORETICALLY SECURE ENHANCED JOHNSON NOISE BASED KEY DISTRIBUTION OVER THE SMART GRID

A Dissertation

by

ELIAS GONZALEZ

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Chair of Committee,	Laszlo Kish
Committee Members,	Jun Zou
	Robert Balog
	Andreas Klappenecker
Head of Department,	Miroslav Begovic

August 2016

Major Subject: Electrical Engineering

ABSTRACT

The imperative need for unconditionally secure key exchange is caused by the increasing connectivity of networks and by the increasing number and level of sophistication of cyberattacks. Two concepts that are information theoretically secured are quantum key distribution (QKD) and Kirchhoff-Law-Johnson-Noise (KLJN). However, these concepts require a dedicated connection between hosts in peer-to-peer (P2P) networks which can be impractical and/or cost prohibitive. A practical and cost effective method is to have each host share their respective cable(s) with other hosts such that two remote hosts can realize a secure key exchange without the need of an additional cable or key exchanger.

We introduce a protocol for linear chain networks with a reconfigurable filter system to create non-overlapping single loops in the smart power grid for the realization of the Kirchhoff-Law-Johnson-(like)-Noise secure key distribution system. The protocol is valid for one-dimensional daisy chain networks (chain-like power line) which are typical of the electric distribution network between the utility and the customer. The speed of the protocol (the number of steps needed) versus grid size is analyzed. When properly generalized, such a system has the potential to achieve unconditionally secure key distribution over the smart power grid of arbitrary geometrical dimensions.

In this work we also analyze the cost complexities of cable, key exchangers, and time required in the star network. We mention the reliability of the star network and compare it with other network geometries. We also conceived a protocol and equation for the number of secure bit exchange periods needed in a star network. We then outline other network geometries and trade-off possibilities that seem interesting

to explore.

We also propose a new key exchange trust evaluation for peer-to-peer sensor networks, where part of the network has unconditionally secure key exchange. As the utilization of sensor networks continues to increase, the importance of security becomes more profound. Many industries depend on sensor networks for critical tasks, and a malicious entity can potentially cause catastrophic damage. For a given sensor, the higher the portion of channels with unconditionally secure key exchange, the higher the trust value. We give a brief introduction to unconditionally secured key exchange concepts and mention current trust measures in sensor networks. We demonstrate the new key exchange trust measure on a hypothetical sensor network using both wired and wireless communication channels.

DEDICATION

To God, whose will be done.

ACKNOWLEDGEMENTS

First and foremost a colossal thank you goes to my advisor Laszlo Kish for his infinite patience, and his outstanding leadership, guidance, support, and encouragement. Another huge thank you goes to my committee: Dr. Jun Zou, Dr. Robert Balog, and Dr. Andreas Klappenecker. An enormous thank you goes to professors Dr. Chanan Singh, Dr. Prasad Enjeti, and Dr. Aydin Karsilayan. A gigantic thank you goes to my colleagues Hsien-Pu Chen and Yessica Saez. Also, a hearty thank you goes to my loving and supporting family: Natalie, mom, dad, and sister. My work would not be possible without your support, thank you all!

NOMENCLATURE

AC	Alternating Current
BE	Bit Exchange
CA	Certificate Authority
DC	Direct Current
DDoS	Distributed Denial of Service
FCN	Fully Connected Network
G	Geometric
IoT	Internet of Things
KE	Key Exchange period
KLJN	Kirchhoff-Law-Johnson-Noise
LCH	Linear CHain network
MD5	Message Digest Version 5
MITM	Man-In-The-Middle
NSA	National Security Agency
P2P	Peer-to-Peer
PKI	Public Key Infrastructure
QKD	Quantum Key Distribution
RR	Round Robin
SBEP	Secure Bit Exchange Period
SHA-1	Secure Hash Algorithm Version 1
STAR	Star network

TABLE OF CONTENTS

Page

ABSTRACT ii

DEDICATION iv

ACKNOWLEDGEMENTS v

NOMENCLATURE vi

TABLE OF CONTENTS vii

LIST OF FIGURES x

LIST OF TABLES xiv

1. BACKGROUND AND INTRODUCTION OF SECURE KEY EXCHANGE FOR UNCONDITIONALLY SECURE KEY EXCHANGE ON SMART GRIDS[1,2,3] 1

    1.1 Background 1
    1.2 Software-Based Key Exchange 3
    1.3 Unconditionally Secure Key Exchange 4
    1.4 Hardware-Based Secure Key Exchanges 4
    1.5 Quantum Key Distribution (QKD) 5
    1.6 Kirchhoff-Law-Johnson-Noise (KLJN) 5
    1.7 Applying KLJN to Smart Grid Networks 8

2. UTILIZING SWITCHED FILTERS ON THE SMART GRID TO REALIZE UNCONDITIONALLY SECURE KEY EXCHANGE OVER LINEAR CHAIN NETWORKS[1] 9

    2.1 Motivation 9
        2.1.1 KLJN, the Information Theoretically Secure Wire-Based Key Exchange Scheme 9
        2.1.2 Utilizing the Smart Power Grid for Information Theoretic Secure Key Exchange 12
    2.2 Discussions and Results 13
        2.2.1 Switched Filters 13

2.2.2 Protocol and Speed 19
2.3 Limitations of Realizing KLJN over the Smart Grid, Open Questions, and Future Work 36
2.3.1 Limitations 36
2.3.2 Open Questions 37
2.3.3 Future Work 37

3. A PROTOCOL FOR IMPLEMENTING UNCONDITIONALLY SECURE KEY EXCHANGE ON A STAR NETWORK IN THE SMART GRID AND COMPARING COST COMPLEXITIES AND ROBUSTNESS WITH DIFFERENT NETWORK TOPOLOGIES[2] 39

3.1 Securing Networks 39
3.1.1 Motivation for a Secure Network 39
3.1.2 Secure Key Exchange over P2P Networks and the Fully Connected Network 39
3.1.3 Linear Chain Network with Two Key Exchangers per Host . . 42
3.2 Results and Discussion 43
3.2.1 Star Network 43
3.2.2 Graph Theory and Previous Work on the Star Network 43
3.2.3 Protocol and Analysis of the Star Network 45
3.2.4 Comparing Network Topologies 52
3.2.5 Open Questions and Future Studies 53

4. EVALUATING KEY EXCHANGE TRUST IN SENSOR NETWORKS WITH CONSIDERATION OF UNCONDITIONALLY SECURE KEY EXCHANGE[3] 55

4.1 Sensor Networks, Security Concerns, Trust Mechanisms, and Unconditionally Secure Key Exchanges. 55
4.1.1 Sensor Networks 55
4.1.2 Security Concerns 56
4.1.3 Trust Mechanisms 56
4.1.4 Motivation for a Key Exchange Trust Evaluation 57
4.2 Outline of Combined Wired and Wireless Sensor Networks 57
4.2.1 Network 58
4.2.2 Protocol 59
4.3 Geometric Key Exchange Trust System 60
4.3.1 The Key Exchange Trust Function 60
4.3.2 The Kill Switch 60
4.3.3 Construction of the Key Exchange Trust Function 61
4.3.4 Derivation of G 64
4.3.5 Example 69

4.4 Open Questions and Future Work 70

5. CONCLUSION[1,2,3] 72

REFERENCES 75

LIST OF FIGURES

FIGURE Page

1.1 An illustration of Alice and Bob in a secure key exchange while Eve is seeking to tap the communication channel and extract the key. . . 2

1.2 An illustration of the core KLJN system. Alice and Bob each have a communicator which have noise generators, a "Low" resistor $R_{\rm L}$ (representing the Low bit value), and a "High" resistor $R_{\rm H}$ (representing the High bit value.) The noise voltages are enhanced by generators emulating Johnson noise, $U_{\rm A,L}$ or $U_{\rm A,H}$ for Alice: and $U_{\rm B,L}$ or $U_{\rm B,H}$ for Bob, at very high temperature. Once the communicators select a resistor they measure the mean-squared voltage amplitude $< U^2_{\rm ch}(t) >$ and/or the current amplitude $< I^2_{\rm ch}(t) >$. There is a wire for the key exchange, and there is a channel for data exchange. Against active attacks and attacks exploiting ratio non-idealities, an authenticated public data channel is used to measure and compare bits [47, 54]. . . 7

2.1 Example of a one-dimensional grid, we call it a chain network. This example has a network of size $N = 7$. 14

2.2 Building blocks in a filter box. 15

2.3 Example for network of size $N = 7$. Each host is connected to a filter box and the filters boxes are connected to the power grid. Note how each host has three wire connections to its filter box. 16

2.4 The filter box of the inactive host (when a host is not executing a KLJN key exchange): State 1. Everything ($B_{\rm kljn}$ and $f_{\rm p}$) is passing between the left and right filters, and the host can only access power. Filter A is passing everything (shorted). Filter B is disconnected. Filter C is passing $B_{\rm kljn}$ only. Filters E and D are passing $f_{\rm p}$ only. State 1 is when the host is not allowed to access the KLJN band. State 2 is when the host is allowed to access the KLJN band. This filter box is in State 1. 17

2.5 The filter box of the active host (when a host is executing a KLJN key exchange): State 2. Power is passing between the left and right filters, but the KLJN band is not. The left and right KLJN units are separated while executing a key exchange with hosts towards its left and right side. State 1 is when a host is not allowed to access the KLJN band. State 2 is when a host is allowed to access the KLJN band. This filter box is in State 2. 18

2.6 The first step in the protocol connects the nearest neighbors. This step is the quickest and most efficient. It has the most non-overlapping simultaneous loops and requires only 1 KE to complete. Every host in this step has access to the KLJN band and thus are in State 2. . . 21

2.7 The second step in the protocol connects the second nearest neighbors. This step is the second quickest and the second most efficient. It has the second most non-overlapping simultaneous loops and requires 2 KEs to complete. 22

2.8 The third step in the protocol connects the third nearest neighbors. This step is not as efficient as the first two steps but still has simultaneous loops. This step requires 3 KEs to complete. 23

2.9 The fourth step in the protocol connects the fourth nearest neighbors. This step is the slowest and least efficient step in the protocol in our example of $N = 7$. This step requires 4 KEs to complete. 24

2.10 The fifth step in the protocol connects the fifth nearest neighbors. This step is not efficient since simultaneous non-overlapping loops with disconnected hosts cannot occur. 25

2.11 The sixth step in the protocol connects the sixth nearest neighbors. This step requires only 2 KEs since there are only two possibilities. . 26

2.12 Only one key exchange is performed in this step. Host 1 through 6 are not allowed access to the KLJN band thus they are in State 1. This is the seventh and the last step. This step is not efficient but only requires one KE since there is only one such pair of hosts. 26

2.13 The first step in the protocol connects the nearest neighbors. This step is the quickest and most efficient. It has the most non-overlapping simultaneous loops and requires only 1 KE to complete. 29

2.14 The second step in the protocol connects the second nearest neighbors. This step requires 2 KEs to complete. 29

2.15 The third step in the protocol connects the third nearest neighbors. This step requires 3 KEs to complete. 30

2.16 The fourth step in the protocol connects the fourth nearest neighbors. It requires 4 KEs to complete. 31

2.17 The fifth step in the protocol connects the fifth nearest neighbors. This step is not efficient since simultaneous non-overlapping loops with disconnected hosts cannot occur. It requires 4 KEs to complete. . . . 32

2.18 The sixth step in the protocol connects the sixth nearest neighbors. This step requires only 3 KEs since it is the third to last step and there are only three possibilities. 33

2.19 The seventh step in this example network of size $N = 8$. This step is not efficient but only requires two KEs since there are only two such pairs of hosts. 34

2.20 The last step in the protocol connects the first and last hosts. This step is the least efficient and requires the entire length of the network. Since there is only one pair of hosts at this length this step requires only one KE. 34

3.1 An illustration of a fully connected network with $N-1$ communicators per host (denoted as FCN_{N-1}) has cost complexities of $T_{\mathrm{cable}}(N) \in O(N^2)$, $T_{\mathrm{ke}}(N) \in O(N^2)$, and $T_{\mathrm{time}}(N) \in O(1)$. 41

3.2 An illustration of a linear chain network with 2 key exchangers per host has cost complexities of $T_{\mathrm{cable}}(N) \in O(N)$, $T_{\mathrm{ke}}(N) \in O(N)$, and $T_{\mathrm{time}}(N) \in O(N^2)$. 42

3.3 An illustration of a star network system with one key exchanger per host has cost complexities of $T_{\mathrm{cable}}(N) \in O(N)$, $T_{\mathrm{ke}}(N) \in O(N)$, and $T_{\mathrm{time}}(N) \in O(N)$. 44

3.4 An illustration of an example of the STAR network protocol for a network with five hosts. It takes six Secure Bit Exchange Period (SBEP) steps for every host in the network to process a secure bit exchange with every other host. 47

3.5 This is the plot of Table 3.3. The data points are plotted along with a linear regression line which is $f(N) = 1.3192982456 \cdot N - 1.301754386$, and the coefficient of determination is $R^2 = 0.988989157$. The horizontal axis (independent variable) is N, meaning the number of hosts in the star network. The vertical axis (dependent variable) is SBEP(N), meaning the number of SBEP steps needed for a network with N hosts. 50

4.1 An illustration of a wired-wireless hybrid sensor network. In this example there are ten sensors with only select sensors utilizing wired communication channels and all sensors utilizing wireless communication channels. 59

LIST OF TABLES

TABLE Page

2.1 Truth table of the KLJN Filters in State 1 (inactive host). 17

2.2 Truth table of the Power Filters in State 1 (inactive host). 17

2.3 Truth table of the KLJN Filters in State 2 (active host). 18

2.4 Truth table of the Power Filters in State 2 (active host). 19

3.1 This table demonstrates what every host is doing at every SBEP step in the STAR protocol as described in the example and illustrated in Figure 3.4. 48

3.2 This table is the legend of Table 3.1. 48

3.3 This table shows the number of SBEPs needed in star networks with 2 hosts to 20 hosts, for every host in the network to execute a secure bit exchange with every other host. 49

3.4 This table summarizes the cost complexities of the fully connected networks FCN_{N-1}, FCN_1, the linear chain network protocol LCH, and the star network protocol STAR. 52

4.1 This table lists every sensor's key exchange with all sensors in the network of Figure 4.1. Every sensor is classified as having either a wired KLJN key exchange or a wireless key exchange. Set notation is used to categorize the sets as either KLJN or wireless key exchange. 60

4.2 This table lists G_{ij} key exchange trust values for all the sensors in Figure 4.1. This table assumes $\gamma_j = 1$ for all js. 69

1. BACKGROUND AND INTRODUCTION OF SECURE KEY EXCHANGE FOR UNCONDITIONALLY SECURE KEY EXCHANGE ON SMART GRIDS[1,2,3]

1.1 Background

According to the U.S. Code of Laws statute 44 U.S.C. § 3542 (b)(1), information security is the practice of defending information from unauthorized access, use, disclosure, disruption, modification, perusal, inspection, recording, or destruction. Information security is categorized by [86] into areas of data security, network security, computer security, application security, security operations, and physical security. Encryption is often used in data security, and key exchange is often used in encryption [86]. Cryptography is the science of data protection by encryption [90]. To secure data, Alice (sender) would encrypt the data and transmit it to Bob (receiver). Bob then has encrypted data and needs to decrypt it to be able to access the data. To decrypt the data, Bob needs a key (or keys). Alice must securely send Bob the key so that Bob can decipher the data, in what is known as a *secure key exchange*. The key must be sent in a secure method to prevent an eavesdropper (Eve) from

1 Part of this chapter is reprinted with permission from "Information Theoretically Secure, Enhanced Johnson Noise Based Key Distribution over the Smart Grid with Switched Filters" by Gonzalez, E., Kish, L.B., Balog, R.S., Prasad, E., (2013). *PLoS ONE*, 8(7): e70206 Copyright 2013 by PLOS

2 Part of this chapter is reprinted with permission from "Resource Requirements and Speed *versus* Geometry of Unconditionally Secure Physical Key Exchanges" by Gonzalez, E., Balog, R.S., Kish, L.B., (2015). *Entropy*, 17(4), pp. 2010-2024; DOI:10.3390/e17042010 Copyright 2015 by MDPI

3 Part of this chapter is reprinted with permission from "Key Exchange Trust Evaluation in Peer-to-Peer Sensor Networks With Unconditionally Secure Key Exchange" by Gonzalez, E., Kish, L.B., (2016). A print and electronic version of this article published in *Fluctuation and Noise Letters*, Vol. 15, No. 1, 2016, pp. 165008 (17 pages) DOI:10.1142/S0219477516500085 ©World Scientific Publishing Company http://www.worldscientific.com/worldscinet/fnl

obtaining a copy of the key.

Secure communication channels are needed to prevent eavesdropping or intervention. Increasingly though, communications is directed away from expensive, dedicated networks in favor of the open Internet. In order to ensure secure communications, security keys are needed to set up a secure communication. The keys are generated and shared via a publicly accessible channel by secure key distribution protocols. Consider a secure key exchange between Alice and Bob; Alice and Bob must consider that an eavesdropper (Eve) is trying to extract the key as illustrated in Figure 1.1.

Figure 1.1: An illustration of Alice and Bob in a secure key exchange while Eve is seeking to tap the communication channel and extract the key.

Keys can be either symmetric or asymmetric [24]. Symmetrical keys use the same key to encrypt and decrypt data, while asymmetrical keys (also known as public key cryptography) utilizes two keys, where one key is used to encrypt data and another different key is used to decrypt data [24].

Authentication is one of the the most common ways to verify identity. Authentication is the process used to verify an identity, e.g., a username and password is an example of *single-factor authentication* [86]. The password is often stored as a *hash*, which is an alphanumeric code used to hide passwords. Hash functions are

one-way mathematical functions that convert a string of characters to a hash, often used to store passwords and as digital signatures. In theory, hash functions cannot be reversed, unlike encryption methods. Examples of hash functions are Secure Hash Algorithm Version 1 (SHA-1) and Message Digest Version 5 (MD5) [86]. Certificate-based authentication often utilizes information that binds an identity to a public key and a digital signature of the issuing authority known as the Certificate Authority (CA) [86]. Certificates are supported by a governing hierarchical body authorized to distribute certificates, and many certificate infrastructure organizations are known as Public Key Infrastructure (PKI) [86]. Modern secure key exchange requires authentication before keys are exchanged. Secure key exchanges can be categorized as either software-based or hardware-based.

1.2 Software-Based Key Exchange

In software-based key distribution (exchange) protocols, the security is only computationally-conditional, meaning that the eavesdropper has all the communicated information, and with enough computing resources or time, the key can be fully extracted. The advantage of software-based key distributions is that they are relatively cheap, easy to install and run, and the key can be exchanged wirelessly.

Software-based key exchanges are based on mathematical algorithms with the assumption that Eve does not have enough computing resources to crack the key. In essence, software-based key exchanges offer no security from an information theoretical point of view. The security is only (computationally-) conditional and is not *future-proof*, meaning that with enough computing resources the key can be extracted. The advantages of software-based key exchanges are the low cost, hardware communicator is not required, and the keys can be exchanged over the Internet, thus eliminating the need of additional infrastructure. The other option is hardware-based

key exchange, which offers an advantage of unconditional security.

1.3 Unconditionally Secure Key Exchange

Unconditionally secure key exchanges are key distribution methods that are information theoretically secure [63], which means that the information is not in the communicated signal, see next section. Thus even with infinite computing resources the eavesdropper cannot extract the key. However, physical (hardware-based) key exchanges are the only schemes that can provide unconditionally secure key exchange. Hardware-based key exchanges are more expensive than software-based schemes; moreover, wireless key exchange is not possible (except quantum key distribution with single photons, which require complete darkness.)

1.4 Hardware-Based Secure Key Exchanges

So far there are two physical key distribution classes that offer unconditionally secure key exchange: Quantum Key Distribution (QKD) [8] and the Kirchhoff-Law-Johnson-Noise (KLJN) scheme [49, 47, 77, 78, 59, 82, 58, 50, 53, 54, 51, 27, 79, 16, 18, 52, 57, 56, 48, 55, 17].

The Quantum Key Distribution (QKD) and the Kirchhoff-Law-Johnson-Noise (KLJN) secure key exchange, are two examples of hardware-based secure key exchange concepts that are information theoretically secure [63]. Thus, even with infinite computing resources the key will not be extracted by Eve, because the security offered by these schemes is based on fundamental laws of physics; to crack the key exchange would require Eve to break the underpinning laws of physics. The main disadvantage of hardware-based key exchanges is the higher cost, as they require a physical communicator at each host and a dedicated connection between communicators. Such communication schemes can be considered peer-to-peer (P2P) [91].

1.5 Quantum Key Distribution (QKD)

The QKD key exchange utilizes the quantum no-cloning theorem of quantum mechanics [8] to distribute key bits. In theory it is information theoretically secure; however, the physical implementation of QKD has been debated and the method has been hacked [100, 25, 67, 73].

In QKD principle, the bits are exchanged via single photon communications and the physical laws of physics, which provide unconditional secure key exchange by the quantum no-cloning theorem [8]. Recently, the fundamental security proofs for QKD have been debated [100, 101, 36]. QKD has also had issues with the non-ideality of practical building elements, which has lead to the cracking of existing communicators, including commercial devices [36, 71, 26, 74, 98, 72, 40, 69, 66, 68]. Although these practical non-ideality problems can be patched, QKD remains vulnerable until the patch is known and applied. Other concerns with QKD systems are the bulky physical size, relative expense, large power consumption, sensitivity to vibrations, and requirement of a "dark optical fiber." These characteristics of QKD make it almost impossible to integrate into a sensor.

