Hindawi Publishing Corporation
The Scientific World Journal
Volume 2013, Article ID 935604, 13 pages
http://dx.doi.org/10.1155/2013/935604

Research Article

A Multihop Key Agreement Scheme for Wireless Ad Hoc Networks Based on Channel Characteristics

Zhuo Hao,[1] Sheng Zhong,[2,3] and Nenghai Yu[4]

[1] *MicroStrategy, Hangzhou, China*
[2] *State Key Laboratory of Novel Software Technology, Nanjing University, Nanjing 210023, China*
[3] *Department of Computer Science and Technology, Nanjing University, Nanjing 210023, China*
[4] *Department of Electronic Engineering and Information Science, University of Science and Technology of China, Hefei, Anhui 230027, China*

Correspondence should be addressed to Sheng Zhong; sheng.zhong@gmail.com

Received 26 March 2013; Accepted 30 April 2013

Academic Editors: A. Miné and A. Paun

A number of key agreement schemes based on wireless channel characteristics have been proposed recently. However, previous key agreement schemes require that two nodes which need to agree on a key are within the communication range of each other. Hence, they are not suitable for multihop wireless networks, in which nodes do not always have direct connections with each other. In this paper, we first propose a basic multihop key agreement scheme for wireless ad hoc networks. The proposed basic scheme is resistant to external eavesdroppers. Nevertheless, this basic scheme is not secure when there exist internal eavesdroppers or Man-in-the-Middle (MITM) adversaries. In order to cope with these adversaries, we propose an improved multihop key agreement scheme. We show that the improved scheme is secure against internal eavesdroppers and MITM adversaries in a single path. Both performance analysis and simulation results demonstrate that the improved scheme is efficient. Consequently, the improved key agreement scheme is suitable for multihop wireless ad hoc networks.

1. Introduction

Network security (see, e.g., [1, 2]) has been studied extensively. In wireless networks, security problems are especially critical, because wireless channels are inherently broadcast channels. When a pair of nodes communicate with each other, nearby nodes within the communication range may be able to overhear their messages. In order to prevent eavesdropping, messages are often encrypted before being sent. Hence, key agreement is of great importance for security of wireless networks.

Recently, Mathur et al. [3] propose a novel key agreement scheme for wireless networks, which is based on the secrecy of the wireless channel itself. In their scheme, the two communicating nodes send probe signals to each other and measure the channels. Then, they extract secret bits from the channel measurements using a level-crossing algorithm. Because of the reciprocity of the channel, the two nodes can extract the same key from their own channel measurements. Any eavesdroppers that are more than half a wavelength away from both nodes can get no knowledge of the key, because their experienced channels are independent of the channel between the two communicating nodes. The broad applicability of this security alternative has been validated by Jana et al. [4], through a series of experiments in real environments.

However, both Mathur et al.'s and Jana et al.'s schemes require that two nodes are within the communication range of each other in order to establish a key. This requirement cannot always be satisfied. In many realistic scenarios, intermediate nodes are needed for relaying messages, because the end nodes cannot communicate directly.

In this paper, we show that it is feasible to build key agreement schemes based on wireless channel measurements in *multihop* wireless networks. We show that, by extracting secrets from the phase characteristics (it is feasible to extract secrets from phase characteristics—please see Section 3 for details) of channels, two end nodes that are more than one

hop away from each other can establish a key between them. We propose a basic key agreement scheme for this purpose and show that it is secure against external eavesdroppers (i.e., eavesdroppers out of the paths connecting the two nodes). After that, we show that the basic scheme is subject to internal eavesdropping and Man-in-the-Middle (MITM) attacks. Therefore, we propose an improved key agreement scheme to prevent these two attacks. The improved scheme is based on the assumption that the network is biconnected. The secrets are extracted from two disjoint paths between the two end nodes. The improved scheme is secure against internal eavesdroppers and MITM adversaries in a single path. (Please see Section 5.3, Remark 7 for the possibility that adversaries control more than a single path.) In both the basic and the improved schemes, we follow the standard assumption [3–6] that adversaries are more than half a wavelength away from all the participating nodes. We give a theoretical analysis of the key agreement probability and show that it is affected by communication SNRs, sampling rates, and quantization parameters. We simulate the improved scheme in GlomoSim [7] and show that the established key has strong randomness and the key agreement efficiency is high.

In summary, we have the following contributions.

(i) We propose a basic multihop key agreement scheme and prove that it is secure against external eavesdroppers.

(ii) Since the basic scheme is not secure against internal eavesdroppers or MITM adversaries, we propose an improved multihop key agreement scheme, and prove that this improved scheme is secure against internal eavesdroppers and MITM adversaries in a single path between the two nodes.

(iii) We give both performance analysis and simulation results of the improved scheme. The results show that the improved scheme is very efficient and the established key has strong randomness.

The rest of this paper is organized as follows. In Section 2, we review the related work. In Section 3, we present technical preliminaries. In Section 4, we present the basic multihop key agreement scheme and give a security analysis. In Section 5, we describe the improved multihop key agreement scheme and prove its security. In Sections 6 and 7, we show that the improved scheme is efficient by both theoretical analysis and simulation results. Finally, we conclude in Section 8.

2. Related Work

Key agreement based on channel characteristics is firstly proposed in Hershey et al. [8], in which the secret key is extracted from the phase differences of continuous waves. After that, Hassan et al. [9] propose to use phase differences between two orthogonal subcarriers as extracted secrets. Tope and McEachen [10] propose a key generation scheme based on polarity of power envelope differences. Recently, a lot of schemes [3–6, 11–22] are proposed to enhance the security and/or improve the performance. In particular, Mathur et al. [3] propose a scheme to extract secret bits from wireless channel measurements. They design a level-crossing algorithm to increase the bit consistency rate. They do experiments using both customized 802.11 platform and off-the-shelf 802.11 network cards. In order to validate the effectiveness of the key extraction schemes based on signal strengths, Jana et al. [4] carry out extensive experiments in various environments. They propose adaptive quantization method to improve the performance. Patwari et al. [6] propose a high-rate uncorrelated bit extraction scheme based on fractional interpolation, decorrelation transformation and multibit adaptive quantization. Ye et al. [5] propose a secret key extraction approach that is suited for more general channel state distributions. Zhang et al. [21] find that mobility patterns have important impact on the correlation of channel measurements at the end nodes. They show that more diffusion in the mobility brings less correlation in the measured channel impulse responses. Gollakota and Katabi [23] propose a secret communication method based on receiver's jamming. Their method eliminates the reliance on channel variance and has high secret communication speed.

There are also many analytical works [24–27] that provide theoretical analysis of secret key exchange protocols and propose improved algorithms. In addition, secret key extraction schemes from UWB (Ultra-WideBand) channels are proposed in [28–31]. Croft et al. [32] propose a secret bit extraction scheme for wireless sensors, while Ali et al. [33] develop a key extraction approach in body area networks.

It is important to note that all the previous approaches focus on one single channel between two nodes. Therefore, they have the requirement that the two nodes are within the communication range of each other. In contrast, in this paper, we propose schemes that are suitable for multihop networks, in which nodes can be out of the communication range of each other. Consequently, our proposed schemes can be used for key agreement in multihop wireless networks.

Recently Wang et al. [34] propose a group key agreement scheme in wireless networks. Wang et al.'s scheme is based on the phase characteristics of wireless channels. They use phase randomness for bit generation and remove the reliance on the node mobility. According to Ren et al. [35], phase-based methods [8, 9, 16, 34] have three advantages compared to RSS-based methods [3–6], including having uniform distribution, providing high resolution phase estimation, and enabling phase accumulation across multiple nodes. Similar to [34], the schemes proposed in this paper are also based on channel phase randomness. However, our proposed schemes consider a completely different setting, in which the involved nodes can be more than one hop away from each other. In fact, allowing nodes to be multiple hops away from each other is a major technical challenge addressed in this paper. Hence, our schemes are independent from, and complementary to, the results in [34].

3. Technical Preliminaries

In a typical multihop mobile ad hoc network, there are no infrastructures. Each node is both an end host and a router. Denote the nodes in the network by $\{N_1, N_2, \ldots, N_a\}$. If node

N_i is within the communication area of N_k, then we say N_i is a neighbor of N_k. Without loss of generality, we assume that wireless channels are symmetric; that is, whenever a node N_i is a neighbor of N_k, N_k is also a neighbor of N_i. Just as in previous work [3, 4], we assume the channel between any two neighboring nodes to be reciprocal. (This assumption implies that our work is most suitable for a homogeneous network. If the network is heterogeneous, then our work needs to be modified before it can be applied.) Denote the channel from N_i to N_k by $h_{ik}(t)$, and denote the channel from N_k to N_i by $h_{ki}(t)$. Then the channel reciprocity indicates that $h_{ik}(t) = h_{ki}(t)$ for any time t.

