

Article

# Resource Requirements and Speed *versus* Geometry of Unconditionally Secure Physical Key Exchanges

Elias Gonzalez \*, Robert S. Balog and Laszlo B. Kish

Department of Electrical and Computer Engineering, Texas A&M University, 3128 TAMU College Station, TX 77843, USA; E-Mails: rbalog@tamu.edu (R.S.B.); Laszlo.Kish@ece.tamu.edu (L.B.K.)

\* Author to whom correspondence should be addressed; E-Mail: eliasg23@tamu.edu.

Received: 1 February 2015 / Accepted: 31 March 2015 / Published: 3 April 2015

---

**Abstract:** The imperative need for unconditional secure key exchange is expounded by the increasing connectivity of networks and by the increasing number and level of sophistication of cyberattacks. Two concepts that are theoretically information-secure are quantum key distribution (QKD) and Kirchoff-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. In this article we analyze the cost complexities of cable, key exchangers, and time required in the star network. We mentioned 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.

**Keywords:** KLJN; QKD; networks; star network; secure key exchange; information theoretically secure; unconditional security

---

## 1. Introduction

### 1.1. Motivation for a Secure Network

In the advent of intelligent vehicle information networks [1], the smart power grid [2], and the Internet of Things (IoT) [3], 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 [4].

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. Secure key exchanges can be categorized as either software-based or hardware-based.



**Figure 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.

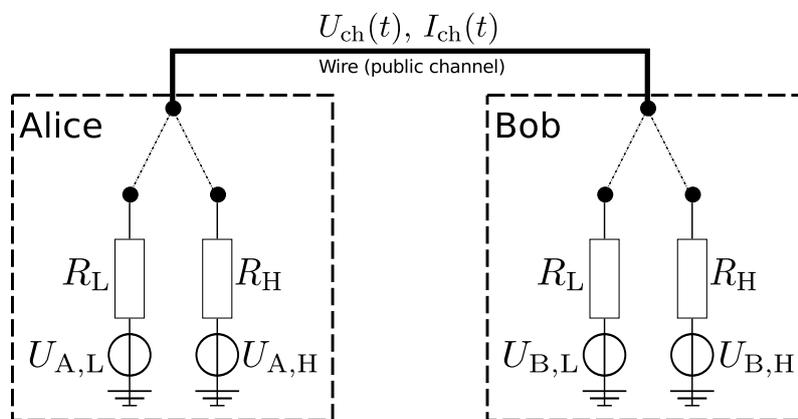
Software-based key exchanges are based on mathematical algorithms with the assumption that Eve does not have enough computing resources to crack the key. In essence, software-based key exchanges offer no security from an information theoretical point of view. The security is only (computationally-) conditional 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 extra infrastructure. The other option is hardware-based key exchange, these offer an advantage of unconditional security.

### 1.2. Hardware-Based Secure Key Exchanges

The Quantum Key Distribution (QKD) [5] and the Kirchhoff-Law-Johnson-Noise (KLJN) [6–26] secure key exchange are two examples of hardware-based secure key exchange concepts that are information theoretically secure [27]. Thus, even with infinite computing resources the key will not be extracted by Eve, because the security offered by these schemes are 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) [28].

The QKD key exchange utilizes the quantum no-cloning theorem of quantum mechanics [5] 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 [29–32].

The KLJN key exchange utilizes the laws and properties of classical mechanics [6] to generate and distribute key bits. In the KLJN key exchange depicted in Figure 2, Alice and Bob have two identical resistor pairs,  $R_L$  and  $R_H$  (the values of the resistors are such that  $R_L < R_H$ ) which represents the low  $R_L$  and high  $R_H$  bits respectively. At the beginning of each bit exchange period, the communicators randomly generate a bit value and connect the corresponding resistor to the wire line. The effective value of the resulting thermal noise in the cable has three possible levels. When it is at the intermediate level, Alice and Bob will know that the other party has the opposite bit value than their own. Thus, a secure bit exchange took place because Eve, while she also knows that Alice and Bob have opposite bit values, she does not know who has the  $R_L$  value and who has the  $R_H$  value [6,8].