1.6 Kirchhoff-Law-Johnson-Noise (KLJN)

In the KLJN scheme, the key bit is exchanged via a wired channel and utilizes statistical physics [49]. The actual physical laws of providing security are the second law of thermodynamics and the properties of Gaussian fluctuations. Relative to QKD, KLJN can be integrated on a microchip; thus it does not have issues with physical size, energy required, sensitivity to vibrations, etc. KLJN can be implemented into a sensor, but it will require a wire to connect every sensor that intends to acquire an unconditionally secure key exchange.

An illustration of the KLJN setup is in Figure 1.2. In this figure, Alice and

Bob have two identical resistor pairs which are $R_{\rm L}$ for the Low resistor and $R_{\rm H}$ for the High resistor. Each resistor has noise voltages that are enhanced by Johnson noise; $U_{\rm A,L}$ for Alice's Low resistor, $U_{\rm A,H}$ for Alice's High resistor, $U_{\rm B,L}$ for Bob's Low resistor, and $U_{\rm B,H}$ for Bob's High resistor. During the key bit exchange period the first step is for Alice and Bob to select either $R_{\rm L}$ or $R_{\rm H}$. The selection of $R_{\rm L}$ and $R_{\rm H}$ is random and both are equally likely to be selected. Since the selection of $R_{\rm L}$ and $R_{\rm H}$ is random, neither Alice or Bob know which resistor will be selected. Once Alice and Bob select their respective resistor, they measure the voltage and/or current in the wire. The channel voltage can be modeled by $< U_{\rm ch}^2(t) >= 4{\rm k}T_{\rm eff}R_{\rm loop}B_{\rm KLJN}$ and the channel current can be modeled by $< I_{\rm ch}^2(t) >= 4{\rm k}T_{\rm eff}B_{\rm KLJN}/R_{\rm loop}$ with k being Boltzmann's constant, $T_{\rm eff}$ measuring the effective temperature, $R_{\rm loop}$ being the loop resistance, and $B_{\rm KLJN}$ being the KLJN bandwidth [49]. From $< U_{\rm ch}^2(t) >$ or $< I_{\rm ch}^2(t) >$, Alice and Bob know which resistor the other end selected, and they already know which resistor they selected. If the voltage noise level is high, then they both selected high resistors and if the voltage noise level is low, then they both selected low resistors; in these outcomes the key bit is discarded and the next period begins. If an intermediate voltage noise level or current noise level is measured, then a secure key bit is generated, stored, and the next period begins. This process continues until the desired number of key bits are generated.

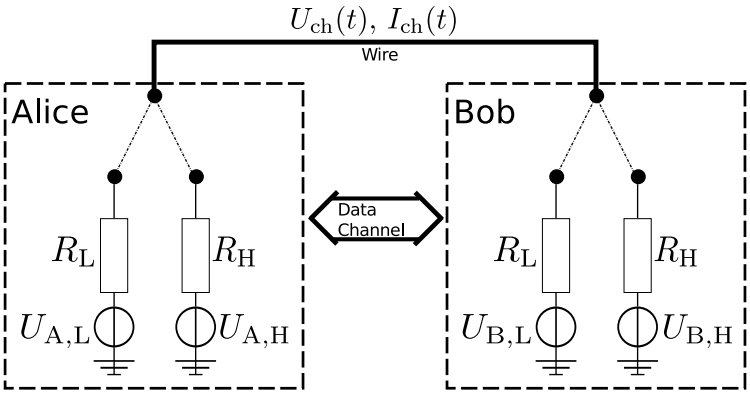

Figure 1.2: An illustration of the core KLJN system. Alice and Bob each have a communicator which have noise generators, a "Low" resistor $R_{\rm L}$ (representing the Low bit value), and a "High" resistor $R_{\rm H}$ (representing the High bit value.) The noise voltages are enhanced by generators emulating Johnson noise, $U_{\rm A,L}$ or $U_{\rm A,H}$ for Alice: and $U_{\rm B,L}$ or $U_{\rm B,H}$ for Bob, at very high temperature. Once the communicators select a resistor they measure the mean-squared voltage amplitude $< U^2_{\rm ch}(t) >$ and/or the current amplitude $< I^2_{\rm ch}(t) >$. There is a wire for the key exchange, and there is a channel for data exchange. Against active attacks and attacks exploiting ratio non-idealities, an authenticated public data channel is used to measure and compare bits [47, 54].

The core system in Figure 1.2 is secure against passive (non-invasive) attacks in the idealized case. However in [47, 54], when Eve is tampering with or changing the system via an active (invasive) intervention, such as launching a MITM (man-in-the-middle) attack [47], the core system is not enough to guarantee security. Similarly, non-idealities, which represent deviations from the original scheme, cause information leak [54]. For defending the system against these kind of attacks, the instantaneous voltage and current amplitudes are measured by Alice and Bob, and these quantities are communicated and compared via a public authenticated data channel. Alice and Bob have a full and deterministic model of the system; because it is a classical physical system, incessant measurement of the current and voltage is allowed. Based on their comparison and preconditions, Alice and Bob decide to keep or discard

the compromised bit [47]. The authentication uses only $\log_2(M)$ secure bits of the exchanged bits, where M is the number of bits carrying the current and voltage data in the public channel. In practical applications, this channel can be wireless or wired.

1.7 Applying KLJN to Smart Grid Networks

The smart grid consists of several geometric topologies such as linear chain networks, star networks, and other networks. Some networks are wireless and thus KLJN cannot be utilized in these situations. To realize KLJN, we propose the use of switched filters and/or install an additional cable to send KLJN bandwidth. We want to minimize cost complexities of the required cables, key exchangers, and time. To minimize cost, we propose protocols that take advantage of network topologies. For wireless networks, we propose the use of a key exchange trust evaluation that gives a higher trust evaluation to sensors or networks with KLJN key exchange and lower evaluations to sensors or networks without KLJN exchange. We discuss and analyze these ideas in the following chapters.

2. UTILIZING SWITCHED FILTERS ON THE SMART GRID TO REALIZE UNCONDITIONALLY SECURE KEY EXCHANGE OVER LINEAR CHAIN NETWORKS[1]

2.1 Motivation

2.1.1 KLJN, the Information Theoretically Secure Wire-Based Key Exchange Scheme

On February 12, 2013, President Obama issued an executive order to outline policies directing companies and operators of vital infrastructure, such as power grids, to set standards for cybersecurity [23]. This step is an indication of the urgent need to protect intelligence, companies, infrastructure, and personal data in an efficient method. In this chapter, we propose a solution that provides information theoretic (that is, unconditional) secure key exchange over the smart grid. This method is controlled by filters and protects against MITM attacks.

A smart grid [5, 44] is an electrical power distribution network that uses information and communications technology to improve the security [75, 60], reliability, efficiency, and sustainability of the production and distribution of electricity. A form of a cyber-physical system, the smart grid enables greater efficiency through a higher degree of awareness and control, but also introduces new failure modes associated with data being intercepted and compromised.

Private key based secure communications require a shared secret key between

1 Part of this chapter is reprinted with permission from "Information Theoretically Secure, Enhanced Johnson Noise Based Key Distribution over the Smart Grid with Switched Filters" by Gonzalez, E., Kish, L.B., Balog, R.S., Prasad, E., (2013). *PLoS ONE*, 8(7): e70206 Copyright 2013 by PLOS

Alice and Bob who may communicate over remote distances. In today's secure communications, sharing such a key also utilizes electronic communications because courier and mail services are slow. However, the software-based key distribution methods offer only limited security levels that are only computationally-conditional; thus, they are not future-proof. By having sufficiently enhanced computing power, Eve can crack the key and all communications that are using that key. Therefore, unconditional security (indicating that the security holds even under infinite computational power), which is the popular wording of the term "information theoretic security" [63], requires more than a software solution. It needs the utilization of the proper laws of physics.

The oldest scheme that claims information theoretic security by utilizing the laws of physics is quantum key distribution (QKD). While the security available in QKD schemes has been debated and compromised [100, 25, 71, 26, 74, 98, 72, 40, 69, 66, 68, 89, 67, 70, 73], the discussions indicate that there is a potential to reach a satisfactory security level in the future, though it may require a new approach [100]. However, current QKD devices are prohibitively expensive and have other practical limitations, such as they are sensitive to vibrations, bulky, limited in range, and require a special "dark optical fiber" cable with sophisticated infrastructure.

On the other hand, the smart grid offers a unique way to process a secure key exchange because each household (host) in the grid is electrically connected. To utilize a wire connection for secure key exchange, a different set of the laws of physics (not the laws applied for QKD that work with optical fibers) must be utilized. Recently a classical statistical physical alternative to QKD, the Kirchhoff-Law-Johnson-(like)-Noise (KLJN) key exchange system has been proposed [49, 47, 77, 78, 59, 82, 58], which is a wired-based scheme that is free from several limitations of QKD. A recent survey is given in [77]. Similar to QKD, KLJN is also an information theoretically

secure key distribution [77]. However, it is robust, not sensitive to vibrations, has unlimited range [78], can be integrated on chips [59], can use existing wire infrastructure such as power lines [82], and KLJN based networks can also be constructed [58].

The KLJN channel is a semiconducting wire [77]. At the beginning of each clock cycle (note, the 50 Hz/60 Hz AC grid provides a universal time synchronization), Alice and Bob, who have an identical pair of resistors $R_{\rm L}$ and $R_{\rm H}$ (representing the 0 and 1 bit situations) randomly select and connect one of the resistors, as shown in Figure 1.2. In practical applications, voltage noise generators enhance the Johnson noise of the resistors so that all resistors in the system have the same, publicly known effective noise-temperature $T_{\rm eff}$ (where $T_{\rm eff} \geq 10^9$ Kelvin). The enhanced Johnson noise voltages $\{U_{\rm L,A}(t)$ or $U_{\rm H,A}(t)$; and $U_{\rm L,B}(t)$ or $U_{\rm H,B}(t)\}$ of the resistor result in a channel noise voltage between the wire and the ground, and a channel noise current, $I_{\rm ch}(t)$, in the wire. Low-pass filters are used because the noise-bandwidth, which we also call KLJN-band $B_{\rm kljn}$ (its value depends on the range), must be chosen so that wave, reflection, and propagation and delay effects are negligible, otherwise the security is compromised [49]. Alice and Bob can measure the mean-squared amplitudes $< U_{\rm ch}^2(t) >$ and/or $< I_{\rm ch}^2(t) >$ within the KLJN-band in the line. From any of these values, the loop resistance can be calculated [49] by using the Johnson noise equations (2.1), with the noise-bandwidth $B_{\rm kljn}$, effective temperature $T_{\rm eff}$, loop resistance $R_{\rm loop}$, and Boltzmann's constant k:

$$\begin{aligned} < U_{\rm ch}^2(t) > &= 4kT_{\rm eff}R_{\rm loop}B_{\rm kljn} \\ < I_{\rm ch}^2(t) > &= \frac{4kT_{\rm eff}B_{kljn}}{R_{\rm loop}}. \end{aligned} \tag{2.1}$$

Alice and Bob know their own choice resistor; thus, from the loop resistance they can deduce the resistance value and the actual bit status at the other end of the wire. In an ideal situation, the cases $R_{\mathrm{L}}|R_{\mathrm{H}}$ and $R_{\mathrm{H}}|R_{\mathrm{L}}$ represent a secure bit exchange event because they cannot be distinguished by the measured mean squared values. Eve can do the very same measurements, but she has no knowledge about any of the resistance selections by Alice and Bob and thus she is unable to extract the key bits from the measured loop resistance.

The KLJN protocol can also be applied to several other wired networks such as electronic equipment that do not desire to be reverse engineered. However, in this chapter we focus on applying the KLJN protocol on the smart grid.

2.1.2 Utilizing the Smart Power Grid for Information Theoretic Secure Key Exchange

The disadvantage of the KLJN key exchange protocol is that it requires a wired connection. Investors are hesitant to cover the cost of additional infrastructure solely for the purpose of security. On the other hand, virtually every building in the civilized world is connected to the electrical power grid. This fact is very motivating to explore the possibility of using the power grid as the infrastructure for the KLJN protocol. However, only the single loop shown in Figure 1.2 is unconditionally secure. When Alice and Bob are two remote hosts in the smart grid, they should indeed experience a single loop connection as in Figure 1.2. Thus for smart grid applications, proper filters must be installed and controlled for the KLJN frequency band to operate. Though simple examples have been outlined to prove that a KLJN key exchange between two remote points in a radial power grid with filters [82, 58] can be achieved, neither details about the structure of the filter units nor network protocols to connect every host on the grid with all other hosts has been shown. The method described

in [58] is high speed, because if the units do simultaneous key exchange they have a joint network key, and the units must trust each other. In the present system the units have independent keys.

The present chapter aims to make the first steps in this direction by presenting a working scheme with scaling analysis of the speed of key exchange versus network size. We limit our network to a one-dimensional linear chain network to utilize the smart power grid for KLJN secure key exchange. We show and analyze a protocol to efficiently supply every host with proper secure keys so that they can separately communicate with all the other hosts.

2.2 Discussions and Results

Because the pattern of connections between KLJN units must be varied to provide the exchange of a separate secure key for each possible pair of hosts, the network of filters and their connections must be varied accordingly. The power line filter technology is already available [6, 45] and we will show that the required results can be achieved by switching on/off proper filtering units in a structured way on the smart grid. We will need filters to pass or reject the KLJN frequency band, $B_{\rm kljn}$, and/or the power frequency, $f_{\rm p}$ (50 or 60 Hz). When both $B_{\rm kljn}$ and $f_{\rm p}$ are passed, it is a short; and when both are rejected, it is a break. We will call these filters *switched filters*.

2.2.1 Switched Filters

We call the functional units connected to the smart power grid hosts. A host is able to execute a KLJN key exchange toward its left and right in a simultaneous manner. That means each host has two independent KLJN units. The filter system must satisfy the following requirements:

(i) Hosts that currently do not execute KLJN key exchange should not interfere

with those processes, even if the KLJN signals passes through their connections.

(ii) Each host should be able to extract electrical power from the electric power system without disturbing the KLJN key exchanges.

We define the size of a network as being of size N when that network has $N + 1$ hosts. An example of a network of size $N = 7$ is illustrated in Figure 2.1.

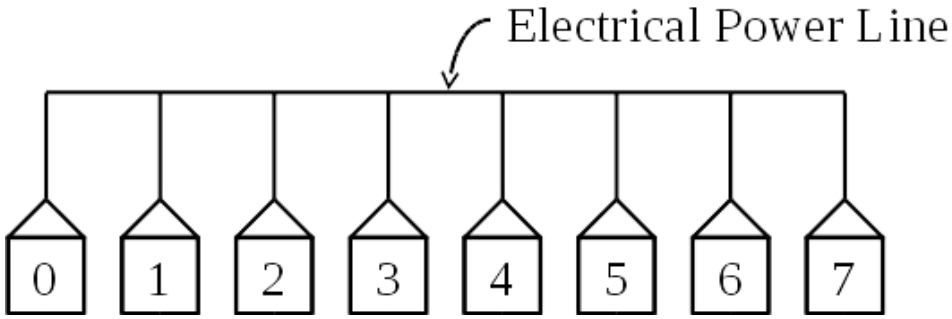

Figure 2.1: Example of a one-dimensional grid, we call it a chain network. This example has a network of size $N = 7$.

Intermediate hosts on the network can be in two different states according to the need:

(α) State 1 is defined when KLJN bandwidth (B_{kljn}) is not allowed into the host.

(β) State 2 is defined when KLJN bandwidth (B_{kljn}) is allowed into the host.

Hosts at the two ends can be in similar situations except that they can communicate in only a single direction; thus they are special, limited cases of the intermediate hosts to which we are focusing our considerations when discussing filters.

Filter boxes at each host will distribute the KLJN signals and the power, and they are responsible for connecting the proper parties for a KLJN key exchange and

to supply hosts with power, see Figure 2.2. The filters boxes can be controlled either by a central server and/or an automatic algorithm. In the following section, we discuss the protocol of this control. Each filter box has three switched filters and a corresponding output wire, see Figure 2.2:

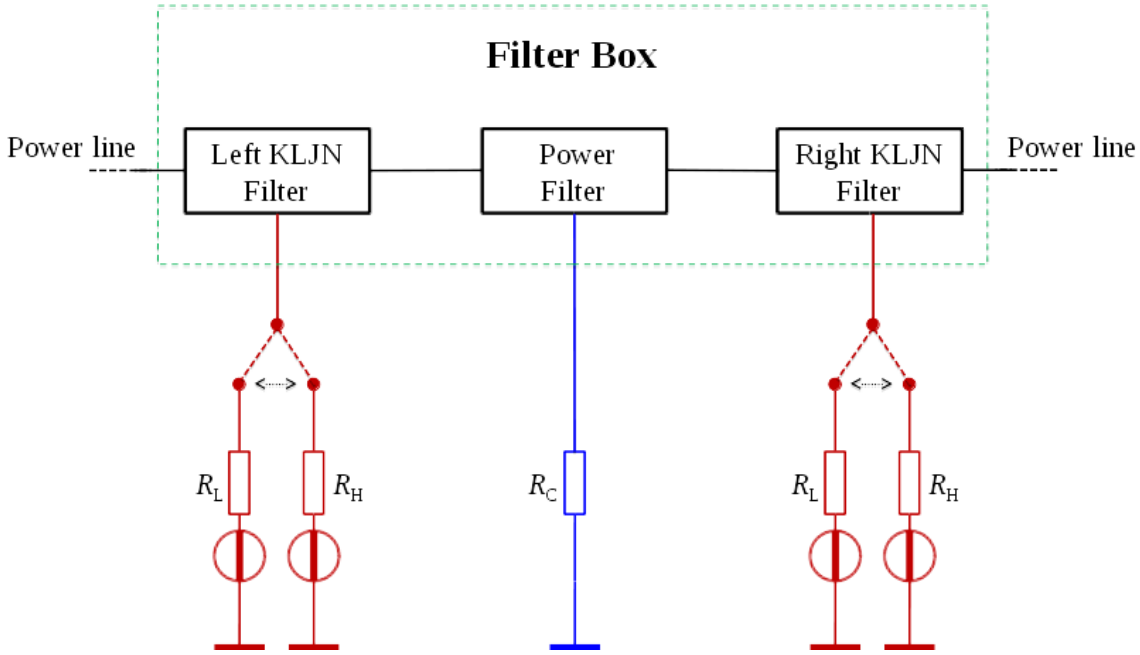

Figure 2.2: Building blocks in a filter box.

(a) The Left KLJN Filter for the KLJN key exchange toward left,

(b) The Right KLJN Filter for the KLJN key exchange toward right.

(c) The Power Filter to supply power to the host.

The properly controlled filter boxes will provide non-overlapping KLJN loops between the hosts, see below. KLJN loops need to be non-overlapping loops because the KLJN protocol is fundamentally P2P. If overlapping loops were allowed, then

there is a possibility that Eve might be in between and will require the trust of the intermediate hosts. A problem with P2P networks is that they require direct connections. QKD also require direct connections. The reason for having two KLJN units per host is to decrease the time needed to connect every host by having simultaneous loops toward left and right, without overlapping. Figure 2.3 shows an example for $N = 7$. The solid black line means that both KLJN bandwidth and power frequency is passing through (ordinary wire: the original line). The (red) dotted lines carry B_{kljn} (f_{p} is rejected). The (blue) dashed lines indicate the opposite situation: only the power frequency is passing and the KLJN bandwidth is rejected.

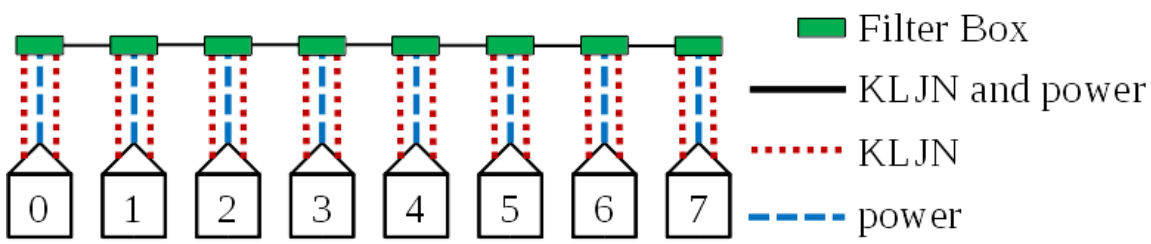

Figure 2.3: Example for network of size $N = 7$. Each host is connected to a filter box and the filters boxes are connected to the power grid. Note how each host has three wire connections to its filter box.

When there is a key exchange between the first host (host 0) and the last host (host 7) over the whole network (Figure 2.3), then all hosts in between (host 1 through host 6) are not allowed to access the KLJN band. In this state, the filter boxes of host 1 through 6 must separate their respective host from the KLJN band, and at the same time supply them power. We call this working mode of the filter boxes of non-active hosts *State 1*. The wiring and frequency transfer of the Filter Box in State 1 are shown in Figure 2.4 and Tables 2.1 and 2.2.

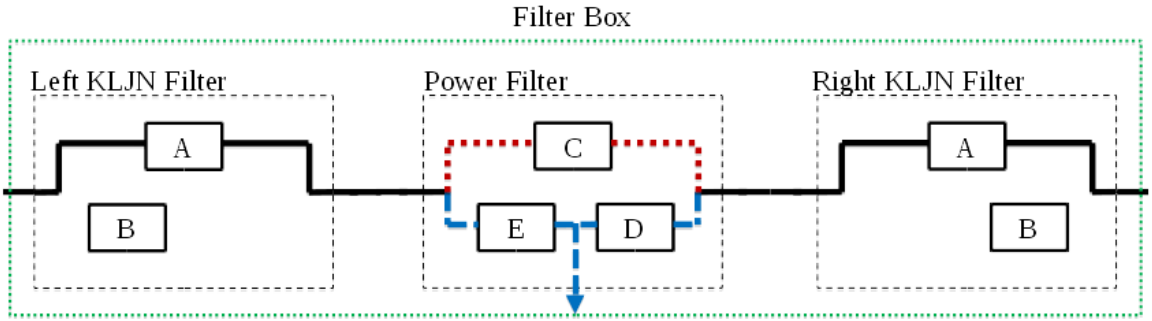

Figure 2.4: The filter box of the inactive host (when a host is not executing a KLJN key exchange): State 1. Everything (B_{kljn} and f_{p}) is passing between the left and right filters, and the host can only access power. Filter A is passing everything (shorted). Filter B is disconnected. Filter C is passing B_{kljn} only. Filters E and D are passing f_{p} only. State 1 is when the host is not allowed to access the KLJN band. State 2 is when the host is allowed to access the KLJN band. This filter box is in State 1.