We use the phase characteristics of both the initial signals and the channel as a random source to extract the shared secret key from. (Note that using the channel phase characteristics as a source of randomness is a feasible approach, which has been adopted in existing work, e.g., [34]. A possible way to implement this can be found in [35].) From the channel reciprocity, we know that within the channel coherence time, the channel between two nodes can be assumed to be invariant. We divide the channel coherence time to equal time slots: $T_1, T_2, \ldots, T_d$. Let the length of each time slot be TS, and denote the coherence time of the channel by CT. Let $d = \lfloor CT/TS \rfloor$.

During one time slot T_k, when N_i sends the initial signal to N_j, we denote the signal sent from N_i by $s_i(t)$. $s_i(t)$ has the following representation:

$$s_i(t) = C_i(t)\, e^{j(\omega_c(t-t_0)+\phi(t))}. \quad (1)$$

In (1), $C_i(t)$ is the amplitude of $s_i(t)$. ω_c and $\phi(t)$ are the center frequency and the initial phase of $s_i(t)$, respectively. We emphasize that it is *feasible* to send a signal with a given phase $\phi(t)$—in fact, some existing schemes like [34] already include such operations. In order to implement such an operation, one can use analog-to-digital converters [35].

Definition of Adversaries. In this paper, we consider three different kinds of adversaries: *internal eavesdropper*, *external eavesdropper*, and *MITM adversary*. Here both internal eavesdroppers and external eavesdroppers refer to *passive* adversaries that eavesdrop messages and attempt to figure out the established key. The difference between these two types of adversaries is that an internal eavesdropper is an intermediate node in a path selected for transmitting messages for key agreement, while an external eavesdropper is not an intermediate node in any such path. Unlike these two types of passive adversaries, an MITM adversary is an *active* adversary who controls one or more node in a path selected for transmitting messages for key agreement and carries out an MITM attack. A little more formally, we have the following definitions.

Definition 1. A multihop key agreement scheme is secure against a set of external eavesdroppers if, assuming all involved nodes follow the protocol faithfully, all signals overheard by this set of eavesdroppers are statistically independent from the final key generated by this scheme.

Definition 2. A multihop key agreement scheme is secure against a set of internal eavesdropper if, assuming all involved nodes follow the protocol faithfully, all packets received by this set of eavesdroppers, together with all signals overheard by this set of eavesdropper, are statistically independent from the final key generated by this scheme.

Definition 3. A multihop key agreement scheme is secure against a set of MITM adversaries if, assuming all involved nodes except this set of MITM adversaries follow the protocol faithfully, the final keys different nodes obtain are consistent; furthermore, all packets received by this set of MITM adversaries, together with all signals overheard by this set of adversary, are statistically indepedent from the final key generated by this scheme.

4. The Basic Multihop Key Agreement Scheme

In this section, we propose a basic multihop key agreement scheme. The basic scheme is built on one selected path between the two nodes that want to agree on a secret key. It is secure against any external eavesdroppers as long as those eavesdroppers are more than half a wavelength away from all the nodes in the selected path.

4.1. Scheme Outline. The basic idea of this multihop key agreement scheme is to use both the channel phase characteristics of the selected path and the randomly selected initial phases to extract common secrets (i.e., secrets known only to A and B). By using quantization, these common secrets are quantized into common secret bits. After that, information reconciliation and privacy amplification are used [36–38] on the common secret bits, so that a secret key can be generated. When the external eavesdroppers are more than half a wavelength away, they will experience channels that are independent of the channels in the selected path [3, 4].

In order to have k common secret bits, the two parties (denoted by A and B) need to interact with each other for $\lceil k/q \rceil$ rounds, assuming in each round that they can get q bits from quantization. In each round, A picks a random phase value, and sends an initial signal with this initial phase value to B using the selected path. Each intermediate node in this path estimates the phase of the signal received from its antecedent node and sends a new signal with this estimated phase to its subsequent node. Note that A is the first node in the path, and B is the last node in the path. Hence, A has a subsequent node only, and B has an antecedent node only. After B receives the signal from its antecedent node, it picks a random phase value and sends an initial signal with this initial phase value back to A, along the reverse path. Each intermediate node estimates the phase of the signal received from its subsequent node, and sends a new signal with the estimated phase to its antecedent node. Finally B (resp., A) estimates the phase of the signal received from its antecedent (resp., subsequent) node and adds the estimated phase with its randomly generated initial phase. The sums generated by A and B both reflect characteristics of all the channels in the path and the random initial phase values picked by A and B.

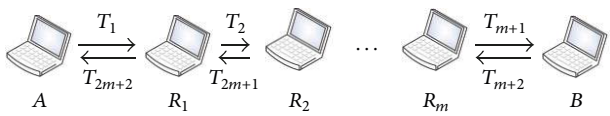

Figure 1: Illustration of signal transmission in one round.

In order to make sure that they are highly correlated, each round is completed within the channel coherence time. The random initial phase values picked by A and B are sources of randomness of the extracted common secrets.

After extracting common secrets from the channels and the random initial phase values, A and B perform independent quantization on these secrets and get common secret bits. The discrepancies between common secret bits of A and B are corrected by information reconciliation. The lost entropy of performing the information reconciliation is reduced by privacy amplification. In the following, we give detailed descriptions of these steps. After that, we give analysis of the basic scheme.

4.2. Common Secret Extraction. The common secret extraction consists of $\lceil k/q \rceil$ rounds, and each round contains $(2m+2)$ time slots. Figure 1 illustrates the signal transmission involved in one round.

In the following, we describe steps involved in one round.

(1) In the time slot $T_1 = [0, 0 + TS]$, A sends the initial signal $s_a(t)$ with phase ϕ_1 to R_1, where the value of ϕ_1 is randomly picked by A from $[0, 2\pi)$ (and thus *known to A*). Without loss of generality, we assume that $s_a(t)$ has a unit power level. Denote the signal received at R_1 by r_{A,R_1}. Then we get that $r_{A,R_1}(t) = \alpha_{A,R_1}(t)e^{j(\omega_c t+\phi_{A,R_1})} + n_{R_1}(t)$, where $\alpha_{A,R_1}(t)$ and ϕ_{A,R_1} denote the amplitude and phase of the signal received from A, and $n_{R_1}(t)$ denotes the receiver noise at R_1.

(2) The phase of $r_{A,R_1}(t)$ is $\phi_{A,R_1} = \phi_1 + \psi_{A,R_1}$, in which ψ_{A,R_1} denotes the phase offset of the channel between A and R_1. R_1 computes the estimate of ϕ_{A,R_1}, which we denote by $\widehat{\phi}_{A,R_1}$. After that in T_2, R_1 sends a unit signal to R_2 whose phase is tuned to $\widehat{\phi}_{A,R_1}$.

(3) For $i = 2, 3, \ldots, m-1$, in the time slot T_{i+1}, R_i computes the phase estimate of the signal received from R_{i-1} and sends a new unit signal with this phase estimate to R_{i+1}. In T_{m+1}, R_m sends the signal $e^{j(w_c(t-m\cdot TS)+\widehat{\phi}_{A,R_m})}$ to B.

(4) In the time slot T_{m+2}, B sends the initial signal $s_b(t)$ with phase ϕ_2 to R_m, where $s_b(t)$ also has a unit power level, and ϕ_2 is picked randomly by B from $[0, 2\pi)$ (and thus *known to B*). Denote the signal received at R_m by r_{B,R_m}. Then $r_{B,R_m}(t) = \alpha_{B,R_m}(t)e^{j(\omega_c(t-(m+1)\cdot TS)+\phi_{B,R_m})} + n_{R_m}(t)$. The phase of $r_{B,R_m}(t)$ is $\phi_{B,R_m} = \phi_2 + \psi_{B,R_m}$, in which ψ_{B,R_m} denotes the phase of the channel between B and R_m.

(5) For $i = m+3, m+4, \ldots, 2m+1$, in T_i, R_{2m+3-i} sends the signal $e^{j(w_c(t-(i-1)\cdot TS)+\widehat{\phi}_{B,R_{2m+3-i}})}$ to R_{2m+2-i}. In T_{2m+2}, R_1 sends the signal $e^{j(w_c(t-(2m+1)\cdot TS)+\widehat{\phi}_{B,R_1})}$ to A.