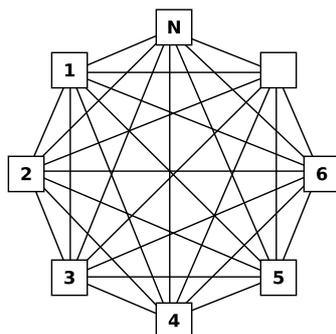
**Figure 2.** An illustration of a Kirchhoff-Law-Johnson-Noise (KLJN) system. Alice and Bob each have a communicator which have noise generators, a low resistor  $R_L$ , and a high resistor  $R_H$ . The noise voltages are enhanced by Johnson noise  $U_{A,L}$  or  $U_{A,H}$  for Alice; and  $U_{B,L}$  or  $U_{B,H}$  for Bob, which is measured between the wire and the ground. Once the communicators select a resistor they measure the mean-squared voltage amplitude  $\langle U_{ch}^2(t) \rangle$  and or the current amplitude  $\langle I_{ch}^2(t) \rangle$ .

### 1.3. 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  $T_{cable}(N)$  for number of cables,  $T_{ke}(N)$  for number of key exchangers, and  $T_{time}(N)$  for amount of time required or speed to complete a secure bit exchange, with  $N$  representing the number of hosts in the network.

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. 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 with the order of  $N^2$

for cables and key exchangers, which makes this network impractical for very large networks. The complexities 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  $\text{FCN}_{N-1}$ . 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.** An illustration of a fully connected network with  $N - 1$  communicators per host (denoted as  $\text{FCN}_{N-1}$ ) has complexities of  $T_{\text{cable}}(N) \in O(N^2)$ ,  $T_{\text{ke}}(N) \in O(N^2)$ , and  $T_{\text{time}}(N) \in O(1)$ .

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 complexities for the fully connected network will be;  $T_{\text{cable}}(N) \in O(N^2)$ ,  $T_{\text{ke}}(N) \in O(N)$ , and  $T_{\text{time}}(N) \in O(N)$ , and will require a protocol which we will denote as  $\text{FCN}_1$  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  $\text{FCN}_1$  the protocol will need to consider the added host.

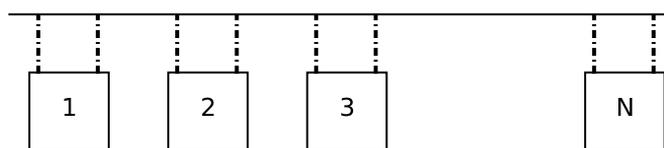
#### 1.4. Linear Chain Network with Two Key Exchangers per Host

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 4, and were analyzed in [33] in the contexts of smart grids. The linear chain network with 2 key exchangers per host has 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, say host  $i-a$  for any positive integer  $a$  and the other host is upstream, say host  $i+b$  for any positive integer  $b$  of the  $i$ th host.

The first host and the last host are special cases which cannot have simultaneous secure key exchanges with other hosts [33].

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 will be able connect with all other hosts in 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  $N$ .

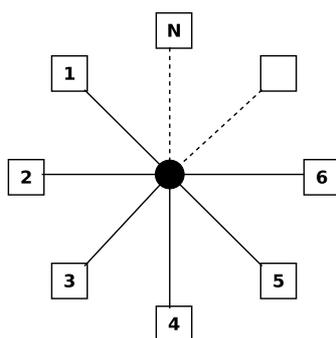


**Figure 4.** An illustration of a linear chain network with 2 key exchangers per host has complexities of  $T_{\text{cable}}(N) \in O(N)$ ,  $T_{\text{ke}}(N) \in O(N)$ , and  $T_{\text{time}}(N) \in O(N^2)$ .

## 2. Results and Discussion

### 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 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 5 is an example of a star network with  $N$  branches.



**Figure 5.** An illustration of a star network system with one key exchanger per host has complexities of  $T_{\text{cable}}(N) \in O(N)$ ,  $T_{\text{ke}}(N) \in O(N)$ , and  $T_{\text{time}}(N) \in O(N)$ .

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 complexities.