State 1 (inactive host) KLJN Filters	Filter A	Filter B
KLJN B_{kljn} Allowed?	Yes	No
Power Frequency f_{p} Allowed?	Yes	No

Table 2.1: Truth table of the KLJN Filters in State 1 (inactive host).

State 1 (inactive host) Power Filters	Filter C	Filter D	Filter E
KLJN B_{kljn} Allowed?	Yes	No	No
Power Frequency f_{p} Allowed?	No	Yes	Yes

Table 2.2: Truth table of the Power Filters in State 1 (inactive host).

See Figures 2.5 and 2.6, as additional examples of seven key exchanges occurring simultaneously with every host in that network being active (allowed access to the KLJN band). The KLJN filters of these hosts must separate the KLJN loops by preventing B_{kljn} from entering Filter A. We call this working mode *State 2* with the filter boxes executing key exchanges. The wiring and frequency transfer of the Filter Box in State 2 are shown in Figure 2.5 and Tables 2.3 and 2.4.

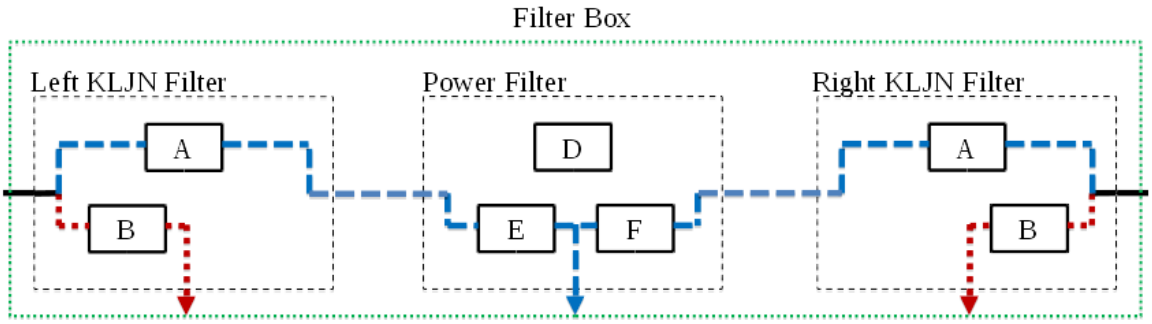

Figure 2.5: The filter box of the active host (when a host is executing a KLJN key exchange): State 2. Power is passing between the left and right filters, but the KLJN band is not. The left and right KLJN units are separated while executing a key exchange with hosts towards its left and right side. State 1 is when a host is not allowed to access the KLJN band. State 2 is when a host is allowed to access the KLJN band. This filter box is in State 2.

State 2 (active host) KLJN Filters	Filter A	Filter B
KLJN B_{kljn} Allowed?	No	Yes
Power Frequency f_{p} Allowed?	Yes	No

Table 2.3: Truth table of the KLJN Filters in State 2 (active host).

State 2 (active host) Power Filters	Filter D	Filter E	Filter F
KLJN B_{kljn} Allowed?	No	No	No
Power Frequency f_{p} Allowed?	No	Yes	Yes

Table 2.4: Truth table of the Power Filters in State 2 (active host).

In this section we have shown that the line can be packed with non-overlapping KLJN loops to execute simultaneous key exchanges between selected hosts. In the next section, we propose a network protocol to provide secure keys for each host so that they will be able to communicate securely via the Internet or other publicly accessible channels between arbitrary pairs of hosts. The time requirement for a key exchange over the entire network versus a network of size N will be analyzed.

2.2.2 Protocol and Speed

To quickly and efficiently connect every host with all other hosts in the same one-dimensional networks we need to establish a protocol. The protocol must make every possible connection in the network, must not overlap loops, and must be quick and efficient by making as many simultaneous loops without overlapping as possible.

To determine the time and speed requirements to establish a KLJN secure key exchange we must first define terms. In the classical KLJN system, where only the noise exist in the wire, the low-frequency cutoff of the noise was 0 Hz and the high-frequency cutoff was B_{kljn}. In the case of KLJN on the smart grid, this situation will be different because of the power frequency. However, at short distances (less than 10 miles), the B_{kljn} band can be beyond the power frequency (f_{p}) and the difference is negligible. Thus the shortest characteristic time in the system is the correlation time τ_{kljn} of the noise ($\tau_{\text{kljn}} \approx 1/B_{\text{kljn}}$). B_{kljn} is determined by the distance L between Alice and Bob so that $B_{\text{kljn}} << \text{c}/L$ [73]; (for example, $B_{\text{kljn}} << 100$ kHz for $L = 1$

kilometer). Alice and Bob must make a statistic on the noise, which typically requires around $100 \cdot \tau_{\text{kljn}}$ duration [78] (or 0.01 seconds if we use $B_{\text{kljn}} = 10$ kHz) to have a sufficiently high fidelity. Note, faster performance is expected in advanced KLJN methods [50]. A bit exchange (BE) occurs when Alice and Bob have different resistor values; this occurs on average of $200 \cdot \tau_{\text{kljn}}$ or 0.02 seconds if $B_{\text{kljn}} = 10$ kHz. The length of the secure key exchange can be any arbitrary length. For example, if we have a key length of 100 bits then we need 100 BE, which requires on average $20000 \cdot \tau_{\text{kljn}}$, which is approximately 2 seconds if B_{kljn} is 10 kHz. Once the KLJN secure key has been exchanged, the total amount of time needed to complete this is one KLJN secure key Exchange period (KE). While the key exchange is slow, the system has the advantage that it is running continuously (not only during the handshake period as some common secure Internet protocols require) and thus a large number of secure key bits are produced during the continuous operation.

For the sake of simplicity, we use a pessimistic estimation by assuming a uniform duration for KE determined by the largest distance in the network, even though short distances can exchange keys at a higher speed.

The protocol we propose here first connects the nearest neighbor of every host; this allows the highest number of simultaneous, non-overlapping loops per KE and only requires one KE to complete this first step. The protocol then connects the second nearest neighbors, which allows the second highest numbers of simultaneous loops per KE. However, due to the requirement of avoiding overlapping loops, connecting each pair of second nearest neighbors requires two KEs. The protocol then connects the third nearest neighbors, which require 3 KEs to complete, and connects the third most simultaneous loops per KE. This procedure continues until the ith nearest neighbor is equal to or less than half of the size of the network. If the number of steps i between the ith nearest neighbors satisfies the relation $i > N/2$, then to

avoid overlapping loops only one connection per KE is possible.

As an example, we will show in the next section that for $N = 7$ (see Figure 2.1), 16 KEs (approximately 32 seconds if B_{kljn} is 10 kHz) are required when the keys are 100 bits long. Using this protocol, the analytical form of the exact time required to fully secure every host with enough keys so that they can securely communicate with every other host on the network is dependent on the size of the network and whether the network has an even or odd size. In the following sections we will deduce the analytical relations and show examples.

2.2.2.1 If N is Odd for a Network of Size N

We illustrate the steps the protocol takes and calculate the time requirements with an example shown in the following figures. A general formula for an arbitrary size network when N is odd is given later. In this example we have a network of size $N = 7$. We have 8 hosts with index i $(0 \leq i \leq 7)$. We have 7 intermediate connections between the first and last host.

The first step in the protocol connects the nearest neighbors, see Figure 2.6.

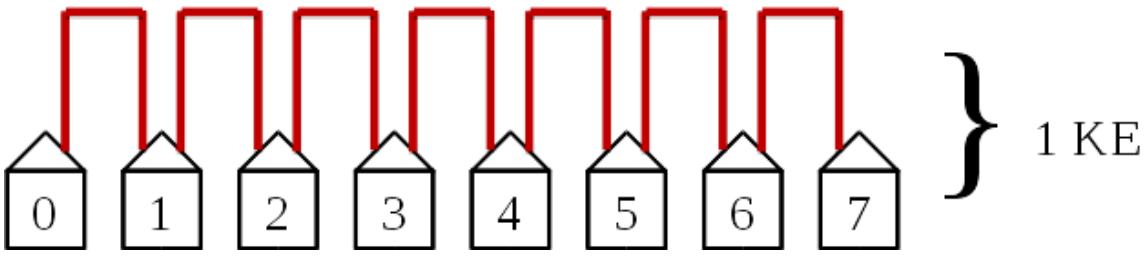

Figure 2.6: The first step in the protocol connects the nearest neighbors. This step is the quickest and most efficient. It has the most non-overlapping simultaneous loops and requires only 1 KE to complete. Every host in this step has access to the KLJN band and thus are in State 2.

The second step in the protocol will then connect the second nearest neighbors, see Figure 2.7. This step is the second quickest and the second most efficient. It has the second most non-overlapping simultaneous loops and requires 2 KEs to complete.

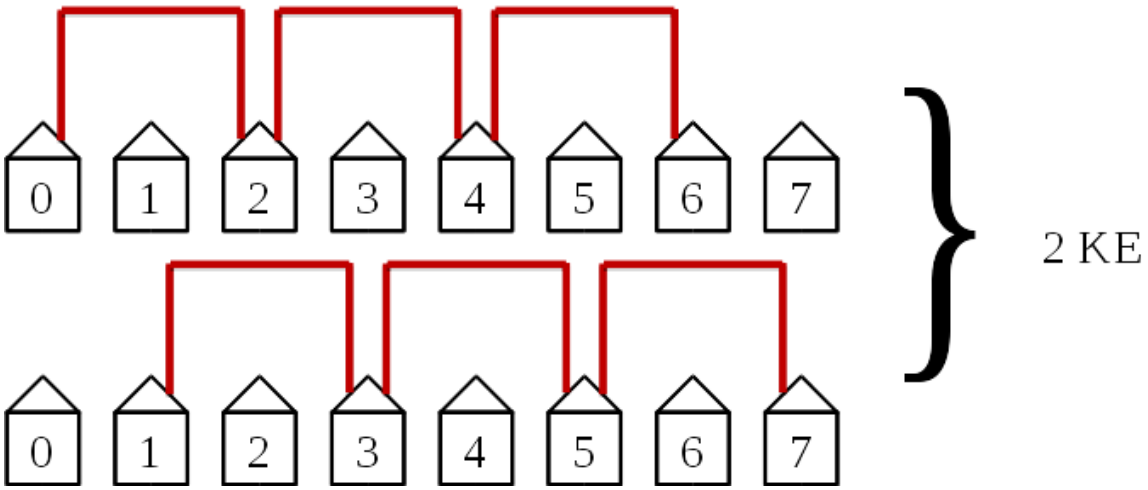

Figure 2.7: The second step in the protocol connects the second nearest neighbors. This step is the second quickest and the second most efficient. It has the second most non-overlapping simultaneous loops and requires 2 KEs to complete.

The protocol will then connect the third closest neighbors as shown in Figure 2.8. This will take 3 KEs to complete and is not as efficient as the first two steps in the protocol but still has simultaneous loops in two of its KE steps.

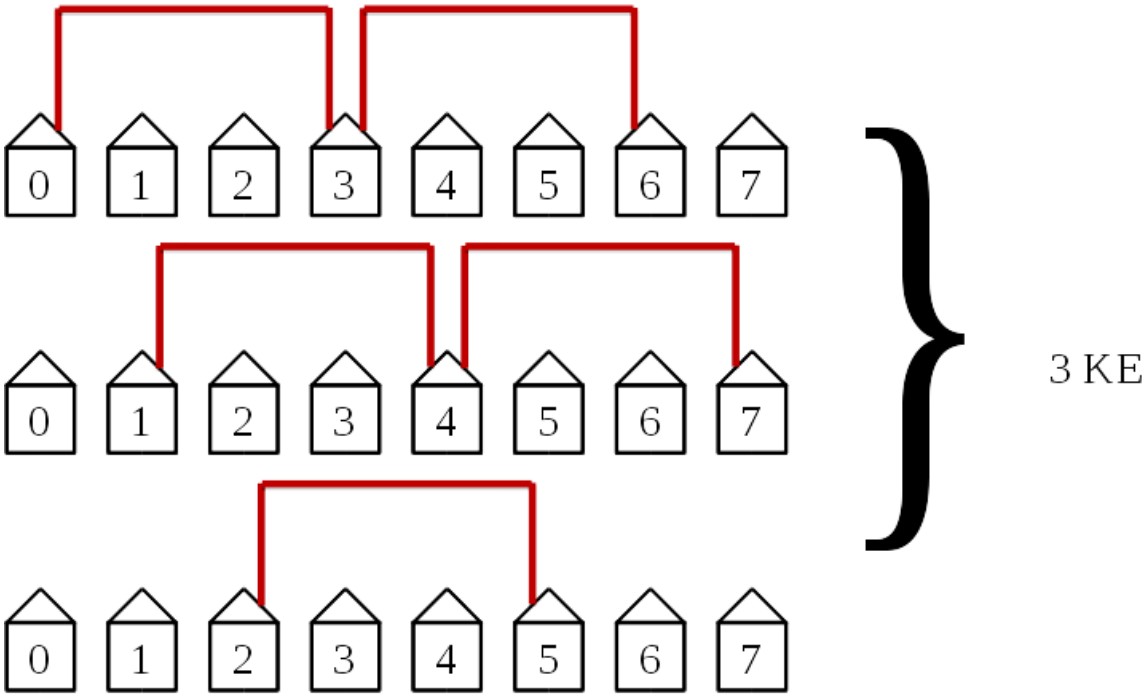

Figure 2.8: The third step in the protocol connects the third nearest neighbors. This step is not as efficient as the first two steps but still has simultaneous loops. This step requires 3 KEs to complete.

The protocol will then connect the fourth nearest neighbors as shown in Figure 2.9. This is above the midpoint for our example with $N = 7$ and is the slowest and least efficient step in the protocol. The midpoint is considered when the distance between Alice and Bob is equal to half the length of the network. These steps will take 4 KEs to complete. Simultaneous loops with disconnected hosts are no longer possible beyond the midpoint (according to our protocol). The slowest and least efficient steps occur at the midpoint of the protocol.

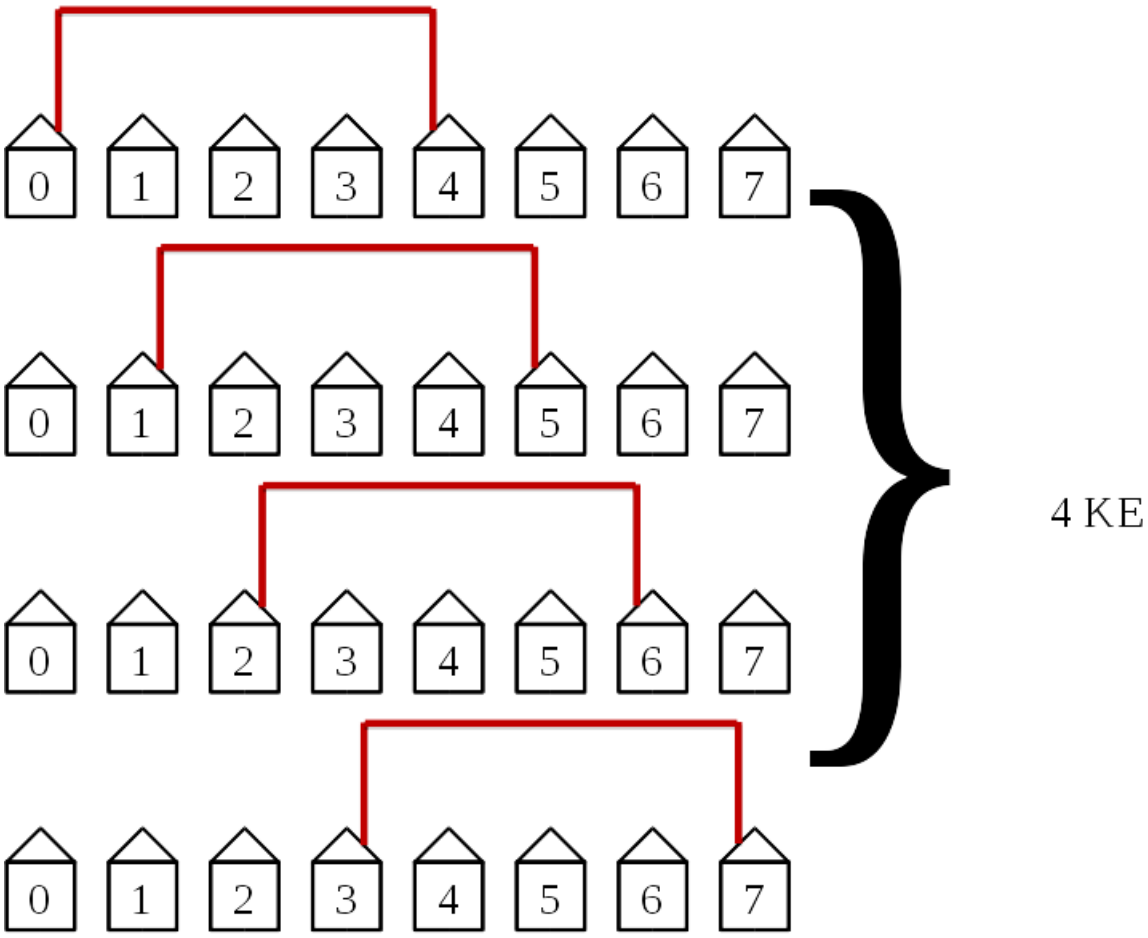

Figure 2.9: The fourth step in the protocol connects the fourth nearest neighbors. This step is the slowest and least efficient step in the protocol in our example of $N = 7$. This step requires 4 KEs to complete.

The protocol will then connect the fifth nearest neighbors as shown in Figure 2.10. This step will take 3 KEs to complete. It is also inefficient since it is beyond the midpoint thus only a single loop is possible, but it requires fewer KEs since there are only three such pairs.

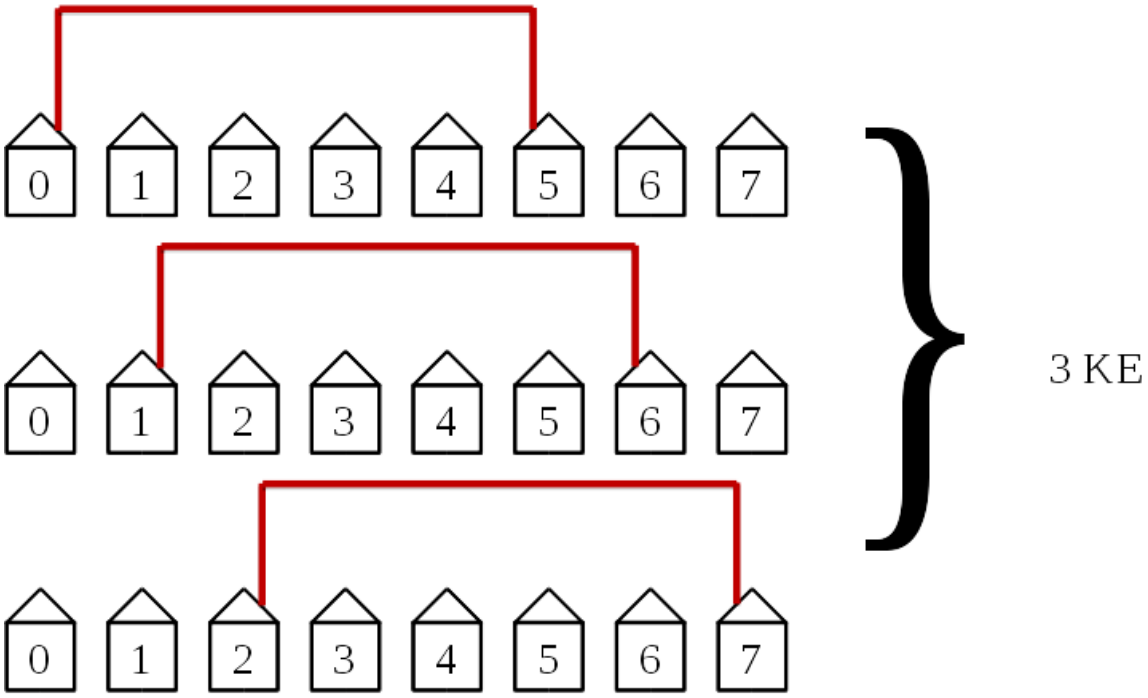

Figure 2.10: The fifth step in the protocol connects the fifth nearest neighbors. This step is not efficient since simultaneous non-overlapping loops with disconnected hosts cannot occur.

The protocol will then connect the sixth nearest neighbors as shown in Figure 2.11. This step will take 2 KEs because there are only two possibilities.