(6) From the previous steps B receives $r_{R_m,B}$, and A receives $r_{R_1,A}$. It is easy to see that

$$r_{R_m,B}(t) = \alpha_{R_m,B}(t)\, e^{j(w_c(t-m\cdot TS)+\phi_{A,B,1})} + n_B(t),$$
$$r_{R_1,A}(t) = \alpha_{R_1,A}(t)\, e^{j(w_c(t-(2m+1)\cdot TS)+\phi_{B,A,1})} + n_A(t), \quad (2)$$

where $\phi_{A,B,1}$ and $\phi_{B,A,1}$ denote the signal phases of $r_{R_m,B}$ and $r_{R_1,A}$, respectively. B computes $I_B = (\widehat{\phi}_{A,B,1} + \phi_2)\mathsf{mod}2\pi$, and A computes $I_A = (\widehat{\phi}_{B,A,1} + \phi_1)\mathsf{mod}2\pi$. From I_B and I_A, B and A extract common secret bits.

We denote such a round by Round(A, B, m). Apparently Round(A, B, m) needs to take $(2m+2)$ time slots.

From the previous protocol process, we can get that $I_B = (\widehat{\phi}_{A,B,1} + \phi_2)\mathsf{mod}2\pi = \{\mathsf{est}(\phi_1 + \psi_{A,R_1} + \sum_{i=1}^{m-1}\psi_{R_i,R_{i+1}} + \psi_{R_m,B}) + \phi_2\}\mathsf{mod}2\pi$ and $I_A = (\widehat{\phi}_{B,A,1} + \phi_1)\mathsf{mod}2\pi = \{\mathsf{est}(\phi_2 + \psi_{B,R_m} + \sum_{i=1}^{m-1}\psi_{R_{i+1},R_i} + \psi_{R_1,A}) + \phi_1\}\mathsf{mod}2\pi$. From the channel reciprocity, I_B and I_A are highly correlated if the measurements are within the channel coherence time. Hereafter, suppose that A and B carry out z rounds of Round(A, B, m), and denote the extracted secret vectors by $[I_{A,1}, I_{A,2}, \ldots, I_{A,z}]$ and $[I_{B,1}, I_{B,2}, \ldots, I_{B,z}]$, respectively.

4.3. Quantization. After z rounds of common secret extraction, A has got the secret vector $[I_{A,1}, I_{A,2}, \ldots, I_{A,z}]$, and B has got the secret vector $[I_{B,1}, I_{B,2}, \ldots, I_{B,z}]$. For $Z \in \{A, B\}$ and $k = 1, 2, \ldots, z$, $I_{Z,k}$ is in the range of $[0, 2\pi)$. Now A and B quantize each value in their vectors into common secret bits. Specifically, we divide the interval $[0, 2\pi)$ into q equal subintervals. Denote these subintervals by $[0, 2\pi/q), [2\pi/q, 4\pi/q), \ldots, [2(q-1)\pi/q, 2\pi)$. We quantize each subinterval into $\log_2(q)$ bits using the Gray code [39]. By using Gray code, adjacent subintervals have only one bit discrepancy after quantization, which reduces the number of bit errors caused by estimation errors.

Denote the length of the targeted secret key by k. In order to generate the key, A and B need to interact with each other for at least $\lceil k/q \rceil$ rounds.

4.4. Information Reconciliation and Privacy Amplification. Because there exist noises and interferences at the receivers, A and B can get discrepancies at some common secret bits. They can achieve secret bits reconciliation by transmitting error correcting information through a public channel, which is called information reconciliation [40, 41]. We use the classic Cascade protocol [40] to perform reconciliation between the extracted secret bits. For completeness we briefly review the Cascade protocol.

Denote the two secret bit strings at A and B by BS_A and BS_B. In the Cascade protocol, each of the two bit strings

are divided into disjoint blocks. One party sends the parity values of all the blocks to the other party. If an odd number of errors are found within any block, A and B perform an interactive binary error search on that block, until one bit error is corrected. The Cascade protocol consists of several rounds, depending on the rate of bit discrepancies between BS_A and BS_B. If in the kth ($k \geq 2$) round, one error is corrected at the ith bit, and then any other block that contains the ith bit also contain an odd number of errors, which need to be corrected subsequently. Only minimal information gets leaked out if the number of rounds and the block size are selected appropriately.

After the information reconciliation, privacy amplification [36–38] is used to reduce the side information leaked during information reconciliation. We use the following 2-universal hash family [4]:

$$\begin{aligned} g_{a,b}(x) &= (ax+b) \bmod p_M, \\ h_{a,b}(x) &= g_{a,b}(x) \bmod m, \quad x \in \{1,2,\ldots,M\}, \\ a &\in [1, p_M - 1], \qquad b \in [0, p_M - 1], \end{aligned} \tag{3}$$

where p_M is a prime number that satisfies $p_M > M$. This 2-universal hash family consists of all the functions h that map from $\{1,2,\ldots,M\}$ to $\{0,1\}^m$. One party randomly selects a and b and sends them to the other party. We divide the secret bits after reconciliation into blocks of $\log_2(M)$ bits, and m is decided based on the required secret key length.

After these two processes, the generated keys at A and B are cryptographic secure keys. A and B can use the generated key for secret communications.

4.5. Security Analysis of the Basic Scheme. In this section, we present a security analysis of the basic scheme. Firstly we argue that the basic scheme is secure against any external eavesdroppers that are more than half a wavelength away from all the nodes in the selected path. Secondly we show that threats from internal adversaries can affect the security of the scheme. Finally we show that MITM attack is possible in the basic scheme. (Recall that internal eavesdroppers, external eavesdroppers, and MITM adversary are defined at the end of Section 3.)

4.5.1. Security against Any External Eavesdropper. If all the external eavesdroppers are more than half a wavelength away from all the nodes in the selected path, then their experienced channels are independent of channels between nodes in the selected path.

In the following we analyze the security of the basic scheme when there exists only one external eavesdropper. The analysis can be similarly extended to the case in which there are more than one eavesdroppers. In Figure 2, denote

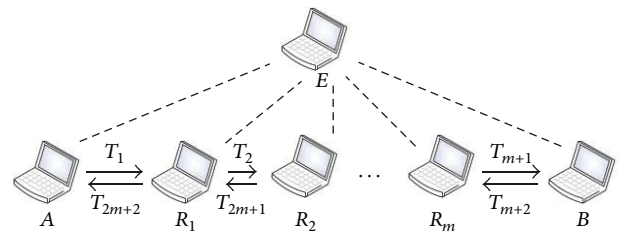

Figure 2: Illustration of one external eavesdropper in the basic scheme.

the eavesdropper by E. From Round(A,B,m), E gets the following estimated phases from its received signals:

$$\begin{gathered} \mathsf{est}\left(\phi_1 + \psi_{A,E}\right) \\ \mathsf{est}\left(\phi_1 + \psi_{A,R_1} + \sum_{i=1}^{k-1} \psi_{R_i,R_{i+1}} + \psi_{R_k,E}\right), \quad k \in [1,m] \\ \mathsf{est}\left(\phi_2 + \psi_{B,E}\right) \\ \mathsf{est}\left(\phi_2 + \psi_{B,R_m} + \sum_{i=k}^{m-1} \psi_{R_{i+1},R_i} + \psi_{R_k,E}\right), \quad k \in [1,m]. \end{gathered} \tag{4}$$

In (4), E gets $\mathsf{est}(\phi_1 + \psi_{A,E})$ at T_1 from A and gets $\mathsf{est}(\phi_1 + \psi_{A,R_1} + \sum_{i=1}^{k-1} \psi_{R_i,R_{i+1}} + \psi_{R_k,E})$ at T_{k+1} from R_k, $k \in [1,m]$. On the other hand, E gets $\mathsf{est}(\phi_2 + \psi_{B,E})$ at T_{m+2} from B and gets $\mathsf{est}(\phi_2 + \psi_{B,R_m} + \sum_{i=k}^{m-1} \psi_{R_{i+1},R_i} + \psi_{R_k,E})$ at T_{m+2+k} from R_{m+1-k}, $k \in [1,m]$.

Because ϕ_1 and ϕ_2 are randomly selected by A and B, respectively, these estimated phases are also random. Because $\psi_{A,E}$ is independent of ψ_{A,R_1}, E cannot get any knowledge of $(\phi_1 + \psi_{A,R_1})$ from $\mathsf{est}(\phi_1 + \psi_{A,E})$. Similarly, E cannot get any knowledge of $(\phi_1 + \psi_{A,R_1} + \sum_{i=1}^{k-1} \psi_{R_i,R_{i+1}} + \psi_{R_k,B})$, $k \in [1,m]$ from $\mathsf{est}(\phi_1 + \psi_{A,R_1} + \sum_{i=1}^{k-1} \psi_{R_i,R_{i+1}} + \psi_{R_k,E})$, $k \in [1,m]$. Finally, during the channel coherence time, no probe signals are transmitted between the nodes in the selected path, so $\psi_{A,E}$, $\psi_{R_k,E}$, $k \in [1,m]$ and $\psi_{B,E}$ are unknown to E. Therefore, from these estimated phase values, E gets no knowledge of the extracted secrets at A or B.