## 2.2. Graph Theory and Previous Work on the Star Network

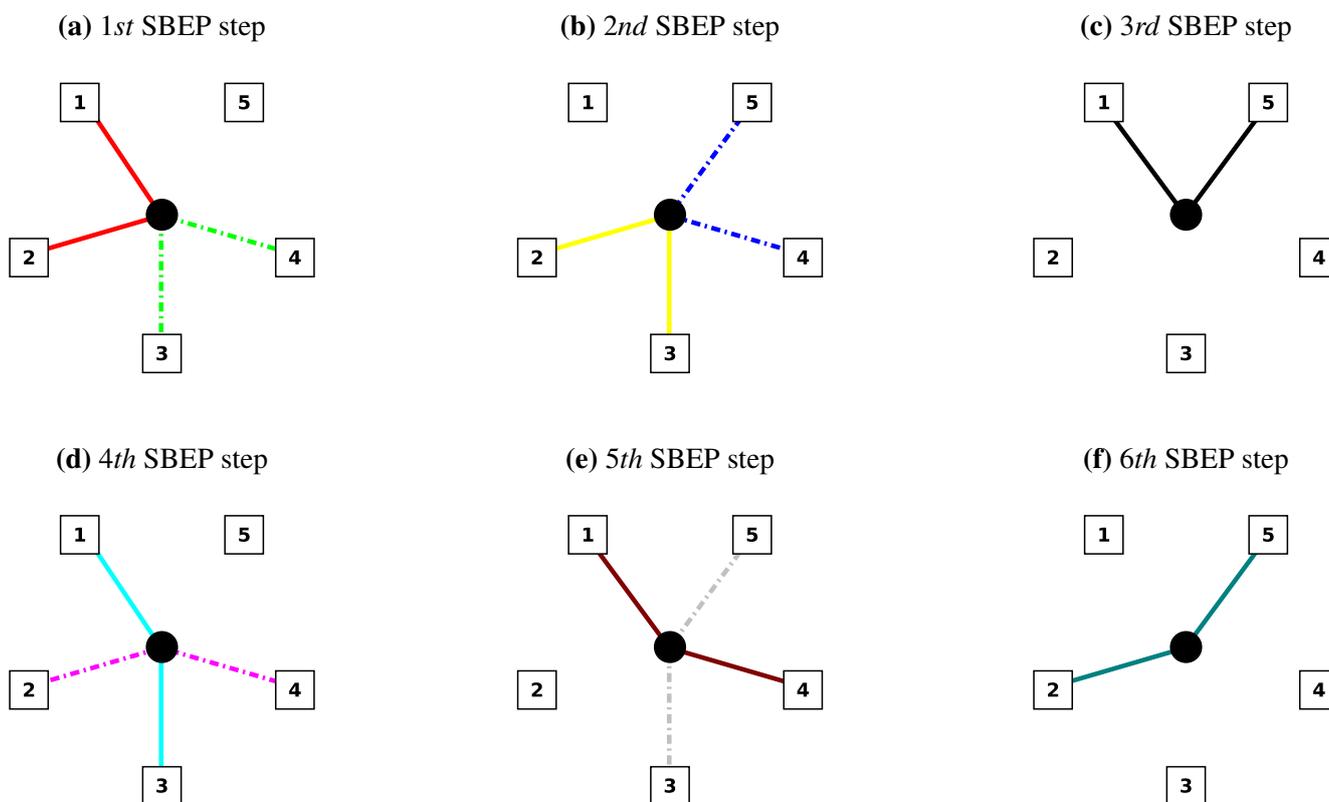
In graph theory the hosts are considered vertices and the cables are considered edges [34]. The protocol of the star network is to connect every host in the network to process a secure bit exchange with every other host in the network in the least number of Secure Bit Exchange Period (SBEP) steps. In graph theory the star network protocol can be described as a special case of an edge-color problem [35] known as round-robin (RR) tournament or all-play-all tournament problem [36]. 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 [37–43] and applied to various infrastructure networks [44–47], it has not been applied to P2P hardware-based secure key exchange networks other than [33]. Many network applications assume overlapping signals in the same channel is 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 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 is in [39].