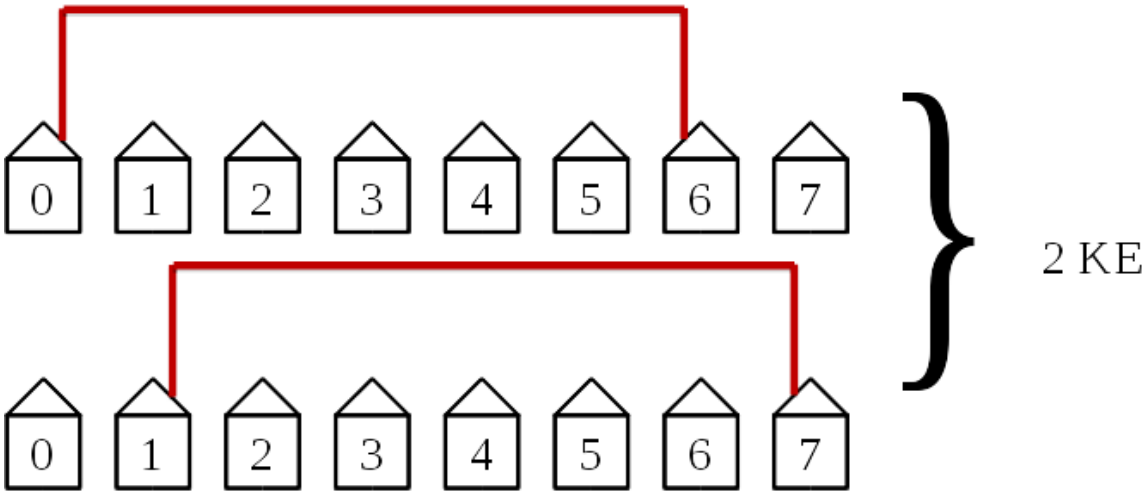

Figure 2.11: The sixth step in the protocol connects the sixth nearest neighbors. This step requires only 2 KEs since there are only two possibilities.

The protocol will then connect the seventh closest neighbors as shown in Figure 2.12. This will take 1 KE since there is only one such pair of hosts. Host 1 through 6 are not allowed access to the KLJN band thus they are in State 1. This is the seventh and the last step. This step is not efficient.

Figure 2.12: Only one key exchange is performed in this step. Host 1 through 6 are not allowed access to the KLJN band thus they are in State 1. This is the seventh and the last step. This step is not efficient but only requires one KE since there is only one such pair of hosts.

This completes the protocol for an example of size $N = 7$. Notice the pattern that occurs for N being odd. We have a pattern of 1 KE, 2 KE, 3 KE, 4 KE, 3 KE, 2 KE, and 1 KE. This is essentially Gauss's counting technique up to $N/2$ and back. The total number of KEs needed will be 1 KE + 2 KE + 3 KE + 4 KE + 3 KE + 2 KE + 1 KE = 16 KE.

The speed or time requirement of the protocol for a network of arbitrary size N with N being odd is $\left(\frac{N+1}{2}\right)^2$ KEs and can be derived as follows.

Since N is odd we can express it as equation (2.2):

$$N = 2n + 1. \tag{2.2}$$

To find the midpoint we can solve for n and express it in terms of N; this gives the following equation (2.3):

$$\frac{N-1}{2} = n. \tag{2.3}$$

The pattern when N is odd has the following form given by equation (2.4):

$$1 + 2 + \cdots + (n-1) + n + (n-1) + \cdots + 2 + 1 = \left(\frac{N-1}{2}\right)^2. \tag{2.4}$$

Expressing n in terms of N gives equation (2.5):

$$1+2+\cdots+\left(\frac{N-1}{2}-1\right)+\left(\frac{N-1}{2}\right)+\left(\frac{N-1}{2}-1\right)+\cdots+2+1 = \left(\frac{N-1}{2}\right)^2. \tag{2.5}$$

We know from Gauss's counting method that

$$1 + 2 + \cdots + N = \frac{N(N+1)}{2}.$$

In our pattern we can use Gauss's counting method twice to find the sum as follows in equation (2.6):

$$\underbrace{1+2+\cdots+\left(\frac{N-1}{2}-1\right)}_{\frac{\left(\frac{N-1}{2}-1\right)\left(\frac{N-1}{2}\right)}{2}}+\left(\frac{N-1}{2}\right)+\underbrace{\left(\frac{N-1}{2}-1\right)+\cdots+2+1}_{\frac{\left(\frac{N-1}{2}-1\right)\left(\frac{N-1}{2}\right)}{2}}=\left(\frac{N-1}{2}\right)^2. \tag{2.6}$$

Equation (2.6) simplifies to equation (2.7), given below:

$$\frac{\left(\frac{N-1}{2}-1\right)\left(\frac{N-1}{2}\right)}{2}+\left(\frac{N-1}{2}\right)+\frac{\left(\frac{N-1}{2}-1\right)\left(\frac{N-1}{2}\right)}{2}=\left(\frac{N-1}{2}\right)^2. \tag{2.7}$$

Thus the speed of the network is proportional to $N^2/4$ with N being the number of hosts in the network and odd numbered. The pattern for when N is even is similar.

2.2.2.2 If N is Even for a Network of Size N

For the sake of completeness and for those without a communications background, we will again illustrate the steps the protocol takes and calculate the time requirements with an example shown in the following figures. In this case, we have an even number as the network size is $N = 8$. We have 9 hosts with index i ($0 \leq i \leq 8$). We have 8 intermediate connections between the first and last host.

The first step in the protocol connects the nearest neighbors. This step is the quickest and most efficient. It has the most simultaneous non-overlapping loops and requires only one KE to complete. Figure 2.13 illustrates this first step in the protocol.

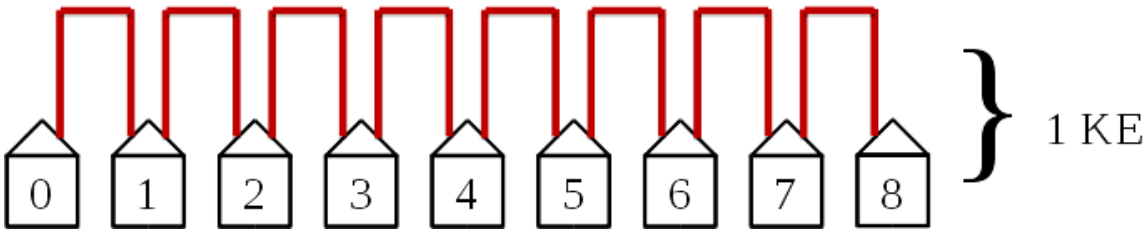

Figure 2.13: The first step in the protocol connects the nearest neighbors. This step is the quickest and most efficient. It has the most non-overlapping simultaneous loops and requires only 1 KE to complete.

The second step in the protocol will then connect the second nearest neighbors as shown in Figure 2.14. This step will take two KEs to complete and has the second most simultaneous non-overlapping loops. It is the second quickest and second most efficient step.

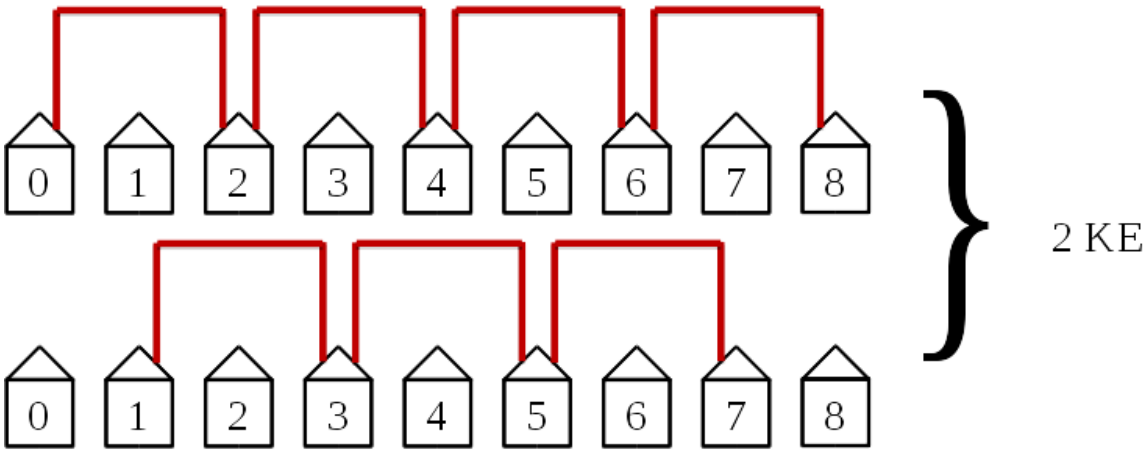

Figure 2.14: The second step in the protocol connects the second nearest neighbors. This step requires 2 KEs to complete.

The protocol will then connect the third nearest neighbors as shown in Figure 2.15. This will take 3 KEs to complete and is not as efficient as the first two

steps in the protocol but still has simultaneous loops in this example with $N = 8$.

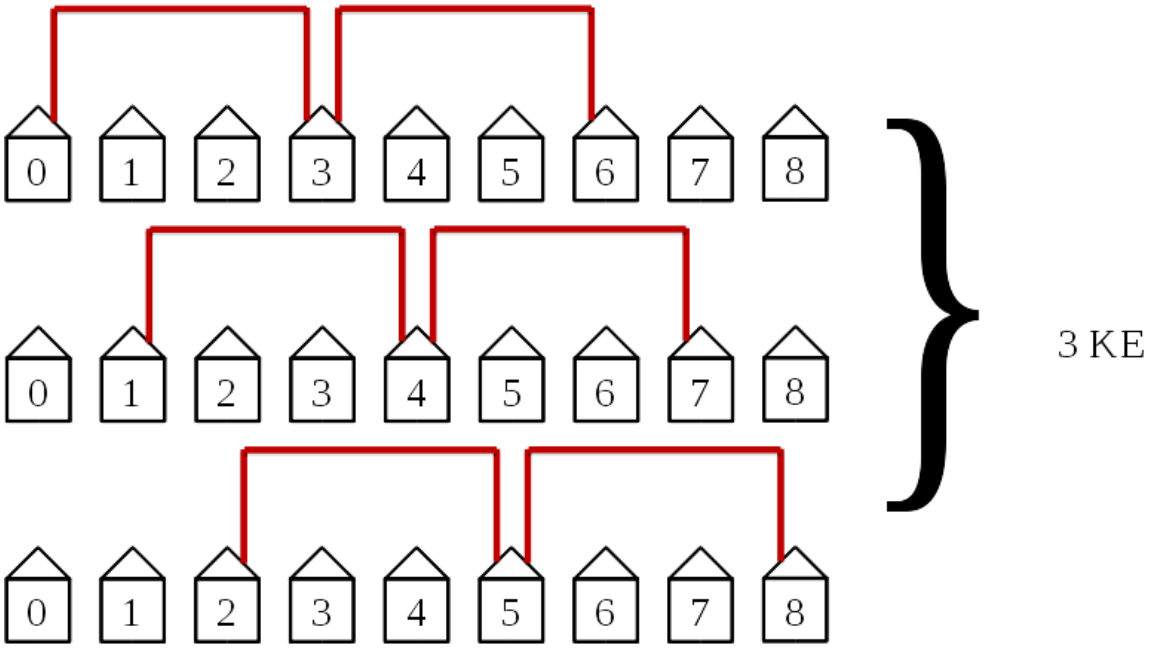

Figure 2.15: The third step in the protocol connects the third nearest neighbors. This step requires 3 KEs to complete.

The protocol will then connect the fourth nearest neighbors as shown in Figure 2.16. This is at the midpoint for our example with $N = 8$ and is the slowest and least efficient step in the protocol. The midpoint is defined when the distance between Alice and Bob is equal to half the length of the network. This step will take 4 KEs to complete. The slowest and least efficient steps occur at the midpoint of the protocol.

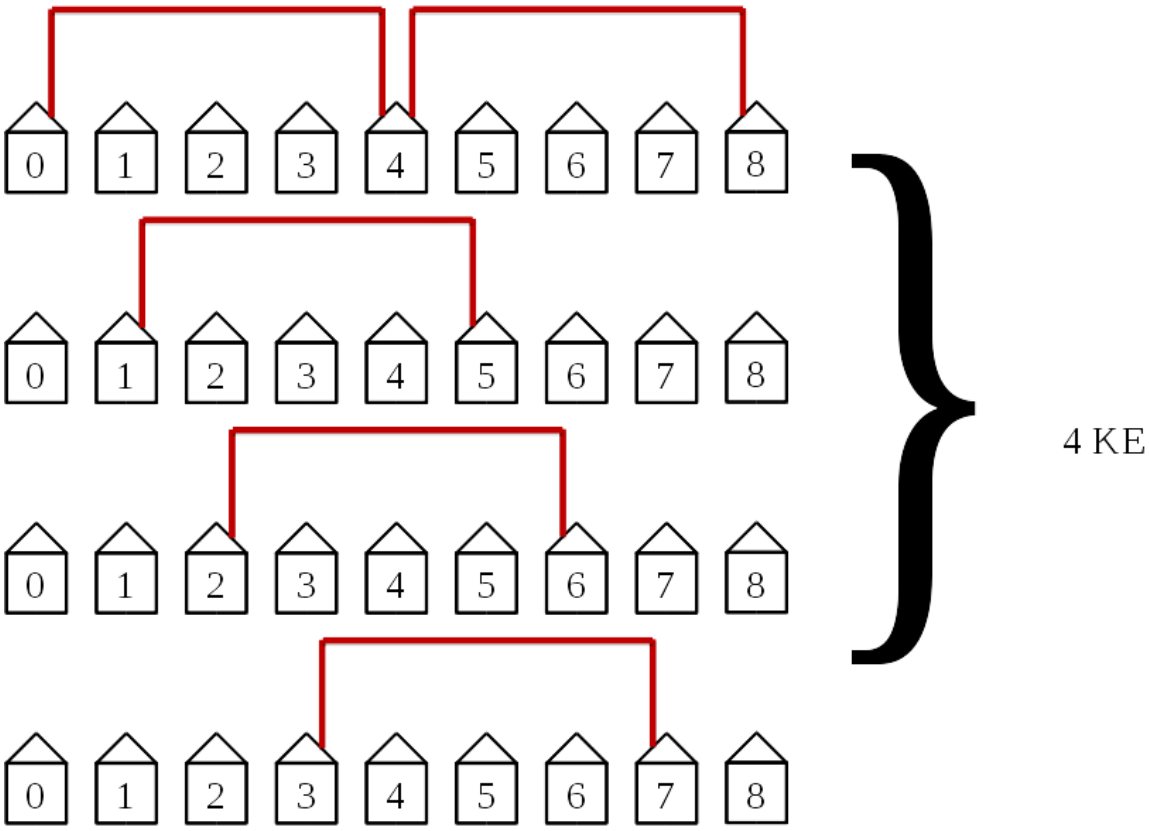

Figure 2.16: The fourth step in the protocol connects the fourth nearest neighbors. It requires 4 KEs to complete.

The protocol will then connect the fifth nearest neighbors as shown in Figure 2.17. This step will take 4 KEs to complete. It is not efficient since it is at the midpoint.

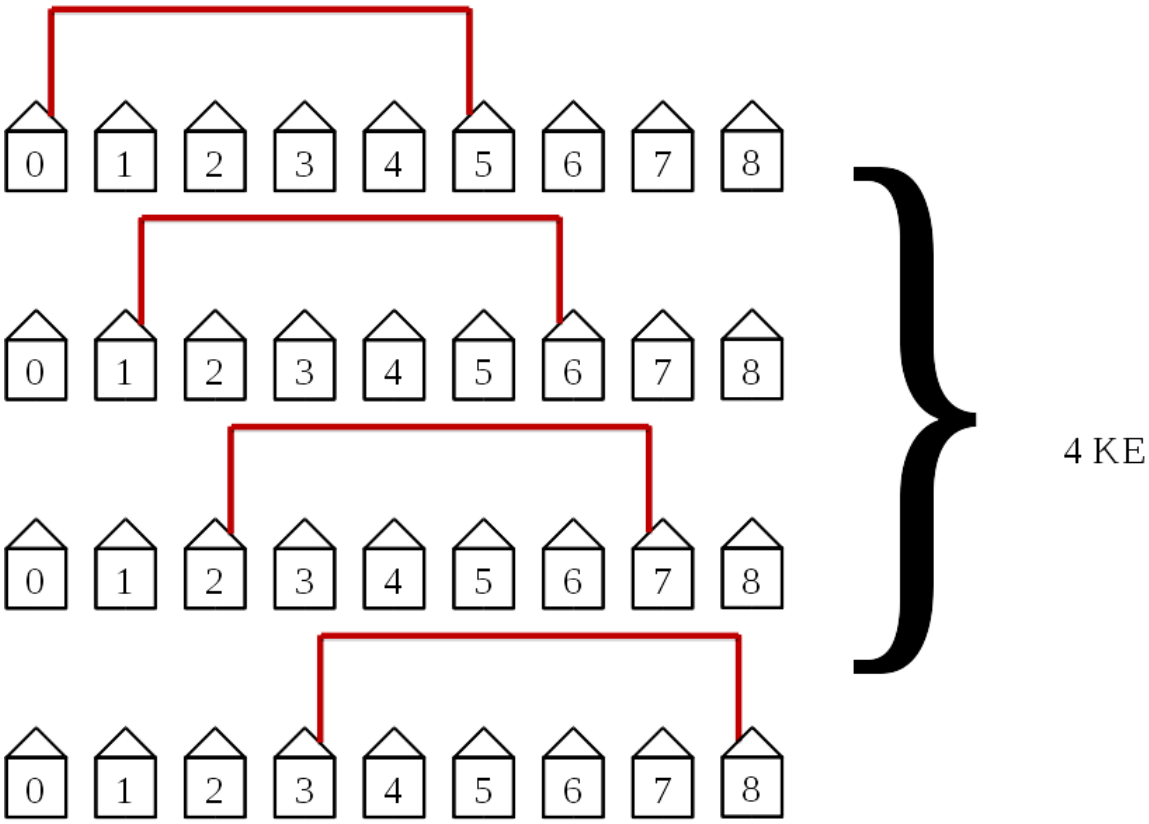

Figure 2.17: The fifth step in the protocol connects the fifth nearest neighbors. This step is not efficient since simultaneous non-overlapping loops with disconnected hosts cannot occur. It requires 4 KEs to complete.

The protocol will then connect the sixth nearest neighbors as shown in Figure 2.18. This step will take 3 KEs because there are only three possibilities at this distance in this example network of size $N = 8$.

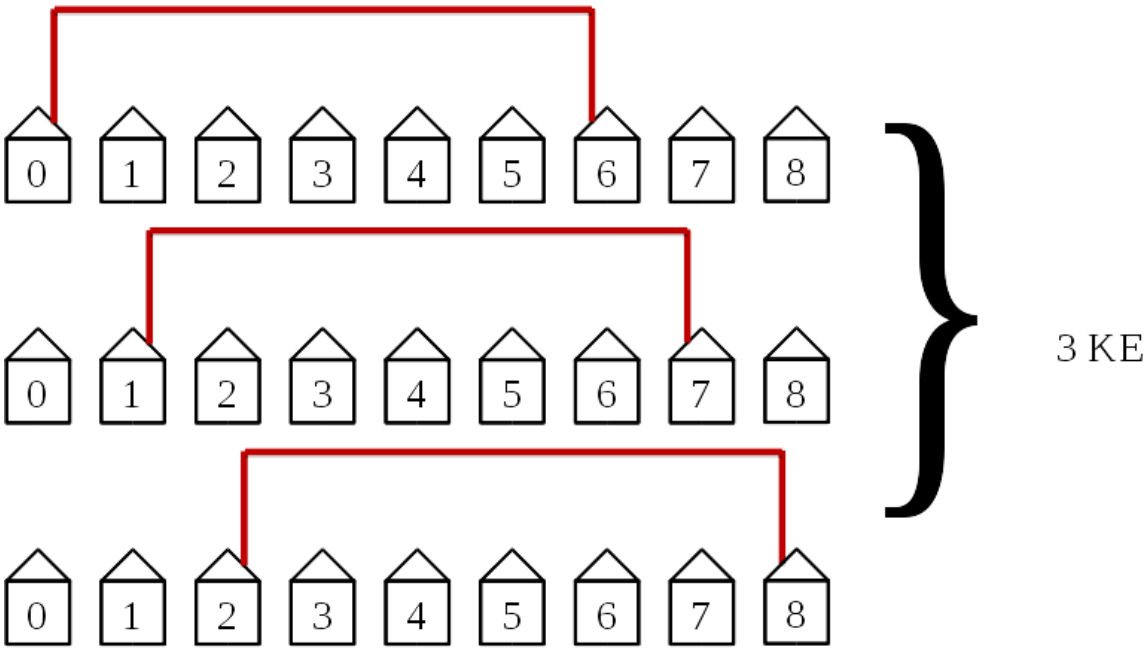

Figure 2.18: The sixth step in the protocol connects the sixth nearest neighbors. This step requires only 3 KEs since it is the third to last step and there are only three possibilities.

The protocol will then connect the seventh nearest neighbors as shown in Figure 2.19. This will take 2 KEs since there are only two pairs of hosts with a length of seven hosts between them.

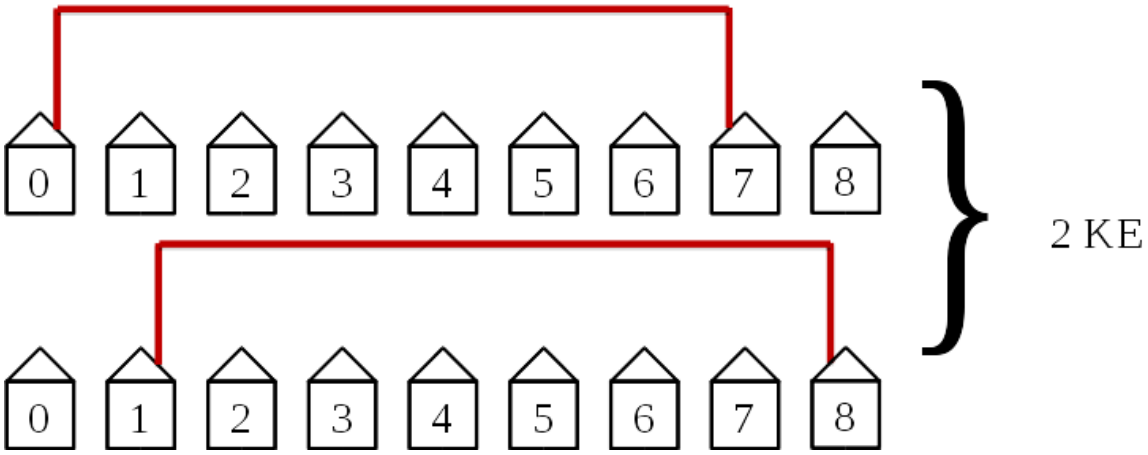

Figure 2.19: The seventh step in this example network of size $N = 8$. This step is not efficient but only requires two KEs since there are only two such pairs of hosts.