We stress that it is realistic to assume that the external eavesdroppers are at least half a wavelength away. When the carrier frequency is 2.437 GHz (one of the frequency band of 802.11 b), the wavelength of the carrier is $(3 \cdot 10^8 \text{ m/s})/(2.437 \cdot 10^9 \text{ Hz}) \approx 0.12$ m. Half a wavelength is only about 6 centimeters. Within such a distance, it is hard for an eavesdropper to avoid being detected.

4.5.2. Threats of Internal Adversaries. In the basic scheme, each of the internal nodes can get the complete knowledge of the extracted secrets at A and B. If one of them is corrupted, then the scheme is not secure. For example, if R_k is corrupted, based on its received signals from R_{k-1} and R_{k+1}, it gets $\widehat{\phi}_{A,R_k} = \mathsf{est}(\phi_1 + \psi_{A,R_1} + \sum_{i=1}^{k-1} \psi_{R_i,R_{i+1}})$ and $\widehat{\phi}_{B,R_k} = \mathsf{est}(\phi_2 + \sum_{i=k}^{m-1} \psi_{R_{i+1},R_i} + \psi_{B,R_m})$. By adding up these two values, R_k gets an estimate, which is highly correlated to both I_B and I_A.

Therefore, if one of the intermediate nodes is corrupted, the basic scheme is not secure.

4.5.3. MITM Attack. Because there are m intermediate nodes between A and B, any of them can carry out an MITM attack. Suppose that R_k intends to carry out an MITM attack and establish two different keys with A and B, respectively. Specifically, R_k agrees on one key with A, based on the subpath $A \to R_1 \to \cdots \to R_k$; R_k agrees on another key with B, based on the other subpath $R_k \to R_{k+1} \to \cdots \to B$. The MITM attack consists of the following steps:

(1) In each round, R_k performs the following steps:

(a) When R_k receives the signal $r_{R_{k-1},R_k} = \alpha_{R_{k-1},R_k}(t)e^{j(w_c(t-(k-1)\cdot TS)+\phi_{A,R_k})}$ from R_{k-1}, it picks a random value $\phi_{k,1} \in [0, 2\pi)$ and sends $s_{k,1}(t) = e^{j(w_c(t-k\cdot TS)+\phi_{k,1})}$ to R_{k+1}.

(b) When R_k receives the signal $r_{R_{k+1},R_k} = \alpha_{R_{k+1},R_k}(t)e^{j(w_c(t-(2m+1-k)\cdot TS)+\phi_{B,R_k})}$ from R_{k+1}, it picks a random value $\phi_{k,2} \in [0, 2\pi)$ and sends $s_{k,2}(t) = e^{j(w_c(t-(2m+2-k)\cdot TS)+\phi_{k,2})}$ to R_{k-1}.

(c) R_k computes the estimates of ϕ_{A,R_k} and ϕ_{B,R_k}. Denote these two estimates by $\widehat{\phi}_{A,R_k}$ and $\widehat{\phi}_{B,R_k}$, respectively.

(d) R_k computes $I_{k,B} = \widehat{\phi}_{B,R_k} + \phi_{k,1}$ and $I_{k,A} = \widehat{\phi}_{A,R_k} + \phi_{k,2}$. R_k then quantizes $I_{k,B}$ and $I_{k,A}$ to generate secret bit strings $Q_{k,B}$ and $Q_{k,A}$. Denote the length of $Q_{k,B}$ and $Q_{k,A}$ by q bits.

(2) After z rounds, R_k gets $ST_{k,B} = Q_{k,B,1} \parallel Q_{k,B,2} \parallel \cdots \parallel Q_{k,B,z}$ and $ST_{k,A} = Q_{k,A,1} \parallel Q_{k,A,2} \parallel \cdots \parallel Q_{k,A,z}$, in which $\parallel$ denotes the string concatenation operation. Both $ST_{k,B}$ and $ST_{k,A}$ have a length of $(z \cdot q)$ bits. R_k uses $ST_{k,B}$ to agree on a secret key $\mathrm{KEY}_{k,B}$ with B, and uses $ST_{k,A}$ to agree on a secret key $\mathrm{KEY}_{k,A}$ with A.

From the attack process we can see that $I_{k,B} = \widehat{\phi}_{B,R_k} + \phi_{k,1} = \mathsf{est}(\phi_2 + \psi_{B,R_m} + \sum_{i=k}^{m-1} \psi_{R_{i+1},R_i}) + \phi_{k,1}$, and $I_B = \mathsf{est}(\phi_{k,1} + \sum_{i=k}^{m-1} \psi_{R_i,R_{i+1}} + \psi_{R_m,B}) + \phi_2$. Both $I_{k,B}$ and I_B can be viewed as estimates of $\phi_{k,1} + \sum_{i=k}^{m-1} \psi_{R_i,R_{i+1}} + \psi_{R_m,B} + \phi_2$. By using follow-up quantization, information reconciliation and privacy amplification techniques, R_k and B can agree on a secret key $\mathrm{KEY}_{k,B}$. Similarly, both $I_{k,A}$ and I_A can be viewed as estimates of $\phi_{k,2} + \sum_{i=1}^{k-1} \psi_{R_i,R_{i+1}} + \psi_{A,R_1} + \phi_1$. So R_k and A can also agree on a secret key $\mathrm{KEY}_{k,A}$. In this way, R_k carries out the MITM attack successfully.

4.6. Possible Reduction of Estimation Errors. Given the basic scheme we have designed, there are possible ways to reduce the estimation errors. For instance, the intermediate nodes between A and B may append fix phase delay on forward and backward paths; that is, let $\Psi_{R_i,R_{i+1}} = \Psi_{R_{i+1},R_i}$. This would not reduce secrecy because ϕ_1 and ϕ_2 are random and unknown to the intermediate nodes.

5. The Improved Multihop Key Agreement

Because the basic scheme suffers from threats of internal adversaries and the MITM attack, in this section, we propose an improved multihop key agreement scheme.

5.1. Scheme Outline. In the improved multihop key agreement scheme, we assume that the network is biconnected. Therefore, between any pair of nodes, we can find at least two disjoint paths. The basic scheme suffers from threats from internal adversaries and the MITM attack because the signals are only transmitted in one path. Any node in that path can get knowledge of the extracted common secret bits and can perform the MITM attack. We design the improved multihop key agreement scheme to make it impossible for nodes in one path to get knowledge of the secret key or control it.

We emphasize that the previous goal of security is nontrivial to achieve. In particular, we consider a simple protocol, which we call SMPP hereafter. Assume that there are two disjoint paths Path_A and Path_B between A and B. SMPP starts by letting A and B generate key K_A over Path_A and key K_B over Path_B. Then, A generates two random sequences S_A and S_B, respectively, and sends $K_A \oplus S_A$ over Path_B to B and $K_B \oplus S_B$ over Path_A to B. Finally, B computes S_A by XORing his received value of $K_A \oplus S_A$ with K_A; similarly, he computes S_B. The final key agreed by A and B is the $S_A \parallel S_B$.

Note that SMPP cannot really work against MITM attacks. For example, suppose that there is a node N_{Adv} controlled by the adversary in the middle of Path_A. When A and B try to generate K_A over Path_A, N_{Adv} launches an MITM attack and makes them disagree on the value of K_A. (This is very easy in general, because N_{Adv} can simply play B's role when talking to its neighbor on A's side and play A's role when talking to its neighbor on B's side. In this way, A and N_{Adv} agree on one value of K_A, while N_{Adv} and B agree on another value of K_A.) Hence, A believes that the value of K_A is K_A^A, while B believes that the value of K_A is K_A^B. Both values (K_A^A and K_A^B) are private against nodes in path B. Also suppose that all nodes in Path_B are honest and so A and B agree on the value of K_B, which is private against nodes in Path_A. Next, A generates S_A and S_B and sends $K_A^A \oplus S_A$ over path B and $K_B \oplus S_B$ over path A. Assume that N_{Adv} does not tamper with these transmitted values. Therefore, B receives these values correctly. However, since B has a different belief about the value of K_A, when B tries to recover the value of S_A, he will get $K_A^A \oplus S_A \oplus K_A^B$ instead of S_A. In other words, A and B will disagree on the value of S_A, which is part of the final key.