## 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  $i$ th host with  $i$  being odd and processes a secure bit exchange with their upstream nearest 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  $i$ th host with  $i$  being even will 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  $i$ th 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  $i$ th 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 6 illustrates every step of the protocol STAR for a network with 5 hosts. The first SBEP step in the protocol is illustrated in Figure 6a, 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 Figure 6b, 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 Figure 6c. 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 Figure 6d. 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 Figure 6e, 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 Figure 6f. Since  $N$  is odd the protocol requires additional residual SBEP steps to process a secure bit exchange with the remaining hosts. Note that for 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 host. Table 1 demonstrates what every host is doing at every step in the protocol of this example as illustrated in Figure 6. Table 2 is the legend for Table 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 “ $\star$ ” means the host of this row is being utilized. The circle symbol “ $\circ$ ” means the host of this row is not active.



**Figure 6.** An illustration of the example of the star network protocol STAR 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.

**Table 1.** This table demonstrates what every host is doing at every SBEP step in the protocol STAR as described in the example and illustrated in Figure 6.

Host	(a) 1st SBEP	(b) 2nd SBEP	(c) 3rd SBEP	(d) 4th SBEP	(e) 5th SBEP	(f) 6th SBEP
1	1 → 2	○	★	1 → 3	★	○
2	★	2 → 3	○	2 → 4	○	★
3	3 → 4	★	○	★	3 → 5	○
4	★	4 → 5	○	★	4 → 1	○
5	○	★	5 → 1	○	★	5 → 2

**Table 2.** This table is the legend of Table 1.

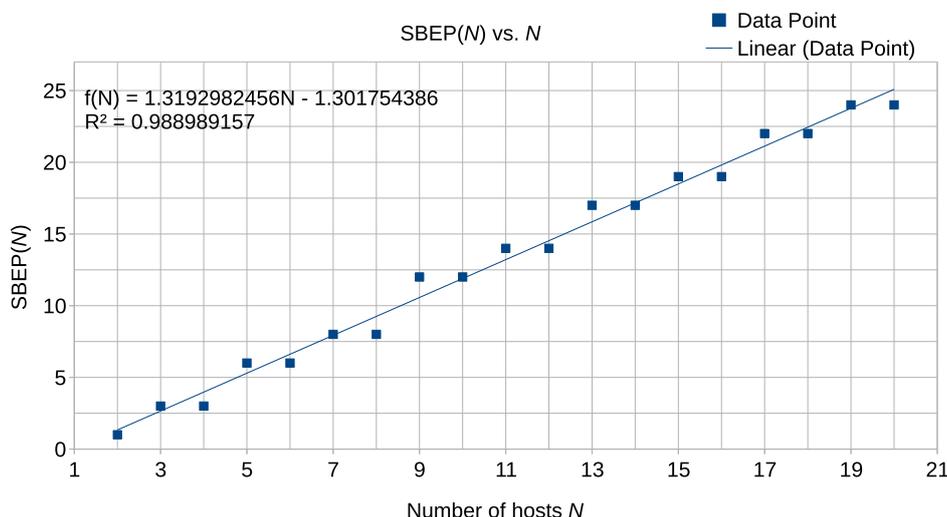
Symbol	Meaning of symbols in Table 1
$x \rightarrow y$	Host $x$ processing a secure bit exchange with host $y$ .
★	Host of this row is being utilized.
○	Host of this row is inactive.

**Table 3.** This table shows the number of SBEPs needed in star networks with 2 hosts to 20 hosts, with every host in the network to process a secure bit exchange with every other host.

$N$ , 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

The number of SBEPs needed in the STAR protocol is dependent on the number of hosts  $N$  in the network. Table 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 7 is the plot of Table 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  $R^2 = 0.988989157$ .

The patterns and relations in the star network protocol can be seen in Table 3 and Figure 7. Note that when  $N$  is evenly divisible by 2 then 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, ...,  $(N - 2)/2$ th nearest neighbor, that is host  $i+2, i+3, i+4, \dots, i+(N - 2)$  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 (1a) through (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 pairs formed in the network, which means each host share  $N - 1$  secure bits. (Note, after this sharing, each possible pairs 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.)



**Figure 7.** This is the plot of Table 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 is  $N$  meaning the number of hosts in the star network. The vertical axis is SBEP( $N$ ) meaning the number of SBEP steps needed for a network with  $N$  hosts.

$$SBEP(N) = N + \left\lceil \frac{N}{4} \right\rceil - 2 \text{ for } N \leq 8 \text{ and } N \text{ is even.} \tag{1a}$$

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

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

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

The reliability of the star network is dependent on its center switch, cable, and key exchanger. 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 hosts 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 [48,49] and inside equipment with components spread around a central processing unit such as a computer.