The last step in the protocol connects the first and last hosts. This step is the least efficient and requires the entire length of the network. Since there is only one pair of hosts at this length this step requires only one KE. This last step is illustrated in Figure 2.20.

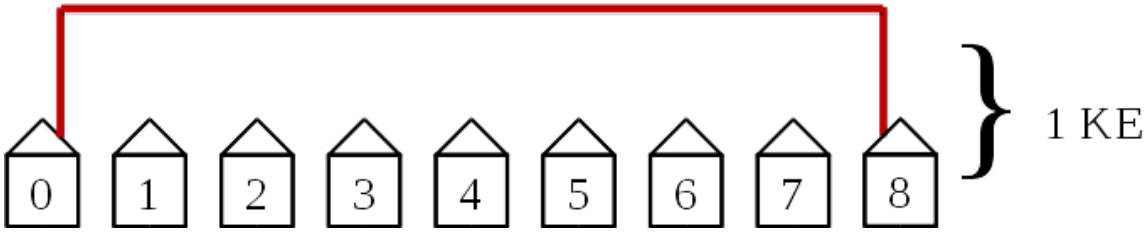

Figure 2.20: The last step in the protocol connects the first and last hosts. This step is the least efficient and requires the entire length of the network. Since there is only one pair of hosts at this length this step requires only one KE.

Notice the pattern that occurs for N being even. We have 1 KE, 2 KE, 3 KE, 4 KE, 4 KE, 3 KE, 2 KE, and 1 KE. This is essentially Gauss's counting technique up

to $N/2$ and back. The total number of KEs needed will be 1 KE + 2 KE + 3 KE + 4 KE + 4 KE + 3 KE + 2 KE + 1 KE = 20 KE. The time needed to connect the entire network will take 20 KEs which is approximately 40 seconds if B_{kljn} is 10 kHz and if the key is 100 bits long.

The speed or time requirement of the protocol for a network of size N with N being even is $\left(N^2/4 + N/2\right)$ KEs and can be derived as follows.

With $N = 8$ the number of KEs needed is 20, according to equation (2.8):

$$\frac{N^2}{4} + \frac{N}{2} = 20. \tag{2.8}$$

Since N is even we can express it as equation (2.9):

$$N = 2n. \tag{2.9}$$

To find the midpoint we can solve n and express it in terms of N; this gives equation (2.10):

$$\frac{N}{2} = n. \tag{2.10}$$

The general pattern when N is even has the following form given in equation (2.11):

$$1 + 2 + \cdots + n + n + \cdots + 2 + 1 = \frac{N^2}{4} + \frac{N}{2}. \tag{2.11}$$

Expressing n in terms of N gives equation (2.12):

$$1 + 2 + \cdots + \frac{N}{2} + \frac{N}{2} + \cdots + 2 + 1 = \frac{N^2}{4} + \frac{N}{2}. \tag{2.12}$$

We know from Gauss's counting method that

$$1 + 2 + \cdots + N = \frac{N(N+1)}{2}.$$

In our pattern we can use Gauss's counting method twice to find the sum as follows given by equations (2.13) and (2.14):

$$\underbrace{1 + 2 + \cdots + \frac{N}{2}}_{\frac{\left(\frac{N}{2}\right)\left(\frac{N}{2}+1\right)}{2}} + \underbrace{\frac{N}{2} + \cdots + 2 + 1}_{\frac{\left(\frac{N}{2}\right)\left(\frac{N}{2}+1\right)}{2}} = \frac{N^2}{4} + \frac{N}{2}, \tag{2.13}$$

$$\frac{\frac{N}{2}\left(\frac{N}{2}+1\right)}{2} + \frac{\frac{N}{2}\left(\frac{N}{2}+1\right)}{2} = \frac{N^2}{4} + \frac{N}{2}. \tag{2.14}$$

Equation (2.14) simplifies to equation (2.15):

$$\left(\frac{N}{2}\right)\left(\frac{N}{2}+1\right) = \frac{N^2}{4} + \frac{N}{2}. \tag{2.15}$$

Thus the speed of the network is proportional to $N^2/4$ with N being the number of hosts on the network, and with N being even.

2.3 Limitations of Realizing KLJN over the Smart Grid, Open Questions, and Future Work

To fully implement the KLJN key exchange protocol over the smart grid will require solutions to further engineering problems. This chapter presents results of our early work, which focuses on the system-concept in a one-dimensional network. Some of the limitations, open questions, and future work are discussed below.

2.3.1 Limitations

The main limitation of the KLJN protocol is that it is a P2P network. This will limit the number of simultaneous KLJN key exchanges a host can have. Since

overlapping loops are not allowed, the time required scales quadratically with the number of hosts in a linear chain network. Another limit is that the KLJN bandwidth is dependent on the distance between Alice and Bob and slows down for longer distances. These limitations make it impractical to connect millions of hosts via a linear chain network, and thus other topologies (and perhaps bridges, routers, or repeaters) will be needed to connect such chains with each other. Practical limitations in the power system, such as tap changing transformers and other devices may also require bridges to couple the KLJN signal around these devices.

2.3.2 Open Questions

The related technical challenges need to be further researched. For example, distribution transformers can shield most of the signals sent from one phase on the load side; this will present a problem, but there are many ways to get around it and accurately transmit the KLJN band. We did not investigate the problems of phase-correcting inductors and capacitors since they are separated by the power filters from the KLJN band. Research and development will be needed for some of these problems including how to setup filters in each node. Accuracies are typically within a few percentage points. In the experimental demo, the cable resistance was 2% of total loop resistance. In practice, the impedance of the power grid would need to be taken into account.

2.3.3 Future Work

Future work will, among others, include protocols for several other power grid topologies. Setting up filters on the power grid and implementing all the filters will also need to be further researched. Penetration hacking attacks against filters and defensively securing the filters are also interesting open problems.

In the next chapter we will analyze the star network and compare its cost com-

plexities and robustness with linear chain networks and fully connected networks.

3. A PROTOCOL FOR IMPLEMENTING UNCONDITIONALLY SECURE KEY EXCHANGE ON A STAR NETWORK IN THE SMART GRID AND COMPARING COST COMPLEXITIES AND ROBUSTNESS WITH DIFFERENT NETWORK TOPOLOGIES[2]

3.1 Securing Networks

3.1.1 Motivation for a Secure Network

In the advent of intelligent vehicle information networks [11], the smart power grid [5], and the Internet of Things (IoT) [94], current infrastructure is becoming increasingly dependent on cyber networks. This dependency makes current infrastructure a larger, more attractive target for cyberattacks, such that the National Security Agency (NSA) director stated the U.S. power grid could be shut down with a cyberattack [1].

3.1.2 Secure Key Exchange over P2P Networks and the Fully Connected Network

Hardware-based key exchanges require P2P networks with a dedicated connection to each host. For very large networks this will be costly due to the infrastructure (cables) and key exchangers. The cost complexity of the growth for different networks can be denoted by $T_{\text{cable}}(N)$ for the number of cables needed, $T_{\text{ke}}(N)$ for the number of key exchangers needed, and $T_{\text{time}}(N)$ for the amount of time required or speed to complete a secure bit exchange, with N representing the number of hosts on the network.

2 Part of this chapter is reprinted with permission from "Resource Requirements and Speed *versus* Geometry of Unconditionally Secure Physical Key Exchanges" by Gonzalez, E., Balog, R.S., Kish, L.B., (2015). *Entropy*, 17(4), pp. 2010-2024; DOI:10.3390/e17042010 Copyright 2015 by MDPI

A simple method to construct P2P networks is a fully connected network also known as the complete graph in graph theory. The fully connected network is illustrated in Figure 3.1. The fully connected network does not require a protocol since every host in the network has a dedicated connection with every other host in the network, and can process a secure bit exchange with any other host at any time simultaneously. This network has $N-1$ key exchangers per host and scales on the order of N^2 for cables and key exchangers, which makes this network impractical for very large networks. The cost complexities for the fully connected network are $T_{\text{cable}}(N) \in O(N^2)$, $T_{\text{ke}}(N) \in O(N^2)$, and $T_{\text{time}}(N) \in O(1)$. We will denote the fully connected network with $N-1$ key exchangers per host as FCN_{N-1}. The fully connected network has $N-1$ key exchangers for every host resulting in $(N-1) \cdot N$ total key exchangers for the entire network, $N-1$ direct connections for every host resulting in $(N-1) \cdot N/2$ total cables for the entire network. The advantage the fully connected network has is time, as every host in the network can simultaneously process a secure bit exchange with every other host in the network.

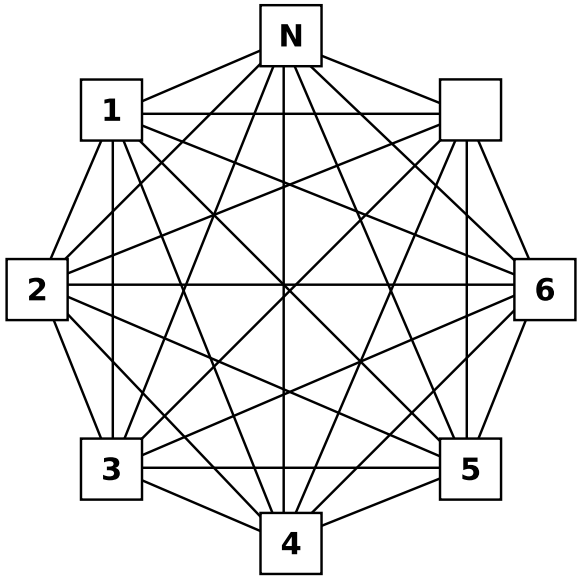

Figure 3.1: An illustration of a fully connected network with $N-1$ communicators per host (denoted as FCN_{N-1}) has cost complexities of $T_{\mathrm{cable}}(N) \in O(N^2)$, $T_{\mathrm{ke}}(N) \in O(N^2)$, and $T_{\mathrm{time}}(N) \in O(1)$.

If the cost of having $(N-1) \cdot N$ key exchangers for the entire network is too costly, then a trade-off between the number of key exchangers and speed might be preferable. If there is only one key exchanger per host in the fully connected network, then the cost complexities for the fully connected network will be $T_{\mathrm{cable}}(N) \in O(N^2)$, $T_{\mathrm{ke}}(N) \in O(N)$, and $T_{\mathrm{time}}(N) \in O(N)$, and will require a protocol which we will denote as FCN_1 to process a secure bit exchange with every host in the network.

The fully connected network is robust and reliable as it does not depend on a single cable or key exchanger. If there is cable destruction or a damaged key exchanger then only the hosts connected by that cable or key exchanger will be affected, and only that connection will be affected. The affected hosts will still be able to process a secure bit exchange with other hosts which do not depend on the damaged cable or key exchanger.

To add additional hosts to the fully connected network will be trivial since it does not have a protocol. In the case of FCN_1 the protocol will need to consider the added host.

3.1.3 Linear Chain Network with Two Key Exchangers per Host

Linear chain networks, also known as bus networks or daisy chain networks, contain a single line and two key exchanges per host as illustrated in Figure 3.2, and were analyzed in the contexts of smart grids in [32] and in the previous chapter. The linear chain network with 2 key exchangers per host has cost complexities of $T_{\text{cable}}(N) \in O(N)$, $T_{\text{ke}}(N) \in O(N)$, and $T_{\text{time}}(N) \in O(N^2)$. By having 2 key exchanges per host the linear chain network can process 2 simultaneous secure bit exchanges as long as one host is downstream, for example host $i-a$ for any positive integer a, and the other host is upstream, for example host $i+b$ for any positive integer b of the ith host. The first host and the last host are special cases which cannot have simultaneous secure key exchanges with other hosts [32, 31].

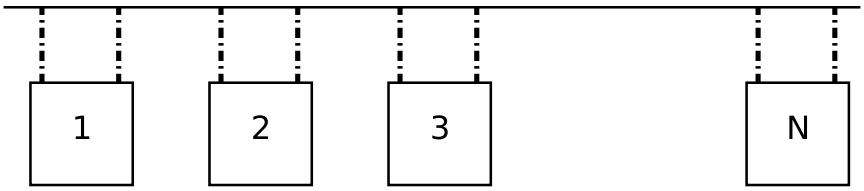

Figure 3.2: An illustration of a linear chain network with 2 key exchangers per host has cost complexities of $T_{\text{cable}}(N) \in O(N)$, $T_{\text{ke}}(N) \in O(N)$, and $T_{\text{time}}(N) \in O(N^2)$.

The reliability of the linear chain network is dependent on the cable. If there is damage to the cable then the network will become two different networks divided at the location of the damaged cable, and the two networks cannot process a secure

bit exchange with each other. The linear chain network is more robust if there is damage to a key exchanger; then only the host with the damaged key exchanger will be slowed down but it will be able connect with all other hosts on the network since there are two key exchangers per host.

If an additional host joins the network with N hosts, then the protocol will consider $N + 1$ hosts instead of N; this will be a relatively simple fix as the the protocol can be preprogrammed in the hosts for any N.

3.2 Results and Discussion

3.2.1 Star Network

The star network is a hub and spoke topology with a center switch like an old telephone exchange switch system and has branches connected to the center. We denote the star network protocol with one key exchanger per host as STAR. The cost complexities of the star network are $T_{\text{cable}}(N) \in O(N)$, $T_{\text{ke}}(N) \in O(N)$, and $T_{\text{time}}(N) \in O(N)$. Figure 3.3 is an example of a star network with N branches.

The most efficient protocol in the star network is similar to the protocol in the linear chain network in regards to first connecting to the nearest neighbors, then connecting the second nearest neighbors, and so on. The star network allows for faster speed than the linear chain network with similar cable and hardware cost complexities.

3.2.2 Graph Theory and Previous Work on the Star Network

In graph theory, the hosts are considered vertices and the cables are considered edges [97]. The protocol of the star network is to connect every host in the network to process a secure bit exchange with every other host on the network in the least number of Secure Bit Exchange Period (SBEP) steps. In graph theory, the star network protocol can be described as a special case of an edge-color problem [93] known

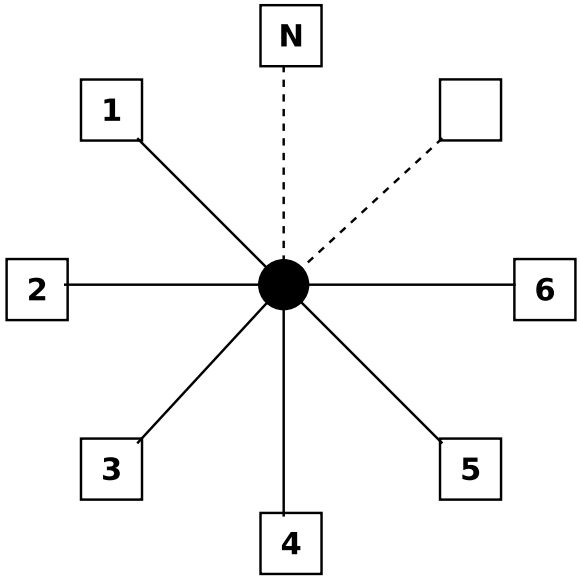

Figure 3.3: An illustration of a star network system with one key exchanger per host has cost complexities of $T_{\text{cable}}(N) \in O(N)$, $T_{\text{ke}}(N) \in O(N)$, and $T_{\text{time}}(N) \in O(N)$.

as round-robin (RR) tournament or all-play-all tournament problem [65]. The number, k, of edge colors needed in graph theory is the number of SBEPs needed in the star network protocol, although many geometric structures and edge-color problems have been studied in graph theory [95, 96, 3, 9, 76, 37, 34] and applied to various infrastructure networks [38, 87, 2, 15], they have not been applied to P2P hardware-based secure key exchange networks other than [32]. Many network applications assume overlapping signals in the same channel are possible and do not have a dedicated channel in which every vertex connects with every other vertex. For QKD and KLJN network applications, these networks require dedicated communication channels with no overlapping signals, and RR solutions to different geometric structures. The star network protocol presented in Section 3.2.3 is specifically for QKD and KLJN networks, and is significant since it combines residual SBEP steps whenever possible, thus lowering the total number of SBEPs needed. After a thorough

literature review a similar RR solution was not found and the most similar solution found was [3].

3.2.3 Protocol and Analysis of the Star Network

For a network with N hosts, the star key exchange network protocol begins with every odd numbered host, say ith host with i being odd, and processes a secure bit exchange with their nearest upstream neighbor, that is host i+1, this will take one Secure Bit Exchange Period (SBEP) and the secure key exchange between different hosts will occur simultaneously. For example, host 1 will process a secure bit exchange with host 2, while host 3 will process a secure bit exchange with host 4, while host $N-1$ will process a secure bit exchange with host N if N is even, or host $N-2$ will process a secure bit exchange with host $N-1$ if N is odd. If N is odd, then the last host, that is, host N, will not process a secure bit exchange in the first SBEP step. The next step in the protocol is for every even numbered host, say ith host with i being even, to process a secure bit exchange with their nearest upstream neighbor, say host i+1, simultaneously. For example, host 2 will process a secure bit exchange with host 3, while host 4 will process a secure bit exchange with host 5, while host $N-1$ will process a secure bit exchange with host N if N is even, or host N will process a secure bit exchange with host 1 if N is odd, note that the protocol will wrap around from the last host N to the first host 1. The circular nature of the star network is a reason why it is faster than the linear chain network with similar cable and hardware complexities. The star network protocol STAR then continues with every odd host to process a secure bit exchange with their upstream second nearest neighbor, that is every ith host with i being odd, with host i+2, then the even numbered hosts will process a secure bit exchange with their second nearest neighbor, say every ith host with i being even with host i+2. The

protocol continues by having every host process a secure bit exchange with their third nearest neighbors, then fourth nearest neighbors, and continues until every host in the network has processed a secure bit exchange with every other host.

As an example, Figure 3.4 illustrates every step of the protocol for a STAR network with 5 hosts. The first SBEP step in the protocol is illustrated in sub-figure 3.4a, note how every odd numbered host, i, has a secure bit exchange with their next upstream nearest neighbor host i+1. The second SBEP step in the protocol is illustrated in sub-figure 3.4b, note how every even numbered host, i, has a secure bit exchange with their next upstream nearest neighbor host i+1. The third SBEP step in the protocol is illustrated in sub-figure 3.4c. Since the number of hosts in the network is odd, it will take additional SBEP steps to process a secure bit exchange with these remaining hosts; these are residual SBEP steps. Note how the last host wraps around to the first host. The fourth SBEP step in the protocol is illustrated in sub-figure 3.4d. In this SBEP step, every odd numbered host, i, has a secure bit exchange with their second upstream nearest neighbor host i+2. The fifth SBEP step in the protocol is illustrated in sub-figure 3.4e, this step is similar to step 4, except that now the even numbered hosts process a secure bit exchange with their second upstream nearest neighbors. The sixth and last SBEP step in the protocol is illustrated in sub-figure 3.4f. Since N is odd, the protocol requires additional residual SBEP steps to process a secure bit exchange with the remaining hosts. Note that this example of the STAR protocol with $N = 5$ hosts requires six SBEP steps for every host in the network to process a secure bit exchange with every other host. Table 3.1 demonstrates what every host is doing at every step in the protocol of this example as illustrated in Figure 3.4. Table 3.2 is the legend for Table 3.1. The arrow symbol "$\rightarrow$" is used as $x \rightarrow y$, meaning host x is processing a secure bit exchange with host y. The star symbol "$\bigstar$" means the host of this row is being utilized. The

circle symbol "◯" means the host of this row is not active.

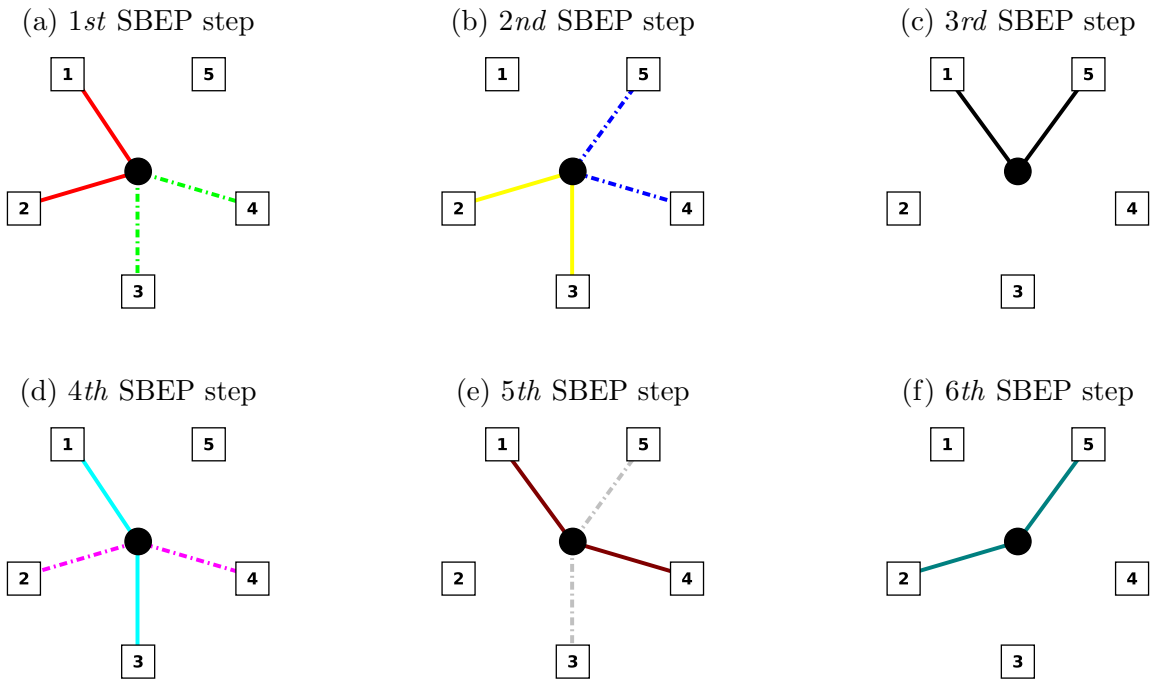

Figure 3.4: An illustration of an example of the STAR network protocol for a network with five hosts. It takes six Secure Bit Exchange Period (SBEP) steps for every host in the network to process a secure bit exchange with every other host.