In order to achieve our goal of security, we use a better approach. We send the initial signals along two disjoint paths between A and B, perform estimation, and forwarding at intermediate nodes and add up the estimated phases of received signals from two paths at the two end nodes. In this way, the sum of phases contain not only the initial random values picked for phases, but also channel phase characteristics of both the two paths. Any adversaries within one single path can neither get the established secret key nor carry out a successful MITM attack.

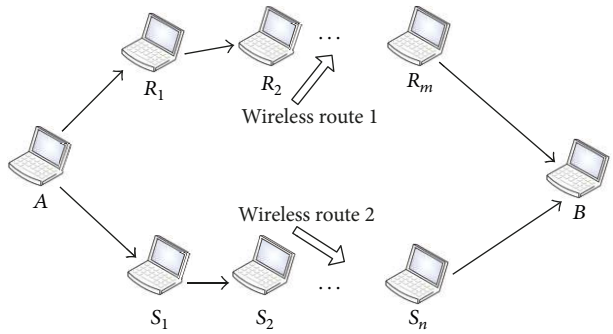

Figure 3: Disjoint routes between A and B.

In the improved multihop key agreement scheme, A and B jointly discover two disjoint paths between them. Denote the lengths of the two paths by m and n, respectively. After that, A and B carry out Round(A, B, m) along the first path and Round(A, B, n) along the second path. They interact with each other for sufficient rounds in order to get the targeted common secret bits. In each round, they add up extracted secrets from both rounds together. Finally, A and B perform quantization, information reconciliation and privacy amplification to get the secret key.

When performing the first step, existing node-disjoint routing discovery protocols [42, 43] can be used. In the improved scheme, we do not assume that there are any preloaded keys or public key infrastructures in the network. Secure routing protocols based on malicious node detection and trust based routing protocols [44–46] can meet this requirement. Using one of these protocols, A can find two disjoint paths to B. After that, A and B perform the rest of the multihop key agreement protocol by using the two paths.

5.2. The Improved Scheme—Detailed Description. Denote the two disjoint paths between A and B by $A \to R_1 \to R_2 \to \cdots \to R_m \to B$ and $A \to S_1 \to S_2 \to \cdots \to S_n \to B$, as shown in Figure 3.

The improved scheme consists of the following steps.

(1) For $i = 1$ to z, A and B perform Round(A, B, m) along the first path and perform Round(A, B, n) along the second path. Without loss of generality, let A (resp., B) use the same initial phase $\phi_{1,i}$ (resp., $\phi_{2,i}$) for Round(A, B, m) and Round(A, B, n). We reset the starting time to 0 after each round. From Round(A, B, m), A and B get $I^{(1)}_{A,i}$ and $I^{(1)}_{B,i}$ as their extracted common secrets; from Round(A, B, n), A and B get $I^{(2)}_{A,i}$ and $I^{(2)}_{B,i}$ as their extracted common secrets. A and B get their final common secrets by computing $I_{A,i} = I^{(1)}_{A,i} + I^{(2)}_{A,i} \bmod 2\pi$ and $I_{B,i} = I^{(1)}_{B,i} + I^{(2)}_{B,i} \bmod 2\pi$, respectively. Denote their extracted secret vectors by $[I_{A,1}, I_{A,2}, \ldots, I_{A,z}]$ and $[I_{B,1}, I_{B,2}, \ldots, I_{B,z}]$, respectively.

(2) A quantizes each value in the vector $[I_{A,1}, I_{A,2}, \ldots, I_{A,z}]$, and B quantizes each value in the vector $[I_{B,1}, I_{B,2}, \ldots, I_{B,z}]$. Denote their generated bit strings by BS_A and BS_B, respectively.

(3) A and B perform information reconciliation and privacy amplification on BS_A and BS_B. After these two processes, they get the secret key.

5.3. Security Analysis. In this section, we give a security analysis of the improved scheme. This security analysis is based on the assumption that all participating nodes are more than half a wavelength away from each other. Just as mentioned in Section 4.5.1, this is a reasonable assumption.

The security of the improved scheme is guaranteed against adversaries in a single path. Collusion attack from adversaries of both paths is not considered. In the following we first prove that the improved scheme is secure against any internal eavesdroppers in a single path. After that we prove that the improved scheme is secure against any MITM adversaries in a single path. (Recall that internal eavesdroppers and MITM adversary are defined at the end of Section 3.)

Theorem 4. *Under the assumption that all nodes are more than half a wavelength away from each other, the improved multihop key agreement scheme is secure against any internal eavesdroppers in a single path.*

Proof. In this proof we enumerate all the phase information that the routing nodes can extract and then point out that they cannot generate any useful information about A and B's secrets.

In the following we consider the collected phase information at an intermediate node in one round. Because the extracted common secrets at each round are quantized separately, they cannot be used for getting knowledge of secrets of other rounds. Consider R_1 in the first path $A \to R_1 \to R_2 \to \cdots \to R_m \to B$. R_1 receives signals from both A and R_2. From the signals received from A and R_2, R_1 gets $\widehat{\phi}_{A,R_1} = \phi_1 + \widehat{\psi}_{A,R_1}$ and $\widehat{\phi}_{B,R_1} = \mathsf{est}(\phi_2 + \sum_{k=1}^{m-1} \psi_{R_{k+1},R_k} + \psi_{B,R_m})$, respectively. From these two phase estimates, R_1 can only get the value of $\widehat{\phi}_{A,R_1} + \widehat{\phi}_{B,R_1}$. However, the secrets obtained by A and B also include the phase estimates through the other path $A \to S_1 \to S_2 \to \cdots \to S_n \to B$. So we can see that R_1 can get no information about the secrets.

For each intermediate node R_k in the first path, we enumerate its estimated phases as follows:

$$\widehat{\phi}_{A,R_k} = \mathsf{est}\left(\phi_1 + \psi_{A,R_1} + \sum_{i=1}^{k-1} \psi_{R_i,R_{i+1}}\right),$$
$$\widehat{\phi}_{B,R_k} = \mathsf{est}\left(\phi_2 + \sum_{i=k}^{m-1} \psi_{R_{i+1},R_i} + \psi_{B,R_m}\right). \quad (5)$$

Because all the m intermediate nodes are more than half a wavelength away from other nodes, they cannot get the phase information from the other path; that is, $\psi_{A,S_1} + \sum_{k=1}^{n-1} \psi_{S_k,S_{k+1}} + \psi_{S_n,B}$. No matter how many nodes in the first path combine their phase information, they cannot gain any knowledge about this value.

Therefore, we can see that the proposed protocol is secure against any internal eavesdroppers in one single path. □

Remark 5. If an eavesdropper is not an intermediate node in either path, and he is more than half a wavelength away from all participating nodes, then he cannot gain any knowledge on the secret key either. This is similar to our analysis in Section 4.5.

Theorem 6. *The improved multihop key agreement scheme is secure against any MITM adversaries in a single path.*

Proof. Without loss of generality, suppose that R_i try to perform the MITM attack to A and B. The purpose of MITM attack is to establish two different keys with A and B, respectively, and after that to relay encrypted messages between them.

In Round(A, B, m), in T_i, R_i receives the signal $r_{A,R_i} = \alpha_{R_{i-1},R_i}(t)e^{j(w_c(t-(i-1)\cdot TS)+\widehat{\phi}_{A,R_{i-1}}+\psi_{R_{i-1},R_i})} + n_{R_i}(t)$ from R_{i-1}. If R_i is an honest node, it will perform the phase estimation of the signal received from R_{i-1} and send the signal $e^{j(w_c(t-i\cdot TS)+\widehat{\phi}_{A,R_i})}$ to R_{i+1}. However, R_i wants to perform the MITM attack, so it generates ϕ_1^e and sends a different signal $e^{j(w_c(t-i\cdot TS)+\phi_1^e)}$ to R_{i+1}. If all other nodes in the first path are honest, then the signal received by B should be

$$\alpha_{R_m,B}(t)\, e^{j(w_c(t-m\cdot TS)+\widehat{\phi}^e_{R_i,R_m}+\psi_{R_m,B})} + n_B(t). \tag{6}$$

In (6), $\widehat{\phi}^e_{R_i,R_m} = \mathsf{est}(\phi_1^e + \sum_{k=i}^{m-1}\psi_{R_k,R_{k+1}})$.