#### 2.4. Comparing Network Topologies

Table 4 compares 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 hosts is denoted by LCH, and the star network protocol with 1 communicator per host denoted by STAR. As can be seen from Table 4 the fastest network is the  $FCN_{N-1}$  network, the networks with the least cost of cables are the linear chain network and the star network, and the networks with the least cost 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 of cables and key exchangers.

**Table 4.** This table summarizes the complexities of the fully connected networks  $FCN_{N-1}$  and  $FCN_1$ , the linear chain network protocol LCH, and the star network protocol STAR.

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

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 weakness 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 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 hosts 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.

### 2.5. Open Questions and Future Studies

The star network has 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 is 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.

## 3. Conclusions

In this study we 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 the linear chain network. We conceived a protocol and Equations (1a) through (1d) to describe the star network. 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.

### Author Contributions

E.G., R.B., and L.K. have equal contributions.

### Conflicts of Interest

The authors declare no conflict of interest.

### References

1. Blum, J.J.; Eskandarian, A.; Hoffman, L. Challenges of intervehicle ad hoc networks. *IEEE Trans. Intell. Transport. Syst.* **2004**, *5*, 347–351, doi:10.1109/TITS.2004.838218.
2. Amin, S.M.; Wollenberg, B.F. Toward a smart grid: Power delivery for the 21st century. *IEEE Power Energ.* **2005**, *3*, 34–41, doi:10.1109/MPAE.2005.1507024.
3. Stankovic, J.A. Research Direction for the Internet of Things. *IEEE Int. Things J.* **2014**, *1*, 3–9, doi:10.1109/JIOT.2014.2312291.
4. NSA Director: China Can Damage US Power Grid. Available online: <https://www.youtube.com/watch?v=Pw79NyHleB8> (accessed on 20 November 2014). NSA director Michael Rogers addressing the House intelligence committee.
5. Bennett, C.H.; Brassard, G. Quantum cryptography: Public key distribution and coin tossing. In Proceeding of IEEE International Conference on Computers, Systems and Signal Processing, Bangalore, India, 10–19 December 1984; pp. 175–179.
6. Kish, L.B. Totally secure classical communication utilizing Johnson(-like) noise and Kirchhoff's law. *Phys. Lett.* **2005**, *352*, 178–182, doi:10.1016/j.physleta.2005.11.062.
7. Kish, L.B.; Granqvist, C.G. Elimination of a Second-Law-attack, and all cable-resistance-based attacks, in the Kirchhoff-law-Johnson-noise (KLJN) secure key exchange system. *Entropy* **2014**, *16*, 5223–5231.
8. Kish, L.B.; Granqvist, C.G. On the security of the Kirchhoff-law-Johnson-noise (KLJN) communicator. *Quantum Inf. Proc.* **2014**, *13*, 2213–2219.