Host	Fig. 3.4a	Fig. 3.4b	Fig. 3.4c	Fig. 3.4d	Fig. 3.4e	Fig. 3.4f
1	$1 \to 2$	◯	★	$1 \to 3$	★	◯
2	★	$2 \to 3$	◯	$2 \to 4$	◯	★
3	$3 \to 4$	★	◯	★	$3 \to 5$	◯
4	★	$4 \to 5$	◯	★	$4 \to 1$	◯
5	◯	★	$5 \to 1$	◯	★	$5 \to 2$

Table 3.1: This table demonstrates what every host is doing at every SBEP step in the STAR protocol as described in the example and illustrated in Figure 3.4.

Symbol	Meaning of symbols in Table 3.1
$x \to y$	Host x processing a secure bit exchange with host y.
★	Host of this row is being utilized.
◯	Host of this row is inactive.

Table 3.2: This table is the legend of Table 3.1.

The number of SBEPs needed in the STAR protocol is dependent on the number of hosts, N, in the network. Table 3.3 shows the number of SBEPs needed in the star network for every host to process a secure bit exchange with every other host in the network, for star networks with up to 20 hosts. Figure 3.5 is the plot of Table 3.3, with N being the independent variable and SBEP being the dependent variable. The linear regression line is $f(N) = 1.3192982456 \cdot N - 1.301754386$, and the coefficient of determination is $R^2 = 0.988989157$.

N, number of hosts in star network	SBEP(N), number of SBEP steps needed for a network with N hosts
2	1
3	3
4	3
5	6
6	6
7	8
8	8
9	12
10	12
11	14
12	14
13	17
14	17
15	19
16	19
17	22
18	22
19	24
20	24

Table 3.3: This table shows the number of SBEPs needed in star networks with 2 hosts to 20 hosts, for every host in the network to execute a secure bit exchange with every other host.

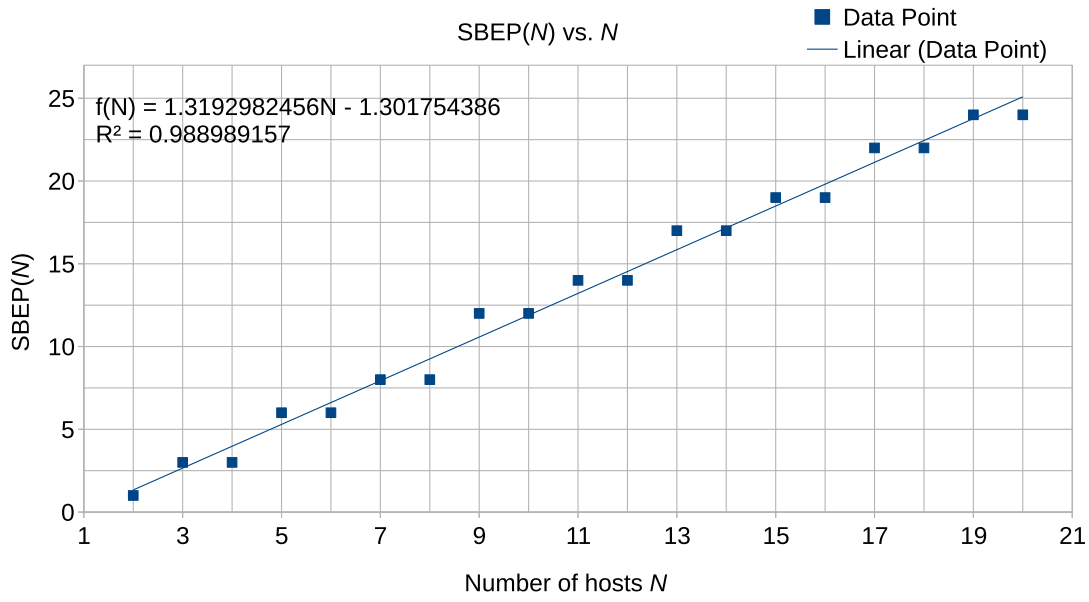

Figure 3.5: This is the plot of Table 3.3. The data points are plotted along with a linear regression line which is $f(N) = 1.3192982456 \cdot N - 1.301754386$, and the coefficient of determination is $R^2 = 0.988989157$. The horizontal axis (independent variable) is N, meaning the number of hosts in the star network. The vertical axis (dependent variable) is SBEP(N), meaning the number of SBEP steps needed for a network with N hosts.

The patterns and relations in the star network protocol can be seen in Table 3.3 and Figure 3.5. Note that when N is evenly divisible by 2 it will take exactly 2 SBEP steps for every host, i, to process a secure bit exchange with their nearest neighbor host i+1. If N is not evenly divisible by 2 then it will take exactly 3 SBEP steps for every host i to process a secure bit exchange with their nearest neighbor host i+1. The results are the same for every case when N is divided by 3, 4, 5, ..., $(N-1)/2$, and every host i processes a secure bit exchange with their second, third, fourth, ..., $(N-2)/2th$ nearest neighbor, that is host i+2, i+3, i+4, ..., i+$(N-2)$ respectively. There is a unique case when N is even and is divided by $N/2$, in this case only one SBEP step is needed to process a secure bit exchange. The residual

steps are combined whenever possible. For example, in the case when $N = 7$, the $6th$ and $9th$ steps can be combined into one step, resulting in one less SBEP step. These patterns and relations were used to conceive equations (3.1a) through (3.1d), where the "$\lceil\rceil$" symbol in the equations is the ceiling function, N is the number of hosts, and SBEP(N) is the number of SBEPs needed to share an independent secure bit for each possible pair formed in the network, which means each host share $N-1$ secure bits. (Note, after this sharing each possible pair formed in the network has only a single bit of their respective secure key. Thus to share a key with k bits, the above process must be repeated k times.)

$$\text{SBEP}(N) = N + \left\lceil \frac{N}{4} \right\rceil - 2 \text{ for } N \leq 8 \text{ and } N \text{ is even.} \tag{3.1a}$$

$$\text{SBEP}(N) = N + \left\lceil \frac{N}{4} \right\rceil - 1 \text{ for } N \leq 8 \text{ and } N \text{ is odd.} \tag{3.1b}$$

$$\text{SBEP}(N) = N + \left\lceil \frac{N}{4} \right\rceil - 1 \text{ for } N > 8 \text{ and } N \text{ is even.} \tag{3.1c}$$

$$\text{SBEP}(N) = N + \left\lceil \frac{N}{4} \right\rceil \text{ for } N > 8 \text{ and } N \text{ is odd.} \tag{3.1d}$$

The reliability of the star network is dependent on its center switch, cables, and key exchangers. One could sabotage the entire network just by damaging the center switch in the star network. If a cable or key exchanger is damaged in the star network, then the affected host will be effectively disconnected from the entire network, but the unaffected hosts will be able to continue processing a secure bit exchange with other hosts in the network.

To add additional hosts in the star network will require every host in the network to change the protocol from N to $N + 1$, which is a relatively simple process since the protocols can be preprogrammed in the hosts.

The star network could be utilized in many situations including vehicle information networks [88, 13] and inside equipment with components spread around a central processing unit such as a computer.

3.2.4 Comparing Network Topologies

Table 3.4 compares cost complexities of the fully connected network with $N-1$ key exchangers per host denoted by FCN_{N-1}, the fully connected network with 1 key exchanger per host denoted by FCN_1, the linear chain network protocol with 2 key exchangers per host is denoted by LCH, and the star network protocol with 1 communicator per host denoted by STAR. As can be seen from Table 3.4, the fastest network is the FCN_{N-1} network, the networks with the least cost (lowest cost complexities) of cables are the linear chain network and the star network, and the networks with the least cost (lowest cost complexities) of key exchangers are FCN_1, linear chain network, and star network. These results will hold for both KLJN and QKD systems. These results show that the star network has better performance than the linear chain network with similar cost complexities for cables and key exchangers.

Network topology	$T_{\text{cable}}(N)$	$T_{\text{ke}}(N)$	$T_{\text{time}}(N)$
FCN_{N-1}	$O(N^2)$	$O(N^2)$	$O(1)$
FCN_1	$O(N^2)$	$O(N)$	$O(N)$
LCH	$O(N)$	$O(N)$	$O(N^2)$
STAR	$O(N)$	$O(N)$	$O(N)$

Table 3.4: This table summarizes the cost complexities of the fully connected networks FCN_{N-1}, FCN_1, the linear chain network protocol LCH, and the star network protocol STAR.

The robustness and reliability of each network is dependent on its geometric

topology. If a cable is damaged, then it is best to have a FCN_{N-1} network since only one connection between two hosts will be lost. In the linear chain network the entire network will be divided. In the star network, the affected host will be completely disconnected from the network. If a key exchanger is damaged, then it is best to have a linear chain network since the only consequences will be a slower secure bit exchange process, but every host will still be able to process a secure bit exchange with every other host. In the FCN_{N-1} network, a damaged key exchanger will only affect one connection between two hosts. In the star network, a damaged key exchanger will completely disconnect the affected host from the entire network. Another vulnerability of the star network is the center switch; if the center switch is damaged then the entire network is disconnected. Based on these three networks, one can argue that the most robust, reliable network is the FCN_{N-1} followed by the linear chain network, and the least robust network of these three would be the star network.

To add hosts to the FCN_{N-1} network would be trivial since the FCN_{N-1} does not need a protocol; all that is needed is to connect the host to every other host. To add hosts to the linear chain network and the star network will require every host in the network to change the protocol from N hosts to $N+1$ hosts; this will be a relatively simple process as every host can be preprogrammed.

3.2.5 Open Questions and Future Studies

The star network has cost complexity of $O(N)$ for the number of cables, key exchangers, and time, but there are still numerous other geometric network topologies that have not been explored that might benefit KLJN and QKD systems. Other examples for possible networks include matrix networks, that is, a grid of several vertical lines and horizontal lines. The matrix network might be a good model for an

urban city with squared blocks. A wheel network is another possibility that might outperform the star network. A wheel network is similar to a star network but with a connecting loop around the branches. A web network is another interesting network similar to the wheel network but with concentric circles connecting the inner branches. A web network is similar to a spider web with each node being a host. A cube network is another interesting possibility that could be utilized in a skyscraper. A cube network is similar to the matrix network except that it has three dimensions. A sphere network might be another interesting three-dimensional network that can be compared with the cube network.

Since different geometrical topologies give different trade-offs, another interest is to explore the trade-offs of the different networks, and why it is preferable to sacrifice speed, communicators, or key exchangers for infrastructure and vice versa. Another possible interest is to analyze and compare every geometric network with different number of communicators and how well they scale with speed. Another possibility is to combine several of these networks into one network and analyze its performance; in graph theory this is known as hybrid networks.

Different geometric network structures have different vulnerabilities; an analysis of each network's vulnerabilities, robustness, reliability, and different kinds of attacks would be interesting to explore and compare.

In the next chapter we will explore wireless networks and propose a key exchange trust evaluation since KLJN is not possible on wireless networks.

4. EVALUATING KEY EXCHANGE TRUST IN SENSOR NETWORKS WITH CONSIDERATION OF UNCONDITIONALLY SECURE KEY EXCHANGE[3]

4.1 Sensor Networks, Security Concerns, Trust Mechanisms, and Unconditionally Secure Key Exchanges.

4.1.1 Sensor Networks

Sensor networks consist of sensors that measure and provide information in remote or spatially distributed areas [20]. With the advancement of miniaturization and wireless technologies, the ubiquity of sensor networks is becoming more prevalent. The benefits of having smaller dies in semiconductors include: physically smaller devices, increased ratio of computing power per energy, better battery life, etc. A few examples that utilize sensor networks include military, health care, environment monitoring, agriculture, etc.

Sensors are often required to be autonomous, decentralized, and in remote areas. Such requirements place limitations on sensors and sensor networks, including low power, limited memory and data storage, physical size, limited communication bandwidth, cost, privacy, and security [84, 80, 4]. There are proposed solutions for some of these limitations such as energy harvesting, low-power processors, smaller memory footprint, etc. However, security is a pressing issue since sensors face unique challenges. Without proper security the entire sensor network can be compromised

3 Part of this chapter is reprinted with permission from "Key Exchange Trust Evaluation in Peer-to-Peer Sensor Networks With Unconditionally Secure Key Exchange" by Gonzalez, E., Kish, L.B., (2016). A print and electronic version of this article published in *Fluctuation and Noise Letters*, Vol. 15, No. 1, 2016, pp. 165008 (17 pages) DOI:10.1142/S0219477516500085

and sabotaged.

4.1.2 Security Concerns

Limited computing power in sensors restrict them from utilizing large, complex encryption algorithms; also with limited memory and data storage, the secure key cannot be too large. Another security issue facing sensors is that the installation of optical fiber or wire connections is often not economical. Thus they are often accessible only by wireless communication, which is restricted to work with conditionally secure key exchange, which makes them vulnerable to packet capture, sniffing, and injection [83, 92, 21, 22, 62, 64, 35, 39]. In an attempt to mitigate some of these security issues, there have been several proposals to secure sensor networks, which include defenses against specific attacks and more efficient protocols [83, 92, 21, 22, 62, 64, 35, 39].

Sensor networks require data confidentiality, data integrity, data freshness, availability, self-organization, time synchronization, authentication, secure broadcasting and multicasting, and sensor privacy. Attacks on sensor networks include Distributed Denial of Service (DDoS) attacks, Sybil attacks, traffic analysis attacks, information flooding attacks, and node replication attacks [83, 92, 21, 22, 62, 64, 35, 39]. Defensive measures against some of these attacks are key establishment, key encryption, policy-based approaches, intrusion detection, and trust management. There have been several approaches for managing trust in sensor networks; the approach to trust management is based on the sensor network's trust mechanism.

4.1.3 Trust Mechanisms

Trust theory has different applications and perspectives, and the concept of trust has been associated with past behaviors and/or reputation from trusted peers [10, 33, 43, 7, 19, 85]. The notion of trust has been specified by trust definitions, trust

characteristics, and trust values [99]. Trust values have been measured by several different methodologies such as: Bayesian models [61], Beta distribution systems [41], subjective logic models [42], entropy models [14], fuzzy models [12], and game theory models [46]. However, these trust value models are not able to distinguish between conditional and unconditionally secure key exchanges, and thus need to be expanded for related applications.

Rather than expanding former models, we propose a new key exchange trust evaluation model, which takes into account the type of key exchange (conditional or unconditional) between two sensors.

Utilizing KLJN in sensor networks could significantly increase the security level in sensor networks due to its unconditionally secure key exchange.

4.1.4 Motivation for a Key Exchange Trust Evaluation

Current trust measures for sensor networks do not utilize unconditionally secure key exchange. Trust is a belief that may change over time, and is usually based on past behaviors and/or reputation from a community. Many sensor networks measure trust based on past behaviors and/or reputation [10, 33, 43, 7, 19, 85], but there has not been a trust measurement that considers the class (conditionally/unconditionally secure) of the key exchange utilized in their measurement of trust. We propose a new key exchange trust system that considers the class of the key exchange. The system utilizes the **G**eometric series to evaluate the key exchange trust, thus we call it the G key exchange trust function.

4.2 Outline of Combined Wired and Wireless Sensor Networks

In this chapter we consider P2P networks only. In such a network it will be impractical to have direct wired connections from every sensor to every other sensor, thus we propose to use both wired and wireless communication channels, and form

a wired-wireless hybrid network. The wired sensors can be utilized in areas where other sensors are in close proximity. Each sensor can then be ranked based on its key exchange and the number of key exchanges with trusted peers. We therefore propose the G key exchange trust measure system.

4.2.1 Network

The wired-wireless network will require sensors to have at least two communication devices, one for wireless and one or more for wired. A cable will also be required and can have either one or two wires inside. One wire will be for the key exchange, and the other optional wire can be utilized as a data communication channel.

Figure 4.1 is an illustration and example of the proposed wired-wireless hybrid sensor network with ten sensors. In this example, sensors A through G utilize both wired and wireless communication channels, and sensors H through J utilize only its wireless communication channel. Sensors A and B have a direct connection with the base station, and thus they can have an unconditionally secure key exchange with the operator. Note how sensor E has two wired connections; this sensor will require two KLJN communicators. Sensors C, F, and G have only one wired connection and will require one KLJN communicator. Sensors A, B, and D have three wired connections, and will require three KLJN communicators. Sensors H through J only use their wireless communication channel; these sensors are the most vulnerable to attacks and thus have a low key exchange trust value. Table 4.1 lists every sensor's key exchange with all sensors in the network of Figure 4.1, e.g., sensor A has a KLJN key exchange with sensors B and D, thus we denote this in set notation as $\mathrm{A_{kljn}} = \{\mathrm{B}, \mathrm{D}\}$. Similarly, sensor A has a wireless key exchange with sensors C, E, F, G, H, I, and J; we denote this as $\mathrm{A_{wireless}} = \{\mathrm{C}, \mathrm{E}, \mathrm{F}, \mathrm{G}, \mathrm{H}, \mathrm{I}, \mathrm{J}\}$. Note that $\mathrm{A_{kljn}} \cap \mathrm{A_{wireless}} = \emptyset$, that is, every sensor communicating with sensor A must be

classified as having either a wired KLJN key exchange or a wireless key exchange, but not both. The G key exchange trust system is discussed and analyzed in the following section.

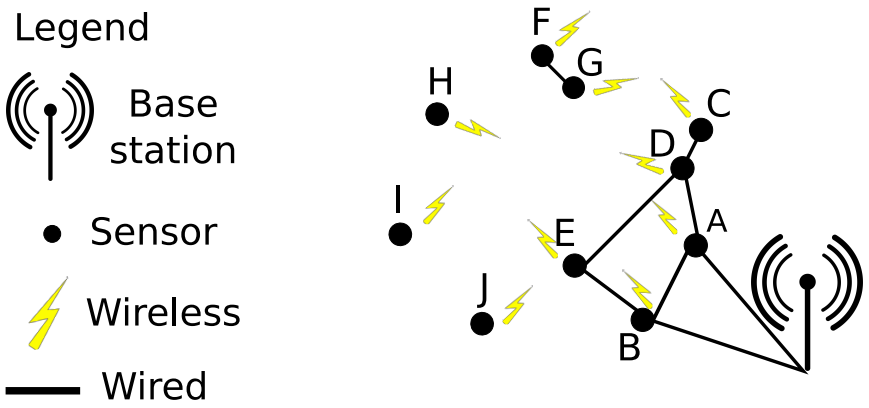

Figure 4.1: An illustration of a wired-wireless hybrid sensor network. In this example there are ten sensors with only select sensors utilizing wired communication channels and all sensors utilizing wireless communication channels.

4.2.2 Protocol

Before sensors can process a KLJN key exchange, the KLJN communicators must be authenticated. The authentication of two KLJN units must be completed before they are separated. The KLJN communicator units have a direct wired connection with each other and thus there is no need for networking protocols, only the KLJN key exchange protocol. However, due to the required pre-authentication of the KLJN communicator units, the sensor network's topography must be planned ahead.

Since the wired KLJN key exchange has been pre-planned, only the wireless key exchanges need to be processed. Once all sensors in the network have a key exchange with every other sensor in the network, every sensor in the network will classify its

Sensor	Wired KLJN Key Exchange	Wireless Key Exchange
A	$A_{kljn} = \{B, D\}$	$A_{wireless} = \{C, E, F, G, H, I, J\}$
B	$B_{kljn} = \{A, E\}$	$B_{wireless} = \{C, D, F, G, H, I, J\}$
C	$C_{kljn} = \{D\}$	$C_{wireless} = \{A, B, E, F, G, H, I, J\}$
D	$D_{kljn} = \{A, C, E\}$	$D_{wireless} = \{B, F, G, H, I, J\}$
E	$E_{kljn} = \{B, D\}$	$E_{wireless} = \{A, C, F, G, H, I, J\}$
F	$F_{kljn} = \{G\}$	$F_{wireless} = \{A, B, C, D, E, H, I, J\}$
G	$G_{kljn} = \{F\}$	$G_{wireless} = \{A, B, C, D, E, H, I, J\}$
H	$H_{kljn} = \emptyset$	$H_{wireless} = \{A, B, C, D, E, F, G, I, J\}$
I	$I_{kljn} = \emptyset$	$I_{wireless} = \{A, B, C, D, E, F, G, H, J\}$
J	$J_{kljn} = \emptyset$	$J_{wireless} = \{A, B, C, D, E, F, G, H, I\}$

Table 4.1: This table lists every sensor's key exchange with all sensors in the network of Figure 4.1. Every sensor is classified as having either a wired KLJN key exchange or a wireless key exchange. Set notation is used to categorize the sets as either KLJN or wireless key exchange.

key exchange with every peer as being either wired or wireless, e.g., A_{kljn} and $A_{wireless}$, B_{kljn} and $B_{wireless}$, etc.