On the other hand, when R_i receives $r_{R_{i+1},R_i}(t) = \alpha_{R_{i+1},R_i}(t)e^{j(w_c(t-(2m+2-i)\cdot TS)+\widehat{\phi}_{B,R_{i+1}}+\psi_{R_{i+1},R_i})} + n_{R_i}(t)$ from R_{i+1} in T_{2m+2-i}, R_i generates another phase ϕ_2^e and sends $e^{j(w_c(t-(2m+3-i)\cdot TS)+\phi_2^e)}$ to R_{i-1}. If $R_1, R_2, \ldots$, and R_{i-1} behave honestly, and then the signal A receives should be

$$\alpha_{R_1,A}(t)\, e^{j(w_c(t-(2m+2)\cdot TS)+\widehat{\phi}^e_{R_i,R_1}+\psi_{R_1,A})} + n_B(t). \tag{7}$$

In (9), $\widehat{\phi}^e_{R_i,R_1} = \mathsf{est}(\phi_2^e + \sum_{k=1}^{i-1}\psi_{R_{k+1},R_k})$.

Now B can get his secret bits by quantizing $\mathsf{est}(\phi_1^e + \sum_{k=i}^{m-1}\psi_{R_k,R_{k+1}} + \psi_{R_m,B}) + I_B^{(2)} + \phi_2$. A can get its secret bits by quantizing $\mathsf{est}(\phi_2^e + \sum_{k=1}^{i-1}\psi_{R_{k+1},R_k} + \psi_{R_1,A}) + I_A^{(2)} + \phi_1$. R_i has $\mathsf{est}(\phi_2 + \psi_{B,R_m} + \sum_{k=i}^{m-1}\psi_{R_{k+1},R_k}) + \phi_1^e$ and $\mathsf{est}(\phi_1 + \psi_{A,R_1} + \sum_{k=1}^{i-1}\psi_{R_k,R_{k+1}}) + \phi_2^e$. However, R_i does not know $I_B^{(2)}$ and $I_A^{(2)}$ either, because R_i is more than half a wavelength from the other path.

From the previous analysis we know that R_i cannot agree on two different keys with A and B. Therefore, it cannot carry out MITM attack successfully. This analysis can be directly extended to the case that any number of intermediate nodes in the first path carry out MITM attacks collaboratively. Because their experienced channels are statistically independent of channels of the second path, they cannot gain any information of $I_B^{(2)}$ or $I_A^{(2)}$.

We conclude that the improved protocol is secure against any MITM adversaries in a single path. □

Remark 7. If the adversary can place cheating nodes on two disjoint paths, there are straightforward ways to extend our protocol to achieve security. For example, we can consider using three disjoint paths between A and B. In general, in order to prevent cheating nodes on k disjoint paths, A and B can use $k+1$ disjoint paths between them for key extraction, as long as there exist $k+1$ disjoint paths between them. (If there are cheating nodes on all disjoint paths between A and B, then no solution is possible because these nodes can choose to simply block all communications between A and B.) This will lead to higher complexity of the protocol—so, there is a tradeoff between security and efficiency.

6. Performance Analysis

As the improved protocol has more than just a pair of nodes, the estimation errors at each intermediate node will aggregate. In this section we present performance analysis of the improved protocol. We mainly focus on the agreement probability of A and B's common secrets.

From the protocol description, we know that the ideal values of I_A and I_B are as follows:

$$\begin{aligned}\overline{I_A} &= 2\phi_2 + \psi_{B,R_m} + \sum_{i=2}^{m}\psi_{R_i,R_{i-1}} + \psi_{R_1,A} \\ &\quad + \psi_{B,S_n} + \sum_{i=2}^{n}\psi_{S_i,S_{i-1}} + \psi_{S_1,A} + 2\phi_1, \\ \overline{I_B} &= 2\phi_1 + \psi_{A,R_1} + \sum_{i=1}^{m-1}\psi_{R_i,R_{i+1}} + \psi_{R_m,B} \\ &\quad + \psi_{A,S_1} + \sum_{i=1}^{n-1}\psi_{S_i,S_{i+1}} + \psi_{S_n,B} + 2\phi_2.\end{aligned} \tag{8}$$

From the channel reciprocity and the assumption that one protocol round is performed within the channel coherence time, we can see that $\overline{I_A} = \overline{I_B}$. We denote this value by $\overline{I}$; that is, $\overline{I} = \overline{I_A} = \overline{I_B}$. However, due to the estimation errors of the phase information, there may be discrepancies between I_A and I_B. In the following we analyze the probability of $I_A = I_B$ during one protocol round. We denote this probability by P_r.

When one node transmits signals to another node, they use the same frequency, so that the receiver does not need to do frequency estimation. Without loss of generality, the noises at the receivers are independent Gaussian noises with zero mean and variance σ^2. The receiver samples the received signal and computes the phase estimate. When the sampling rate is high enough, the estimated phase is a Gaussian random variable whose variance is bounded by the Cramér-Rao bound [47].

From [47], when the signal frequency is known, the variance of the phase is bounded by

$$\sigma_{\widehat{\theta}}^2 \geq \frac{\sigma^2}{b_0^2 N}. \tag{9}$$

In (9), b_0 is the amplitude of the received signal. From (9), we can see that the lower bound of the phase variance depends on the signal to noise ratio (SNR) and the sampling rate. When the SNR is higher, the phase variance can achieve a smaller lower bound. When the sampling rate is increased at the receiver, the lower bound can be further decreased. This is in accordance with the intuition that we should get more precise estimation given a higher SNR and sampling rate. In the following we use the Cramér-Rao bound for our analysis.

The estimation error at each node is modeled as a Gaussian noise, with the zero mean and standard deviation relying on the SNR and the sampling rate. Without loss of generality, we assume that the SNR and the sampling rate are all the same at all the participating nodes. From the protocol execution process, we know that the accumulated estimation error at the source or the destination is the sum of all the intermediate estimation errors. We can write I_B as

$$I_B = \overline{I_B} + Z_B. \tag{10}$$

Z_B represents the accumulated estimation error at B. According to the previous analysis, $Z_B \sim N(0, (m+n+2)\sigma_{\hat{\theta}}^2)$. Because $\overline{I_B} = \overline{I}$, $I_B \sim N(\overline{I}, (m+n+2)\sigma_{\hat{\theta}}^2)$. For ease of analysis, let $\sigma_I^2 = (m+n+2)\sigma_{\hat{\theta}}^2$. From the protocol execution process, we know that $I_A \sim N(\overline{I}, \sigma_I^2)$. Because $\overline{I} \in [0, 2\pi)$, from the property of Gaussian distribution, the probability is much higher when I_B and I_A are close to $\overline{I}$.

The probability P_r is a function of $\overline{I}$. It can be computed using the following equation:

$$P_r = \sum_{i=0}^{q-1} P\left[I_A \in \left[\frac{2\pi i}{q}, \frac{2\pi(i+1)}{q}\right), I_B \in \left[\frac{2\pi i}{q}, \frac{2\pi(i+1)}{q}\right)\right]. \tag{11}$$

Because of the independent noise accumulations at A and B, we can get

$$P_r = \sum_{i=0}^{q-1} P\left[I_A \in \left[\frac{2\pi i}{q}, \frac{2\pi(i+1)}{q}\right)\right] \times P\left[I_B \in \left[\frac{2\pi i}{q}, \frac{2\pi(i+1)}{q}\right)\right]. \tag{12}$$

Denote the interval $[2\pi i/q, 2\pi(i+1)/q)$ by Q_i. Let $P^i(A,B) = P[I_A \in Q_i]P[I_B \in Q_i]$. Then from the distribution function of Gaussian distribution, $P^i(A,B) = \int_{I_A \in Q_i} (1/\sqrt{2\pi}\sigma_I) e^{-(I_A-\overline{I})^2/2\sigma_I^2} \int_{I_B \in Q_i} (1/\sqrt{2\pi}\sigma_I) e^{-(I_B-\overline{I})^2/2\sigma_I^2}$. Because I_A and I_B have the same distributions, P_r can be computed by the following expressions:

$$P_r = \sum_{i=0}^{q-1} P^i(A,B), \qquad P^i(A,B) = \left(\int_{I_A \in Q_i} \frac{1}{\sqrt{2\pi}\sigma_I} e^{-(I_A-\overline{I})^2/2\sigma_I^2}\right)^2. \tag{13}$$

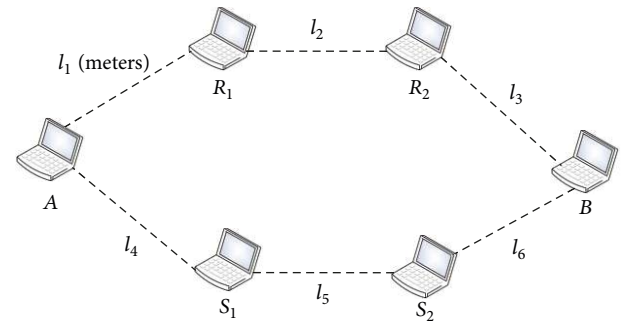

Figure 4: One common node distribution of the simulation.