9. Kish, L.B.; Abbott, D.; Granqvist, C.G. Critical analysis of the Bennett-Riedel attack on secure cryptographic key distributions via the Kirchhoff-law-Johnson-noise scheme. *PLoS One* **2013**, *8*, e81810.
10. Mingesz, R.; Kish, L.B.; Gingl, Z.; Granqvist, C.G.; Wen, H.; Peper, F.; Eubanks, T.; Schmera, G. Unconditional security by the laws of classical physics. *Metrol. Meas. Syst.* **2013**, *20*, 3–16.
11. Gingl, Z.; Mingesz, R. Noise Properties in the ideal Kirchhoff-Law-Johnson-Noise secure communication system. *PLoS One* **2014**, *9*, e96109.
12. Mingesz, R.; Vadai, G.; Gingl, Z. What kind of noise guarantees security for the Kirchhoff-Loop-Johnson-Noise key exchange? *Fluct. Noise Lett.* **2014**, *13*, doi:10.1142/S0219477514500217.
13. Hsien-Pu, C.; Kish, L.B.; Granqvist, C.G.; Schmera, G. On the “cracking” scheme in the paper “A directional coupler attack against the Kish key distribution system” by Gunn, Allison, and Abbott. *Metrol. Meas. Syst.* **2014**, *21*, 389–400, doi:10.2478/mms-2014-0033.
14. Hsien-Pu, C.; Kish, L.B.; Granqvist, C.G.; Schmera, G. Do electromagnetic waves exist in a short cable at low frequencies? What does physics say? *Fluct. Noise Lett.* **2014**, *13*, doi:10.1142/S0219477514500163.
15. Kish, L.B.; Gingl, Z.; Mingesz, R.; Vadai, G.; Smulko, J.; Granqvist, C.G. Analysis of an attenuator artifact in a experimental attack by Gunn-Allison-Abbott against the Kirchhoff-law-Johnson-noise (KLJN) secure key exchange system. *Fluct. Noise Lett.* **2014**, <http://arxiv.org/abs/1411.0818>; <http://vixra.org/abs/1410.0122>.
16. Kish, L.B.; Kwan, C. Physical Uncloneable Function Hardware Keys Utilizing Kirchhoff-Law-Johnson-Noise Secure Key Exchange and Noise-Based Logic. *Fluct. Noise Lett.* **2013**, *12*, 1350018.
17. Kish, L.B. Enhanced secure key exchange systems based on the Johnson-noise scheme. *Metrol. Meas. Syst.* **2013**, *20*, 191–204.
18. Mingesz, R.; Kish, L.B.; Gingl, Z.; Granqvist, C.G.; Wen, H.; Peper, F.; Eubanks, T.; Schmera, G. Unconditional security by the laws of classical physics. *Metrol. Meas. Syst.* **2013**, *20*, 3–16.
19. Kish, L.B.; Scheuer, J. Noise in the wire: the real impact of wire resistance for the Johnson (-like) noise based secure communicator. *Phys. Lett. A* **2010**, *374*, 2140–2142.
20. Kish, L.B.; Horvath, T. Notes on Recent Approaches Concerning the Kirchhoff-Law-Johnson-Noise-based Secure Key Exchange. *Phys. Lett. A* **2009**, *373*, 2858–2868.
21. Kish, L.B.; Saidi, O. Unconditionally secure computers, algorithms and hardware, such as memories. *Fluct. Noise Lett.* **2008**, *8*, L95–L98.
22. Mingesz, R.; Gingl, Z.; Kish, L.B. Johnson(-like)-Noise-Kirchhoff-loop based secure classical communicator characteristics, for ranges of two to two thousand kilometers, via model-line. *Phys. Lett. A* **2008**, *372*, 978–984.
23. Kish, L.B. Response to Feng Hao’s paper “Kish’s Key Exchange Scheme is Insecure”. *Fluct. Noise Lett.* **2006**, *6*, C37–C41.
24. Kish, L.B. Protection against the man-in-the-middle-attack for the Kirchhoff-loop-Johnson(-like)-noise cipher and expansion by voltage-based security. *Fluct. Noise Lett.* **2006**, *6*, L57–L63.