4.3 Geometric Key Exchange Trust System

4.3.1 The Key Exchange Trust Function

The geometric key exchange trust system was designed to have a trust function, G_{ij}, with a range of values, $G_{ij} \in [0, 1]$, as a measure of the key exchange trust of sensor i for its communication channel with sensor j. The function G_{ij} is for sensor i to evaluate the key exchange trust value of sensor j. The input parameters of the function G_{ij} are i_{kljn}, $i_{wireless}$, j_{kljn}, and $j_{wireless}$; these parameters are provided by the operator or the base station.

4.3.2 The Kill Switch

The kill switch, γ_j, is a binary parameter of sensor j in the G_{ij} function, which is set by the operator to $\gamma_j = 0$ when the security of sensor j is compromised, and to

$\gamma_j = 1$ otherwise. The construction of the G_{ij} function (see below) guarantees that for $\gamma_j = 0$ then $G_{ij} = 0$.

4.3.3 Construction of the Key Exchange Trust Function

When constructing the G_{ij} function, the following goals should be satisfied:

(i) The contributing terms to the G_{ij} function are determined by:

(a) The number, K_{ij}, of mutual KLJN key exchanges with sensors i and j, or $K_{ij} = |i_{\text{kljn}} \cap j_{\text{kljn}}|$;

(b) The number, W_j, of KLJN key exchanges with sensor j reduced by K_{ij}, or
$W_j = |j_{\text{kljn}} \setminus (i_{\text{kljn}} \cap j_{\text{kljn}})|$;

(c) The number, Z_j, of only wireless key exchanges in sensor j or $Z_j = |j_{\text{wireless}}|$.

(ii) Strictly monotonic function. The function G_{ij} is a strictly monotonically increasing function determined by the values K_{ij}, W_j, and Z_j. For example, if sensors j and k have values $K_{ij} = K_{ik}$, $W_j = W_k$, and $Z_j > Z_k$, then $G_{ij} > G_{ik}$. As a consequence among the non-compromised sensors, the sensor with a single wireless key exchange should have the lowest contribution.

(iii) Ranks versus class of connections. The contribution of the term containing Z_j will never exceed the contribution of the term containing W_j; the joint contribution of the terms containing W_j and Z_j will never exceed the contribution of the term containing K_{ij}. The reason for this requirement is so that KLJN is the only unconditionally secure key exchange type in the network and thus the rank of its trust is higher.

In terms of $\gamma_j, K_{ij}, W_j, Z_j$ defined in 4.3.3(i) above, equation (4.1) below is the key exchange trust value of sensor i to sensor j:

$$\mathrm{G}_{ij} = \begin{cases} \gamma_j & \text{if } j \in i_{\mathrm{kljn}} \\ \gamma_j \cdot G^*(K_{ij}, W_j, Z_j) & \text{if } j \notin i_{\mathrm{kljn}} \end{cases} \tag{4.1}$$

with

$$G^*(K, W, Z) = \sum_{n=1}^{K} a^n + \sum_{n=1}^{W} b^n + \sum_{n=1}^{Z} c^n \tag{4.2}$$

where $G^* : \{1, 2, ...\}^3 \to [0, 1]$ is a strictly increasing function with respect to the *lexicographic order* (4.3.3(ii)) and also satisfies the requirement of 4.3.3(iii) above. It should be noted that the postulates of G_{ij} allow for several versions of G^*, not only the one given in equation (4.1). The question arises, *why choose geometric series in contributing terms?* We chose the geometric series because of the properties $q = \min G^*$, $\sup G^* = 1$. The postulates are;

$$\begin{aligned} a &= \sum_{n=1}^{\infty} (b^n + c^n), \\ b &= \sum_{n=1}^{\infty} c^n, \\ 1 &= \sum_{n=1}^{\infty} (a^n + b^n + c^n), \\ 1 &> a > b > c > 0. \end{aligned}$$

The ratios a, b, and c can be deduced as follows:

$$\sum_{n=1}^{\infty} a^n + \sum_{n=1}^{\infty}(b^n + c^n) = \frac{a}{1-a} + a = 1$$

$$\Rightarrow a^2 - 3a + 1 = 0$$

$$\Rightarrow a = (3 - \sqrt{5})/2 \approx 0.3820,$$

$$\sum_{n=0}^{\infty} b^n + \sum_{n=0}^{\infty} c^n = \frac{b}{1-b} + b = a$$

$$\Rightarrow b^2 - (a+2)b + a = 0$$

$$\Rightarrow b \approx 0.1729,$$

$$\sum_{n=1}^{\infty} c^n = \frac{c}{1-c} = b$$

$$\Rightarrow c \approx 0.1474,$$

with $a \approx 0.3820$, $b \approx 0.1729$, and $c \approx 0.1474$. The ratios can placed into equation (4.2) to obtain a equation (4.3) below:

$$G^*(K, W, Z) = \sum_{n=1}^{K}(0.3820)^n + \sum_{n=1}^{W}(0.1729)^n + \sum_{n=1}^{Z}(0.1474)^n. \tag{4.3}$$

Equation (4.1) can then be written as equation (4.4),

$$\mathrm{G}_{ij} = \begin{cases} \gamma_j & \text{if } j \in i_{\mathrm{kljn}} \\ \gamma_j \cdot \left(\sum_{n=1}^{K}(0.3820)^n + \sum_{n=1}^{W}(0.1729)^n + \sum_{n=1}^{Z}(0.1474)^n \right) & \text{if } j \notin i_{\mathrm{kljn}} \end{cases} \tag{4.4}$$

with $K = |i_{\text{kljn}} \cap j_{\text{kljn}}|$, $W = |j_{\text{kljn}} \setminus (i_{\text{kljn}} \cap j_{\text{kljn}})|$, $Z = |j_{\text{wireless}}|$, and $\gamma_j = \{0, 1\}$. The case $\gamma_j = 0$ sets $\mathrm{G}_{ij} = 0$. To satisfy the conditions $\mathrm{G}_{ij} \leq 1$ and (i) through (iii) above, we used the following requirements:

(i) The third ($\Sigma_{n=1}^{Z}$) geometric series will saturate at the geometric coefficient of the second ($\Sigma_{n=1}^{W}$) series. That is, the third series, in the case of $Z_j \to \infty$ yields 0.1729.

(ii) The sum of the second ($\Sigma_{n=1}^{W}$) and third ($\Sigma_{n=1}^{Z}$) series, will saturate at the geometric coefficient of the first ($\Sigma_{n=1}^{K}$) series. That is, in the case of $Z_j \to \infty$ and $W_j \to \infty$, their ratio sum yields 0.3820.

(iii) The sum of the three geometric series will saturate to one. That is, in the case of $Z_j \to \infty$, $W_j \to \infty$, and $K_j \to \infty$, their ratio sum yields to one.

The details of the derivation are shown in the next section.

4.3.4 Derivation of G

The G key exchange trust system has a range from zero to one and a kill switch. It must also consider an infinite number of sensors, and that a sensor in a lower level cannot undermine a sensor in a higher level. To achieve this, we propose to utilize the geometric series since the geometric series can add an infinite sum (or the number of sensors) and equal to a finite value (or one.) Since the highest possible value is one, and with an infinite number of sensors, then the G key exchange trust system of sensor j relative to sensor i can be written as

$$\mathrm{G}_{ij}(\gamma_j) = \gamma_j \cdot \sum_{n=1}^{\infty} a^n + b^n + c^n = \gamma_j \cdot 1, \tag{4.5}$$

with $\gamma_j \in \{0, 1\}$ being the kill switch of sensor j, and a, b, and c being the ratios. To solve for ratios a, b, and c, in equation (4.5) we note that

$$\sum_{n=1}^{\infty} a^n + b^n + c^n = 1. \tag{4.6}$$

Note that the following properties must apply according to the G key exchange trust system. The first property is

$$\sum_{n=1}^{\infty} c^n = b, \tag{4.7}$$

which means that an infinite number of sensors in the third series ($\sum_{n=1}^{\infty} c^n$) cannot undermine a single sensor in the second series ($b = \sum_{n=1}^{1} b^n$) or $b^\eta > c^\theta > 0$ for any $\eta, \theta \in \mathbb{N}$. The second property is

$$\sum_{n=1}^{\infty} b^n + c^n = a, \tag{4.8}$$

which means that an infinite number of sensors in the second series ($\sum_{n=1}^{\infty} b^n$) and an infinite number of sensors in the third series ($\sum_{n=1}^{\infty} c^n$) cannot undermine a single sensor in the first series ($a = \sum_{n=1}^{1} a^n$) or $1 > a^\iota > b^\eta + c^\theta > 0$ for any $\iota, \eta, \theta \in \mathbb{N}$. Equation (4.7) and equation (4.8) can be rewritten to isolate the infinite summation of b as follows:

$$\sum_{n=1}^{\infty} b^n = a - b.$$

Also, note that if $r \in \mathbb{R} : |r| < 1$ then $\sum_{n=1}^{\infty} r^n = \frac{r}{1-r}$. Given these properties equation (4.6) can be derived as:

$$\begin{aligned}
\sum_{n=1}^{\infty} a^n + b^n + c^n &= 1, \\
\sum_{n=1}^{\infty} a^n + \sum_{n=1}^{\infty} b^n + \sum_{n=1}^{\infty} c^n &= 1, \\
\sum_{n=1}^{\infty} a^n + \big(a - b\big) + \big(b\big) &= 1, \\
\sum_{n=1}^{\infty} a^n + a &= 1, \\
\frac{a}{1-a} + a &= 1.
\end{aligned}$$

The resulting equation, $a/(1-a) + a = 1$, can be solved for a by using the quadratic formula giving values $a = (3 - \sqrt{5})/2$ and $a = (3 + \sqrt{5})/2$. Since $|a| < 1$, then the only converging value is $a = (3 - \sqrt{5})/2$. Thus the ratio a is

$$a = \frac{3 - \sqrt{5}}{2} \approx 0.3820. \tag{4.9}$$

A similar method can be used to solve for b and c in equation (4.6).

To solve for b note that

$$\sum_{n=1}^{\infty} b^n = a - b,$$

$$\frac{b}{1-b} = a - b. \tag{4.10}$$

Solving for b in equation (4.10) gives two solutions. The converging solution is,

$$b = \frac{a + 2 + \sqrt{a^2 + 4}}{2}. \tag{4.11}$$

Given equation (4.9) and substituting for a in equation (4.11) gives

$$b = \frac{7 - \sqrt{5} - \sqrt{30 - 6\sqrt{5}}}{4} \approx 0.1729. \tag{4.12}$$

Thus the ratio b is given by equation (4.12).

The ratio c can be solved by utilizing equation (4.7). Note that

$$\sum_{n=1}^{\infty} c^n = b,$$

$$\frac{c}{1-c} = b. \tag{4.13}$$

Given equation (4.12) and substituting for b in equation (4.13), then solving for c will give

$$c = \frac{\sqrt{30 - 6\sqrt{5}} + \sqrt{5} - 7}{\sqrt{30 - 6\sqrt{5}} + \sqrt{5} - 11} \approx 0.1474. \tag{4.14}$$

Thus the ratio c is given by equation (4.14).

The derivations above were derived to consider any number of sensors, thus the G key exchange trust function holds for zero sensors to an infinite number of sensors. In reality there will be a limited number of sensors in a network.

The ratio a will only consider sensors that are conditionally secured with mutual KLJN key exchanges, e.g., if sensor i and sensor j have mutual KLJN key exchanges with third parties, then this can be written in set notation as the intersection of sensor i's i_{kljn} set and sensor j's j_{kljn} set. This can be expressed as $i_{\text{kljn}} \cap j_{\text{kljn}}$. The number of mutual KLJN key exchanges with third parties between sensors i and j can be expressed as $K = |i_{\text{kljn}} \cap j_{\text{kljn}}|$. Thus, there are K mutual sensors between sensors i and j.

The ratio b will only consider sensors that are conditionally secured without mutual KLJN key exchanges, e.g., if sensor i evaluates the number of key exchanges in sensor j, then only the number of KLJN key exchanges in sensor j that do not have mutual KLJN key exchanges with sensor i will be noted. This can be expressed as $W = |j_{\text{kljn}} \setminus (i_{\text{kljn}} \cap j_{\text{kljn}})|$. The purpose of having ratio b is based on the belief that a sensor with a KLJN key exchange should have a higher key exchange trust value than a sensor without a KLJN key exchange.

The ratio c will only consider sensors that are conditionally secured with only wireless key exchanges, e.g., if sensor j only has wireless key exchanges with other sensors then the number of sensors that can verify a wireless key exchange with sensor j is $Z = |j_{\text{wireless}}|$.

The G key exchange trust system can evaluate the key exchange trust level of sensor j relative to sensor i, which can be expressed as $\mathrm{G}_{ij}(\gamma_j)$, with γ_j being the kill switch for sensor j. $\mathrm{G}_{ij}(\gamma_j)$ can be expressed as the following equation;

$$\mathrm{G}_{ij}(\gamma_j) = \begin{cases} \gamma_j & \text{if } j \in i_{\text{kljn}} \\ \gamma_j \cdot \left(\sum_{n=1}^{K} (0.3820)^n + \sum_{n=1}^{W} (0.1729)^n + \sum_{n=1}^{Z} (0.1474)^n \right) & \text{if } j \notin i_{\text{kljn}}, \end{cases}$$

with $K = |i_{\text{kljn}} \cap j_{\text{kljn}}|$, $W = |j_{\text{kljn}} \setminus (i_{\text{kljn}} \cap j_{\text{kljn}})|$, $Z = |j_{\text{wireless}}|$, and $\gamma_j = \{0, 1\}$.

4.3.5 Example

Equation (4.4) was applied to the network in Figure 4.1. The G key exchange trust values for all the sensors in Figure 4.1 are in Table 4.2. From Table 4.2 some properties of G can be observed. The G function is asymmetric, e.g., in Table 4.2 note that $\text{G}_{BC} \neq \text{G}_{CB}$. There is also incomplete transitive, e.g., in Table 4.2 note that $\text{G}_{AD} = 1$ and $\text{G}_{DC} = 1$, but $\text{G}_{AC} = 0.555$ and does not equal one. Note that the G function given by equation (4.4) is unique for the given conditions. The conditions are to have a range between zero and one and a kill switch. Also note that an infinite number of sensors in lower levels will not undermine a single sensor in a higher level.

		j									
	Sensor	A	B	C	D	E	F	G	H	I	J
i	A	1	1	0.555	1	0.701	0.346	0.346	0.173	0.173	0.173
	B	1	1	0.346	0.874	1	0.346	0.346	0.173	0.173	0.173
	C	0.728	0.376	1	1	0.728	0.346	0.346	0.173	0.173	0.173
	D	1	0.701	1	1	1	0.346	0.346	0.173	0.173	0.173
	E	0.701	1	0.555	1	1	0.346	0.346	0.173	0.173	0.173
	F	0.376	0.376	0.346	0.381	0.376	1	1	0.173	0.173	0.173
	G	0.376	0.376	0.346	0.381	0.376	1	1	0.173	0.173	0.173
	H	0.376	0.376	0.346	0.381	0.376	0.346	0.346	1	0.173	0.173
	I	0.376	0.376	0.346	0.381	0.376	0.346	0.346	0.173	1	0.173
	J	0.376	0.376	0.346	0.381	0.376	0.346	0.3458	0.173	0.173	1

Table 4.2: This table lists G_{ij} key exchange trust values for all the sensors in Figure 4.1. This table assumes $\gamma_j = 1$ for all js.

As shown in Table 4.2, the G key exchange trust system will give a higher key exchange trust evaluation to sensors that are part of a KLJN key exchange, the more KLJN key exchanges a sensor has, the higher the key exchange trust evaluation. Sen-

sors without a KLJN key exchange will have a lower key exchange trust evaluation, even if there are an infinite number of sensors with only wireless key exchange. This mechanism will prevent a lower level sensor attempting to undermine a higher level sensor since there are ceiling limits to sensors that only share a wireless key exchange. A kill switch is in place to allow the G system to remain subjective with any sensor at any time.

4.4 Open Questions and Future Work

Since all sensors in the G system must have both wired and wireless communication channels it will not be practical in some applications. Sensors in the G system will also need to utilize both symmetric encryption for the KLJN key exchange and asymmetric encryption for the wireless key exchange; this will increase energy requirements, computing requirements, memory, and data storage. Sensors are dependent on the operator or base station to provide or broadcast the KLJN and wireless key exchange sets of every sensor in the network; this dependency will require the sensors to remain centralized. For sensors to be autonomous, future work must be done where each sensor can broadcast its key exchange sets. Another concern is concealing the cable between the wired sensors. Unconditionally secure key exchange has not been experimented with in sensor networks, but the realization of such a network should be of significant interest. The cost of having unconditionally secure key exchange for sensor networks is high, but such is the price for high security.

For sensors that cannot communicate with other sensors or the base station due to the distance between them, a multi-hop method is utilized [81]. The G system does not consider multi-hop cases and would give the sensor a key exchange trust evaluation of the last sensor it was able to communicate with; this can be improved in

future work. Sensor networks can also utilize different protocols for different KLJN geometric networks to reduce the cable, time, and KLJN communicators cost as has been analyzed in [32, 31, 28, 29, 30].

5. CONCLUSION[1,2,3]

In this work we have introduced a protocol for linear chain networks to offer unconditionally secure key exchange over the smart grid. We used a reconfigurable filter system and proposed a special protocol for linear chain networks to create non-overlapping single loops in the smart power grid for the realization of Kirchhoff-Law-Johnson-(like)-Noise secure key distribution system. We carried out a scaling analysis for the speed of the protocol versus the size of the grid. When properly generalized, such a system has the potential to achieve unconditionally secure key distribution over the smart power grid of arbitrary dimensions.

Before the implementation of the protocol on linear chain networks can take place, several practical questions must be answered, such as the impact of finite and possibly varying wire resistance, capacitance, power load on the security, and the applications of relevant privacy amplification methods. Other questions include changing size N, hacking penetration attacks against the filter control, and the relevant defensive tools. We also discussed the limitations of the KLJN key exchange protocol, open questions surrounding the implementation on smart grids, and future work required. Since this work is a system-concept study we leave the details to future work.

---

1 Part of this chapter is reprinted with permission from "Information Theoretically Secure, Enhanced Johnson Noise Based Key Distribution over the Smart Grid with Switched Filters" by Gonzalez, E., Kish, L.B., Balog, R.S., Prasad, E., (2013). *PLoS ONE*, 8(7): e70206 Copyright 2013 by PLOS

2 Part of this chapter is reprinted with permission from "Resource Requirements and Speed *versus* Geometry of Unconditionally Secure Physical Key Exchanges" by Gonzalez, E., Balog, R.S., Kish, L.B., (2015). *Entropy*, 17(4), pp. 2010-2024; DOI:10.3390/e17042010 Copyright 2015 by MDPI

3 Part of this chapter is reprinted with permission from "Key Exchange Trust Evaluation in Peer-to-Peer Sensor Networks With Unconditionally Secure Key Exchange" by Gonzalez, E., Kish, L.B., (2016). A print and electronic version of this article published in *Fluctuation and Noise Letters*, Vol. 15, No. 1, 2016, pp. 165008 (17 pages) DOI:10.1142/S0219477516500085 ©World Scientific Publishing Company http://www.worldscientific.com/worldscinet/fnl

We also considered the need for unconditional secure key exchange along with the need to have P2P networks since QKD and KLJN require P2P networks. We reviewed a simple P2P network known as the fully connected network. We also reviewed the linear chain network and analyzed the star network to compared it with fully connected networks and linear chain networks. We conceived a protocol and equations (3.1a) through (3.1d) to describe star networks. The results show that the star network compares favorably to the linear chain network and the fully connected network. Even though the star network utilizes only one key exchanger per host, its time complexity is superior to that of the linear chain network, while its cable complexity is the same. The star network's cable and key exchanger complexity is superior to that of the fully connected network, while its time complexity is worse than FCN_{N-1}, but is similar to FCN_1. We found that the star network fairs worse than the linear chain network and the fully connected network in robustness and reliability as the star network can be entirely disconnected by damaging the center switch. We then considered several other possible network geometries that might be interesting to explore and to compare.

We also introduced sensor networks along with some applications, limitations, and security issues. We then discuss unconditionally secure key exchanges, and mention how the KLJN key exchange can be included in sensor networks. We also mention current trust methodologies for sensor networks. Since current trust methodologies do not consider unconditionally secure key exchange we introduce the geometric key exchange trust system, a new key exchange trust method for sensor networks that considers unconditionally secure key exchange in the key exchange trust measure. An example of sensor networks with sensors utilizing both wired and wireless communication channels is depicted in Figure 4.1. The G key exchange trust system is then introduced and applied to the sensor network example in Figure 4.1. The G key

exchange trust system is then analyzed, discussed, and modeled by equation (4.4). Table 4.2 shows that a higher key exchange trust evaluation is given to sensors with KLJN key exchanges, and the more KLJN key exchanges a sensor has, the higher the key exchange trust evaluation. Equation (4.4) and Table 4.2 also show that there are ceiling limits to sensors that only share a wireless key exchange. The G system depends on the operator or base station to provide the key exchange sets of every sensor in the network. The kill switch allows the G system to remain subjective of every sensor in the network. We then discuss open questions about the G system and possible future improvements.

We believe this work in applying unconditionally secure key exchange over smart grid networks will generate interest as we transcend the dark ages of information security.