From (13) we can see that P_r is the sum of the probability that I_A and I_B fall into the same quantization subinterval; that is, $P^i(A,B)$, $i = 1, 2, \ldots, q$. For each subinterval Q_i, the magnitude of $P^i(A,B)$ is affected by whether $\overline{I} \in Q_i$. Suppose that $\overline{I} \in Q_{i^*}$, and then $P^{i^*}(A,B)$ will be larger than any other $P^i(A,B)$ for $\overline{I} \notin Q_i$. This is because the Gaussian distribution function has a larger value when the variable value is closer to the mean (in this case, $\overline{I}$). Therefore, P_r is dominated by $P^{i^*}(A,B)$, for $\overline{I} \in Q_{i^*}$. On the other hand, $P^{i^*}(A,B)$ is affected by $\overline{I}$'s position in Q_{i^*}. If $\overline{I}$ is close to the center of Q_{i^*}, then $P^{i^*}(A,B)$ will be large; if $\overline{I}$ is close to the end points of Q_{i^*}, then $P^{i^*}(A,B)$ will be small. This is because when $\overline{I}$ is close to the end points, the probability that I_A and I_B fall into two adjacent subintervals increases. In addition, the standard deviation σ_I also has impact on $P^{i^*}(A,B)$. A smaller σ_I will result in a larger $P^{i^*}(A,B)$, because when σ_I is smaller, the probability of I_A or I_B being close to $\overline{I}$ is larger.

7. Simulation Results

In order to measure the performance of the proposed scheme, we simulate the proposed scheme using GlomoSim [7]. By using the PARSEC programming language [48], we write programs for the proposed scheme in the physical layer of GlomoSim protocol stack. We simulate the proposed scheme for different SNRs. Because the receiver SNR is affected mainly by distances between adjacent nodes, we select a set of communication distances, which is {10 m, 20 m, 30 m, 40 m, 50 m, 100 m, 150 m, 200 m, 250 m, 300 m}. For each communication distance (denote it by l), we randomly generate a geometric distribution of 6 nodes. The distance between any pair of adjacent nodes is randomly generated in $[0.7l, 1.3l]$. We denote these distances by $\{l_1, l_2, l_3, l_4, l_5, l_6\}$. Because we select 10 communication distances, we also generate 10 random distributions of nodes. One common node distribution for the simulation is shown in Figure 4. We measure average SNRs under different communication distances. The results are shown in Figure 5.

To best simulate the wireless communication environment in reality, we set the center carrier frequency to be 2.437 GHz and the baseband bandwidth to be 11 MHz. This is one of the standard carrier band of 802.11 b. According

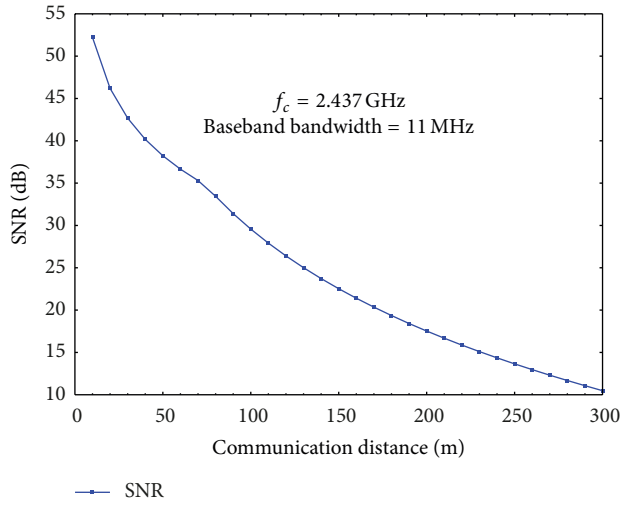

Figure 5: Average SNRs under different communication distances.

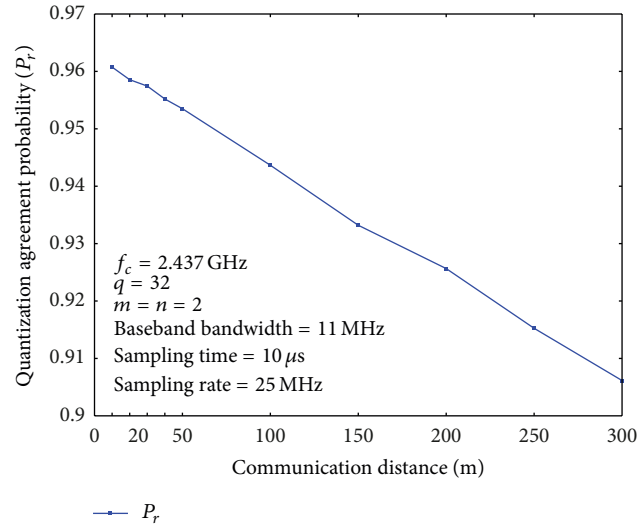

Figure 6: Quantization agreement probabilities under different communication distances.

to Nyquist-Shannon sampling theorem, the sampling rate should be no less than 22 MHz. We choose the sampling rate to be 25 MHz, so that the estimation at the receiver is more accurate. TS is chosen to be 10 μs. For the large scale signal propagation, we use the two-ray ground reflection model [49] which can be expressed by (14)

$$P_r(d) = P_t G_t G_r \frac{h_t^2 h_r^2}{d^4}. \quad (14)$$

In (14), P_t is the transmission power, and $P_r(d)$ is the received power at a distance d away from the transmission antenna. G_t and G_r are the antenna gains at the transmitter and the receiver, respectively; h_t and h_r are the antenna heights at the transmitter and the receiver, respectively; d is the distance between the transmitter and the receiver.

We use the Rayleigh distribution [49] for the small scale wireless fading model. Both the two-ray ground reflection model and the Rayleigh fading model are directly supported by the GlomoSim network simulator [7].

We measure the quantization agreement probability of A and B under different communication distances. We also measure the randomness of the secret key. In addition, we measure the key efficiency of the proposed scheme. The results are shown in Sections 7.1, 7.2, and 7.3.

7.1. Quantization Agreement Probability. Under different communication distances, we measure quantization agreement probabilities and bit error rates (BERs) of the quantized common secret bits. For the quantization step, we choose $q = 32$. Therefore, the interval of $[0, 2\pi)$ is divided into 32 subintervals of equal length. We use the Gray code to encode the quantization indices, so that only one bit discrepancy is introduced for adjacent intervals.

The results are shown in Figures 6 and 7, respectively. From Figures 6 and 7, we can see that when the communication distance is 50 m (approximately 38.23 dB SNR), the quantization agreement probability is 0.9535, and the BER is 0.0093. Even when the communication distance is increased

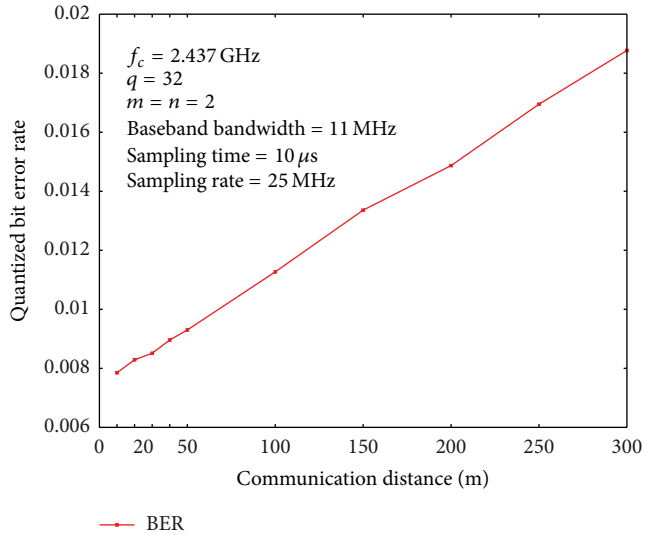

Figure 7: Bit error rates under different communication distances.

to 300 m (approximately 10 dB SNR), the quantization agreement probability is still 0.906, and the BER is 0.019.

7.2. Randomness of the Generated Key. We test the randomness of the generated key using the NIST randomness test suite [50]. We use the 8 tests in the NIST test suite to validate the randomness of one 1024-bit key. The results are shown in Table 1. From Table 1 we can see that the generated key passes all the 8 tests.

7.3. Key Efficiency. In this section, we focus on measuring how long it takes in order to generate a 256-bit key. In order to generate a 256-bit key, A and B need to get more common secret bits, because the Cascade protocol causes entropy loss. We compute the lost entropy rate of Cascade

Table 1: NIST statistical test results. To pass each test, the P value needs to be greater than 0.01.