25. Kish, L.B.; Mingesz, R. Totally secure classical networks with multipoint telecloning (teleportation) of classical bits through loops with Johnson-like noise. *Fluct. Noise Lett.* **2006**, *6*, C9–C21.
26. Kish, L.B.; Peper, F. Information Networks Secured by the Laws of Physics. *IEICE Trans. Fund. Comm. Elec. Infor. Syst.* **2012**, *95*, 1501–1507.
27. Liang, Y.; Poor, H.V.; Shamai, S. Information theoretic security. *Foundations Trends Commun. Inform. Theory* **2008**, *5*, 355–580, doi:10.1561/01000000036.
28. Schollmeier, R. A definition of Peer-to-Peer Networking for the Classification of Peer-to-Peer Architectures and Applications. In Proceedings of IEEE First International Conference on Peer-to-Peer Computing, Linköping, Sweden, 27–29 August 2001; pp. 101–102.
29. Yuen, H.P. On the Foundations of Quantum Key Distribution—Reply to Renner and Beyond. **2012**, arXiv:1210.2804 .
30. Makarov, V. Controlling passively quenched single photon detectors by bright light. *New J. Phys.* **2009**, *11*, doi:10.1088/1367-2630/11/6/065003.
31. Gerhardt, I.; Liu, Q.; Lamas-Linares, A.; Skaar, J.; Kurtsiefer, C.; Makarov, V.; Full-field implementation of a perfect eavesdropper on a quantum cryptography system. *Nat. Commun.* **2011**, *2*, doi:10.1038/ncomms1348.
32. Lydersen, L.; Jain, N.; Wittmann, C.; Maroy, O.; Skaar, J.; Marquardt, C.; Makarov, V.; Leuchs, G. Superlinear threshold detectors in quantum cryptography. *Phys. Rev. A* **2011**, *84*, doi:10.1103/PhysRevA.84.032320.
33. Gonzalez, E.; Kish, L.B.; Balog, R.S.; Enjeti, P. Information Theoretically Secure, Enhanced Johnson Noise Based Key Distribution over the Smart Grid with Switched Filters. *PLoS One* **2013**, *8*, doi:10.1371/journal.pone.0070206.
34. Tutte, W.T. *Graph Theory*; Cambridge University Press: Cambridge, UK, 2001; p. 30.
35. Alexander, S. *The Mathematical Coloring Book*; Springer: New York, NY, USA, 2008.
36. Lucas, E. *Récréations Mathématiques*; Gauthier-Villars: Paris, France, 1883; pp. 161–197.
37. Tutte, W.T. The factorization of linear graphs. *J. Lond Math. Soc.* **1947**, *22*, 107–111.
38. Tutte, W.T. The factors of graphs. *Canad. J. Math.* **1952**, *4*, 314–328.
39. Akiyama, J.; Kano, M. Path Factors of a Graph. *Graph Theory and its Applications* **1984**, 11–12.
40. Bezegová, L.; Lužar, B.; Mockovčiaková, M.; Soták, R.; Škrekovski, R. Star edge coloring of some classes of graphs. *J. Graph. Theor.* **2015**, doi:10.1002/jgt.21862.
41. Meng, K.K.; Fengming, D.; Guan, T.E. *Introduction to Graph Theory: H3 Mathematics*; World Scientific: Hackensack, NJ, USA, 2007.
42. Hsu, L.H.; Lin, C.K. *Graph Theory and Interconnection Networks*; CRC press: Boca Raton, FL, USA, 2009.
43. Gross, J.L.; Yellen, J. *Handbook of Graph Theory*; CRC press: Boca Raton, FL, USA, 2004; pp. 341–445.
44. Huang, S.; Dutta, R.; Rouskas, G.N. Traffic Grooming in Path, Star, and Tree Networks: Complexity, Bounds, and Algorithms. *IEEE J. Selected Areas Comm.* **2006**, *24*, doi:10.1109/JSAC.2006.1613773.

45. Roberts, L.G.; Wessler, B.D. Computer network development to achieve resource sharing. *Proc. AFIPS*. **1970**, 543–549, doi:10.1145/1476936.1477020.
46. Adamy, U.; Erlebach, T.; Mitsche, D.; Schurr, I.; Speckmann, B.; Welzl, E. Off-line admission control for advance reservations in star networks. *Approx. Online Algorithms* **2005**, 3351, 211–224.
47. Chan, S.P. *Network Topology and Its Engineering Applications*; National Taiwan University Press: Taipei, Taiwan, 1975.
48. Saez, Y.; Cao, X.; Kish, L.B.; Pesti, G. Securing Vehicle Communication Systems by the KLJN Key Exchange Protocol. *Fluct. Noise Lett.* **2014**, *13*, 1450020.
49. Cao, X.; Saez, Y.; Pesti, G.; Kish, L.B. On KLJN-based secure key distribution in vehicular communication networks. *Fluct. Noise Lett.* **2015**, *14*, doi:10.1142/S021947751550008X.
50. Ayhan, B.; Kwan, C.; Zhou, J.; Kish, L.B.; Benkstein, K.D.; Rogers, P.H.; Semancik, S. Fluctuation enhanced sensing (FES) with a nanostructured, semiconducting metal oxide film for gas detection and classification. *Sensors Actu. B* **2013**, *88*, 651–600.
51. Saez, Y.; Kish, L.B. Errors and their mitigation at the Kirchhoff-law-Johnson-noise secure key exchange. *PLoS One* **2013**, *8*, e81103.

© 2015 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/4.0/>).