REFERENCES

[1] NSA Director: China can damage U.S. power grid. `https://www.youtube.com/watch?v=Pw79NyHleB8`, November 2014. Accessed: 2014-11-20.

[2] Udo Adamy, Thomas Erlebach, Dieter Mitsche, Ingo Schurr, Bettina Speckmann, and Emo Welzl. *Approximation and Online Algorithms: Second International Workshop, WAOA 2004, Bergen, Norway, September 14-16, 2004, Revised Selected Papers*, chapter Off-line Admission Control for Advance Reservations in Star Networks, pages 211–224. Springer Berlin Heidelberg, Berlin, Heidelberg, 2005.

[3] Jin Akiyama and Mikio Kano. *Path factors of a graph, Graph Theory and its Applications*. Wiley, New York, 1984.

[4] Moshaddique A. Ameen, Jingwei Liu, and Kyungsup Kwak. Security and privacy issues in wireless sensor networks for healthcare applications. *Journal of medical systems*, 36(1):93–101, 2012.

[5] Saurabh M. Amin and Bruce F. Wollenberg. Toward a smart grid: Power delivery for the 21st century. *IEEE Power and Energy Magazine*, 3(5):34–41, Sept 2005.

[6] Robert S. Balog and Philip T. Krein. Coupled-inductor filter: A basic filter building block. *Power Electronics, IEEE Transactions on*, 28(1):537–546, 2013.

[7] Fenye Bao, Ing R. Chen, Moon J. Chang, and Jin H. Cho. Hierarchical trust management for wireless sensor networks and its applications to trust-based routing and intrusion detection. *IEEE Transactions on Network and Service Management*, 9(2):169–183, June 2012.

[8] Charles H. Bennett and Gilles Brassard. Quantum cryptography: Public key distribution and coin tossing. In *International Conference on Computer System and Signal Processing, IEEE, 1984*, pages 175–179, December 1984. Bangalore, India.

[9] L'udmila Bezegová, Borut Lužar, Martina Mockovčiaková, Roman Soták, and Riste Škrekovski. Star edge coloring of some classes of graphs. *Journal of Graph Theory*, 81(1):73–82, 2016.

[10] Matt Blaze, Joan Feigenbaum, and John Lacy. Decentralized trust management. In *Security and Privacy, 1996. Proceedings., 1996 IEEE Symposium on*, pages 164–173, May 1996.

[11] Jeremy J. Blum, Azim Eskandarian, and Lance J. Hoffman. Challenges of intervehicle ad hoc networks. *IEEE Transactions on Intelligent Transportation Systems*, 5(4):347–351, Dec 2004.

[12] Azzedine Boukerche and Yonglin Ren. A trust-based security system for ubiquitous and pervasive computing environments. *Computer Communications*, 31(18):4343–4351, 2008.

[13] Xiaolin Cao, Yessica Saez, Geza Pesti, and Laszlo B. Kish. On KLJN-based secure key distribution in vehicular communication networks. *Fluctuation and Noise Letters*, 14(01):1550008, 2015.

[14] Ariel Caticha and Adom Giffin. Updating probabilities. *arXiv preprint physics/0608185*, 2006.

[15] Shu-Park Chan. *Network topology and its engineering applications*. National Taiwan University Press, 1975.

[16] Hsien-Pu Chen, Laszlo B. Kish, and Claes G. Granqvist. On the "cracking" scheme in the paper "A directional coupler attack against the Kish key distribution system" by Gunn, Allison and Abbott. *Metrology and Measurement Systems*, 21(3):389–400, 2014.

[17] Hsien-Pu Chen, Elias Gonzalez, Yessica Saez, and Laszlo B. Kish. Cable capacitance attack against the KLJN secure key exchange. *Information*, 6(4):719–732, 2015.

[18] Hsien-Pu Chen, Laszlo B. Kish, Claes G. Granqvist, and Gabor Schmera. Do electromagnetic waves exist in a short cable at low frequencies? what does physics say? *Fluctuation and Noise Letters*, 13(02):1450016, 2014.

[19] Youngho Cho, Gang Qu, and Yuanming Wu. Insider threats against trust mechanism with watchdog and defending approaches in wireless sensor networks. In *Security and Privacy Workshops (SPW), 2012 IEEE Symposium on*, pages 134–141, May 2012.

[20] David Culler, Deborah Estrin, and Mani Srivastava. Guest editors' introduction: Overview of sensor networks. *Computer*, 37(8):41–49, Aug 2004.

[21] Raju Dutta, Shishir Gupta, and Debraj Paul. Energy efficient modified spin protocol with high security in wireless sensor networks using tossim. In *Parallel, Distributed and Grid Computing (PDGC), 2014 International Conference on*, pages 290–294, Dec 2014.

[22] Mona El and Eman Shaaban. Enhancing s-leach security for wireless sensor networks. In *Electro/Information Technology (EIT), 2012 IEEE International Conference on*, pages 1–6, May 2012.

[23] Eric Engleman and Jordan Robertson. Obama to share cybersecurity priorities with congress. http://www.bloomberg.com/news/2013-02-27/obama-to-share-cybersecurity-priorities-with-congress.html, February 2013. Accessed: 2013-06-26.

[24] Niels Ferguson and Bruce Schneier. *Practical cryptography*, volume 23. Wiley New York, 2003.

[25] Ilja Gerhardt, Qin Liu, Antía Lamas-Linares, Johannes Skaar, Christian Kurtsiefer, and Vadim Makarov. Full-field implementation of a perfect eavesdropper on a quantum cryptography system. *Nature communications*, 2:349, 2011.

[26] Ilja Gerhardt, Qin Liu, Antía Lamas-Linares, Johannes Skaar, Valerio Scarani, Vadim Makarov, and Christian Kurtsiefer. Experimentally faking the violation of Bell's inequalities. *Physical Review Letters*, 107(17):170404, 2011.

[27] Zoltan Gingl and Robert Mingesz. Noise properties in the ideal Kirchhoff-Law-Johnson-Noise secure communication system. *PLoS ONE*, 9(4):1–4, 04 2014.

[28] Elias Gonzalez, Robert S. Balog, and Laszlo B. Kish. Resource requirements and speed versus geometry of unconditionally secure physical key exchanges. *Entropy*, 17(4):2010, 2015.

[29] Elias Gonzalez, Robert S. Balog, Robert Mingesz, and Laszlo B. Kish. Unconditional security for the smart power grids and star networks. In *Noise and Fluctuations (ICNF), IEEE 2015 International Conference on*, pages 1–4, June 2015.

[30] Elias Gonzalez and Laszlo B. Kish. Key exchange trust evaluation in peer-to-peer sensor networks with unconditionally secure key exchange. *Fluctuation*

and Noise Letters, 0(0):1650008, 0.

[31] Elias Gonzalez, Laszlo B. Kish, and Robert S. Balog. Encryption key distribution system and method, February 23 2016. US Patent 9,270,448.

[32] Elias Gonzalez, Laszlo B. Kish, Robert S. Balog, and Prasad Enjeti. Information theoretically secure, enhanced Johnson noise based key distribution over the smart grid with switched filters. *PLoS ONE*, 8(7):1–10, 07 2013.

[33] Tyrone Grandison and Morris Sloman. A survey of trust in internet applications. *IEEE Communications Surveys Tutorials*, 3(4):2–16, Fourth 2000.

[34] Jonathan L. Gross and Jay Yellen. *Handbook of graph theory*. CRC press: Boca Raton, FL, USA, 2004.

[35] Daojing He, Chun Chen, Sammy Chan, Jiajun Bu, and Laurence T. Yang. Security analysis and improvement of a secure and distributed reprogramming protocol for wireless sensor networks. *IEEE Transactions on Industrial Electronics*, 60(11):5348–5354, Nov 2013.

[36] Osamu Hirota. Incompleteness and limit of quantum key distribution theory. *arXiv preprint arXiv:1208.2106*, 2012.

[37] Lih-Hsing Hsu and Cheng-Kuan Lin. *Graph theory and interconnection networks*. CRC press: Boca Raton, FL, USA, 2008.

[38] Shu Huang, Rudra Dutta, and George N. Rouskas. Traffic grooming in path, star, and tree networks: complexity, bounds, and algorithms. *IEEE Journal on Selected Areas in Communications*, 24(4):82, 2006.

[39] Abhishek Jain, Kamal Kant, and Malay R. Tripathy. Security solutions for wireless sensor networks. In *Advanced Computing Communication Technologies (ACCT), 2012 Second International Conference on*, pages 430–433, Jan 2012.

[40] Nitin Jain, Christoffer Wittmann, Lars Lydersen, Carlos Wiechers, Dominique Elser, Christoph Marquardt, Vadim Makarov, and Gerd Leuchs. Device calibration impacts security of quantum key distribution. *Physical Review Letters*, 107(11):110501, 2011.

[41] Audun Jøsang. A logic for uncertain probabilities. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 9(03):279–311, 2001.

[42] Audun Jøsang, Ross Hayward, and Simon Pope. Trust network analysis with subjective logic. In *Proceedings of the 29th Australasian Computer Science Conference-Volume 48*, pages 85–94. Australian Computer Society, Inc., 2006.

[43] Audun Jøsang, Roslan Ismail, and Colin Boyd. A survey of trust and reputation systems for online service provision. *Decision support systems*, 43(2):618–644, 2007.

[44] Mladen Kezunovic. Smart fault location for smart grids. *IEEE Transactions on Smart Grid*, 2(1):11–22, March 2011.

[45] Sangun Kim and Prasad Enjeti. A new hybrid active power filter (APF) topology. *Power Electronics, IEEE Transactions on*, 17(1):48–54, 2002.

[46] Brooks King-Casas, Damon Tomlin, Cedric Anen, Colin F. Camerer, Steven R. Quartz, and Read P. Montague. Getting to know you: reputation and trust in a two-person economic exchange. *Science*, 308(5718):78–83, 2005.

[47] Laszlo B. Kish. Protection against the Man-In-The-Middle attack for the Kirchhoff-Loop-Johnson(-like)-Noise cipher and expansion by voltage-based security. *Fluctuation and Noise Letters*, 6(01):L57–L63, 2006.

[48] Laszlo B. Kish. Response to Feng Hao's paper "Kish's key exchange scheme is insecure". *Fluctuation and Noise Letters*, 6(04):C37–C41, 2006.

[49] Laszlo B. Kish. Totally secure classical communication utilizing Johnson(-like)-Noise and Kirchoff's law. *Physics Letters A*, 352(3):178–182, 2006.

[50] Laszlo B. Kish. Enhanced secure key exchange systems based on the Johnson-noise scheme. *Metrology and Measurement Systems*, 20(2):191–204, 2013.

[51] Laszlo B. Kish, Derek Abbott, and Claes G. Granqvist. Critical analysis of the Bennett-Riedel attack on secure cryptographic key distributions via the Kirchhoff-Law-Johnson-Noise scheme. *PLoS ONE*, 8(12):1–15, 12 2013.

[52] Laszlo B. Kish, Zoltan Gingl, Robert Mingesz, Gergely Vadai, Janusz Smulko, and Claes-Göran Granqvist. Analysis of an attenuator artifact in an experimental attack by Gunn-Allison-Abbott against the Kirchhoff-Law-Johnson-Noise (KLJN) secure key exchange system. *Fluctuation and Noise Letters*, 14(01):1550011, 2015.

[53] Laszlo B. Kish and Claes G. Granqvist. Elimination of a Second-Law-Attack, and all Cable-Resistance-Based Attacks, in the Kirchhoff-Law-Johnson-Noise (KLJN) Secure Key Exchange System. *Entropy*, 16(10):5223, 2014.

[54] Laszlo B. Kish and Claes G. Granqvist. On the security of the Kirchhoff-law–Johnson-noise (KLJN) communicator. *Quantum Information Processing*, 13(10):2213–2219, 2014.

[55] Laszlo B. Kish and Claes-Göran Granqvist. Enhanced usage of keys obtained by physical, unconditionally secure distributions. *Fluctuation and Noise Letters*, 14(02):1550007, 2015.

[56] Laszlo B. Kish and Tamas Horvath. Notes on recent approaches concerning the Kirchhoff-law–Johnson-noise-based secure key exchange. *Physics Letters A*, 373(32):2858–2868, 2009.

[57] Laszlo B. Kish and Chiman Kwan. Physical unclonable function hardware keys utilizing Kirchhoff-law-Johnson-noise secure key exchange and noise-based logic. *Fluctuation and Noise Letters*, 12(03):1350018, 2013.

[58] Laszlo B. Kish and Robert Mingesz. Totally secure classical networks with multipoint telecloning (teleportation) of classical bits through loops with Johnson-like noise. *Fluctuation and Noise Letters*, 6(02):C9–C21, 2006.

[59] Laszlo B. Kish and Olivier Saidi. Unconditionally secure computers, algorithms and hardware, such as memories, processors, keyboards, flash and hard drives. *Fluctuation and Noise Letters*, 8(02):L95–L98, 2008.

[60] Deepa Kundur, Xianyong Feng, Shan Liu, Takis Zourntos, and Karen L. Butler-Purry. Towards a framework for cyber attack impact analysis of the electric smart grid. In *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on*, pages 244–249, Oct 2010.

[61] Bernd Lahno. *Jahrbuch für Handlungs- und Entscheidungstheorie*, chapter Is Trust the Result of Bayesian Learning?, pages 47–68. VS Verlag für Sozialwissenschaften, Wiesbaden, 2004.

[62] Chun-Ta Li, Chi-Yao Weng, and Cheng-Chi Lee. An advanced temporal credential-based security scheme with mutual authentication and key agreement for wireless sensor networks. *Sensors*, 13(8):9589, 2013.

[63] Yingbin Liang, Vincent Poor, and Shlomo Shamai. Information theoretic security. *Foundations and Trends in Communications and Information Theory*, 5(4–5):355–580, 2009.

[64] Rongxing Lu, Xiaodong Lin, Haojin Zhu, Xiaohui Liang, and Xuemin Shen. BECAN: a bandwidth-efficient cooperative authentication scheme for filtering

injected false data in wireless sensor networks. *IEEE Transactions on Parallel and Distributed Systems*, 23(1):32–43, Jan 2012.

[65] Eduard Lucas. Recreations mathematiques, four volumes: Gautheir-villars. *Paris, France (1882/1894)*, pages 161–197, 1882.

[66] Lars Lydersen, Mohsen K. Akhlaghi, Hamed Majedi, Johannes Skaar, and Vadim Makarov. Controlling a superconducting nanowire single-photon detector using tailored bright illumination. *New Journal of Physics*, 13(11):113042, 2011.

[67] Lars Lydersen, Nitin Jain, Christoffer Wittmann, Øystein Marøy, Johannes Skaar, Christoph Marquardt, Vadim Makarov, and Gerd Leuchs. Superlinear threshold detectors in quantum cryptography. *Physical Review A*, 84(3):032320, 2011.

[68] Lars Lydersen, Vadim Makarov, and Johannes Skaar. Comment on Resilience of gated avalanche photodiodes against bright illumination attacks in quantum cryptography. *Applied physics letters*, 99(19), 2011.

[69] Lars Lydersen, Johannes Skaar, and Vadim Makarov. Tailored bright illumination attack on distributed-phase-reference protocols. *Journal of Modern Optics*, 58(8):680–685, 2011.

[70] Lars Lydersen, Carlos Wiechers, Christoffer Wittmann, Dominique Elser, Johannes Skaar, and Vadim Makarov. Avoiding the blinding attack in QKD. *Nature Photonics*, 4(12):801–801, 2010.

[71] Lars Lydersen, Carlos Wiechers, Christoffer Wittmann, Dominique Elser, Johannes Skaar, and Vadim Makarov. Hacking commercial quantum cryptogra-

phy systems by tailored bright illumination. *Nature photonics*, 4(10):686–689, 2010.

[72] Lars Lydersen, Carlos Wiechers, Christoffer Wittmann, Dominique Elser, Johannes Skaar, and Vadim Makarov. Thermal blinding of gated detectors in quantum cryptography. *Optics express*, 18(26):27938–27954, 2010.

[73] Vadim Makarov. Controlling passively quenched single photon detectors by bright light. *New Journal of Physics*, 11(6):065003, 2009.

[74] Vadim Makarov and Johannes Skaar. Faked states attack using detector efficiency mismatch on SARG04, phase-time, DPSK, and Ekert protocols. *Quantum Information & Computation*, 8(6):622–635, 2008.

[75] Patrick McDaniel and Stephen McLaughlin. Security and privacy challenges in the smart grid. *IEEE Security Privacy*, 7(3):75–77, May 2009.

[76] Koh K. Meng, Dong Fengming, and Tay E. Guan. *Introduction to graph theory: H3 mathematics*. World Scientific Hackensack, NJ, USA, 2007.

[77] Robert Mingesz, Laszlo B. Kish, Zoltan Gingl, Claes-Göran Granqvist, He Wen, Ferdinand Peper, Travis Eubanks, and Gabor Schmera. Unconditional security by the laws of classical physics. *Metrology and Measurement Systems*, 20(1):3–16, 2013.

[78] Robert Mingesz, Zoltan Gingl, and Laszlo B. Kish. Johnson(-like)-Noise-Kirchhoff-Loop based secure classical communicator characteristics, for ranges of two to two thousand kilometers, via model-line. *Physics Letters A*, 372(7):978–984, 2008.

[79] Robert Mingesz, Gergely Vadai, and Zoltan Gingl. What kind of noise guarantees security for the Kirchhoff-Law-Johnson-Noise key exchange? *Fluctuation*

and Noise Letters, 13(03):1450021, 2014.

[80] Hero Modares, Rosli Salleh, and Amirhossein Moravejosharieh. Overview of security issues in wireless sensor networks. In *Computational Intelligence, Modelling and Simulation (CIMSiM), 2011 Third International Conference on*, pages 308–311, Sept 2011.

[81] Miguel Navarro, Tyler W. Davis, German Villalba, Yimei Li, Xiaoyang Zhong, Newlyn Erratt, Xu Liang, and Yao Liang. Towards long-term multi-hop WSN deployments for environmental monitoring: an experimental network evaluation. *Journal of Sensor and Actuator Networks*, 3(4):297–330, 2014.

[82] Ferdinand Peper and Laszlo B. Kish. Information networks secured by the laws of physics. *IEICE transactions on communications*, 95(5):1501–1507, 2012.

[83] Adrian Perrig, Robert Szewczyk, Justin D. Tygar, Victor Wen, and David E. Culler. SPINS: Security protocols for sensor networks. *Wireless networks*, 8(5):521–534, 2002.

[84] Srinivasa Prasanna and Srinivasa Rao. An overview of wireless sensor networks applications and security. *International Journal of Soft Computing and Engineering (IJSCE), ISSN*, pages 2231–2307, 2012.

[85] Amar Rasheed and Rabi N. Mahapatra. The three-tier security scheme in wireless sensor networks with mobile sinks. *IEEE Transactions on Parallel and Distributed Systems*, 23(5):958–965, May 2012.

[86] Mark Rhodes-Ousley. *Information security the complete reference*. McGraw Hill Professional, 2013.

[87] Lawrence G. Roberts and Barry D. Wessler. Computer network development to achieve resource sharing. In *Proceedings of the May 5-7, 1970, spring joint*

computer conference, pages 543–549. ACM, 1970.

[88] Yessica Saez, Xiaolin Cao, Laszlo B. Kish, and Geza Pesti. Securing vehicle communication systems by the KLJN key exchange protocol. *Fluctuation and Noise Letters*, 13(03):1450020, 2014.

[89] Sebastien Sauge, Lars Lydersen, Andrey Anisimov, Johannes Skaar, and Vadim Makarov. Controlling an actively-quenched single photon detector with bright light. *Optics express*, 19(23):23590–23600, 2011.

[90] Bruce Schneider. *Applied Cryptography: Protocols, algorithms, and source code in C*. John Wiley & Sons, 1996.

[91] Rudiger Schollmeier. A definition of peer-to-peer networking for the classification of peer-to-peer architectures and applications. In *Peer-to-Peer Computing, 2001. Proceedings. First International Conference on*, pages 101–102, Aug 2001.

[92] Shio K. Singh, Mp Singh, and Dharmendra K. Singh. A survey on network security and attack defense mechanism for wireless sensor networks. *International Journal of Computer Trends and Technology*, 1(2):9–17, 2011.

[93] Alexander Soifer. *The Mathematical Coloring Book: Mathematics of Coloring and the Colorful Life of its Creators*. Springer: New York, NY USA, 2008.

[94] John A. Stankovic. Research directions for the internet of things. *IEEE Internet of Things Journal*, 1(1):3–9, Feb 2014.

[95] William T. Tutte. The factorization of linear graphs. *Journal of the London Mathematical Society*, 1(2):107–111, 1947.

[96] William T. Tutte. The factors of graphs. *Canad. J. Math*, 4(3):314–328, 1952.

[97] William T. Tutte. *Graph Theory*. Cambridge University Press: Cambridge, UK, 2001.

[98] Carlos Wiechers, Lars Lydersen, Christoffer Wittmann, Dominique Elser, Johannes Skaar, Ch Marquardt, Vadim Makarov, and Gerd Leuchs. After-gate attack on a quantum cryptosystem. *New Journal of Physics*, 13(1):013043, 2011.

[99] Yanli Yu, Keqiu Li, Wanlei Zhou, and Ping Li. Trust mechanisms in wireless sensor networks: Attack analysis and countermeasures. *Journal of Network and computer Applications*, 35(3):867–880, 2012.

[100] Horace P. Yuen. On the foundations of quantum key distribution-Reply to Renner and beyond. *arXiv preprint arXiv:1210.2804*, 2012.

[101] Horace P. Yuen. Essential elements lacking in security proofs for quantum key distribution. In *SPIE Security+ Defence*, pages 88990J–88990J. International Society for Optics and Photonics, 2013.