Test	P value
Frequency	0.70766
Block frequency	0.936991
Runs	0.658522
Longest run of ones	0.871862
FFT	0.066457
Serial	0.815653, 0.586988
Approximate entropy	0.323517
Cumulative sums (forward)	0.745842
Cumulative sums (reverse)	0.745842

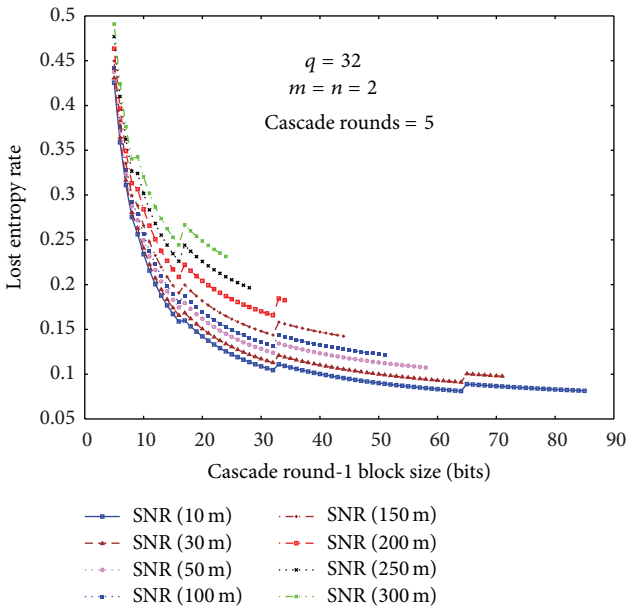

Figure 8: Lost entropy rates under different SNRs and Round-1 cascade blocks.

protocol according to the theoretical results in [40]. After that we measure the key efficiency under different Cascade parameters.

We have completely implemented the Cascade protocol and the privacy amplification method described in Section 4.4. We use the MIRACL library to implement the prime generation and large number arithmetics required for 2-universal hash family. We choose 4~5 rounds for the Cascade protocol, in order that the key agreement ratio is high. We compute the entropy loss rate when the Round-1 block size has different values. For each Round-$(i + 1)$, its block size is two times the block size of Round-i. The results are shown in Figure 8.

As can be seen from Figure 8, when the Round-1 block of Cascade protocol increases, the lost entropy rate decreases. When the communication distance decreases, the lost entropy rate also decreases, because less bits need to be corrected. For example, when the communication distance is 50 m and the round-1 block size is 14, the lost entropy rate is

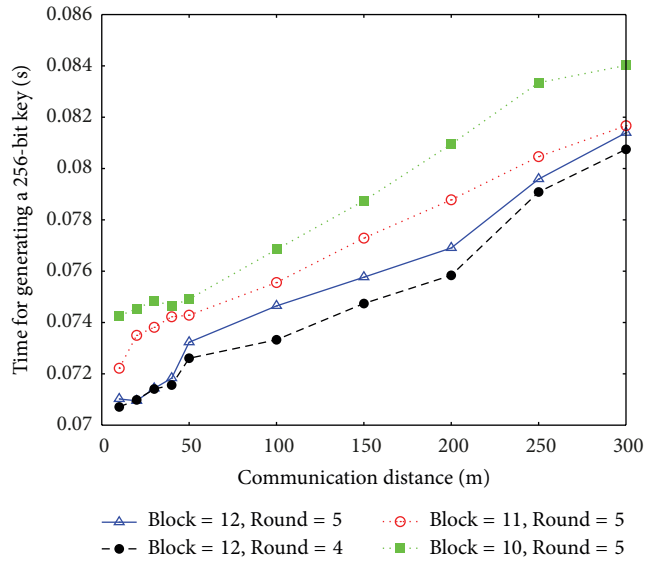

Figure 9: Efficiency of key generation under different communication distances. The measured time is for generating a 256-bit key.

0.1925. Under such a lost entropy rate, in order to generate a 256-bit key, at least 317 common secret bits need to be collected. When the communication distance is 300 m and the round-1 block size is 10, the lost entropy rate is 0.3203. Under such a lost entropy rate, in order to generate a 256-bit key, at least 376 common secret bits need to be collected.

Under the 10 distributions generated for different communication distances, we measure the efficiency of generating a 256-bit key using the multihop key agreement protocol. Different combinations of Cascade rounds and Round-1 block sizes are used. The simulation is run at a laptop with Intel Core2 CPU of 2.33 GHz and 2.0 GB memory. For each different setting, we run the key agreement scheme for 100 times and measure the average time. In all these executions, A and B achieve successful key agreement. The efficiency results are shown in Figure 9.

From Figure 9, we can see that when the Cascade Round-1 block size is decreased, the key efficiency is also decreased. This is because the block number is increased, which increases transferred bits in each round. Furthermore, when the number of Cascade rounds is decreased, the key efficiency is increased. Specifically, for the Cascade parameter (Block = 12, Round = 4), when the communication distance is 50 m, the time of generating a 256-bit key is 0.0726 seconds. At this speed, the proposed key agreement scheme can achieve 3.5 Kbps rate. Even when the communication distance is 300 m, the proposed scheme can still achieve 3.17 Kbps rate.

8. Conclusions and Discussions

In this paper, we propose two key agreement schemes as a novel physical-layer technique in multihop wireless networks. The proposed key agreement schemes enable secret key generation between nodes in multihop wireless

networks, even if they cannot communicate with each other directly. The proposed basic scheme is secure against external eavesdroppers. And the improved two-path-based scheme is secure against external eavesdroppers, as well as internal eavesdroppers and MITM adversaries in a single path. The proposed scheme can achieve high key efficiency under different communication distances among nodes. The secret key generated by the proposed scheme has very strong randomness. By properly selecting the protocol parameters, the proposed scheme can achieve high success ratio. The proposed scheme is suitable for establishing secret keys for multihop wireless networks.

It is worth noting that our paper has covered only key agreement for unicast communications between two nodes. Broadcast and multicast communications may require different protocols for key agreement. In particular, key agreement for broadcast communications in a wireless network is relatively easy if there are only passive eavesdroppers. A straightforward solution is to establish key agreement between neighbor nodes and then transmit a global key in encrypted form throughout the network. If some nodes in the network are dishonest, then leaking the final global key is unavoidable.

For multicast communications, this problem becomes the pretty challenging problem of group key agreement. Existing solutions such as Wang et al.'s [34] are suitable for this case, but further improvement in security and/or efficiency is also possible.

Acknowledgments

This work was partly done while Zhuo Hao and Sheng Zhong were both with University at Buffalo and supported in part by NSF CNS-0845149 and CCF-0915374. Sheng Zhong is currently supported by RPGE and NSFC-61021062.

References

[1] S. Subashini and V. Kavitha, "A survey on security issues in service delivery models of cloud computing," *Journal of Network and Computer Applications*, vol. 34, no. 1, pp. 1–11, 2011.

[2] H. J. Schumacher and S. Ghosh, "A fundamental framework for network security," *Journal of Network and Computer Applications*, vol. 20, no. 3, pp. 305–322, 1997.

[3] S. Mathur, W. Trappe, N. Mandayam, C. Ye, and A. Reznik, "Radio-telepathy: extracting a secret key from an unauthenticated wireless channel," in *Proceedings of the 14th Annual International Conference on Mobile Computing and Networking (MobiCom '08)*, pp. 128–139, ACM, New York, NY, USA, September 2008.

[4] S. Jana, S. N. Premnath, M. Clark, S. K. Kasera, N. Patwari, and S. V. Krishnamurthy, "On the effectiveness of secret key extraction from wireless signal strength in real environments," in *Proceedings of the 15th annual international conference on Mobile computing and networking (MobiCom '09)*, pp. 321–332, ACM, New York, NY, USA, September 2009.

[5] C. Ye, S. Mathur, A. Reznik, Y. Shah, W. Trappe, and N. B. Mandayam, "Information-theoretically secret key generation for fading wireless channels," *IEEE Transactions on Information Forensics and Security*, vol. 5, no. 2, pp. 240–254, 2010.

[6] N. Patwari, J. Croft, S. Jana, and S. K. Kasera, "High-rate uncorrelated bit extraction for shared secret key generation from channel measurements," *IEEE Transactions on Mobile Computing*, vol. 9, no. 1, pp. 17–30, 2010.

[7] X. Zeng, R. Bagrodia, and M. Gerla, "GloMoSim: a library for parallel simulation of large-scale wireless networks," in *Proceedings of the 12th Workshop on Parallel and Distributed Simulation (PADS '98)*, vol. 28, pp. 154–161, July 1998.

[8] J. E. Hershey, A. A. Hassan, and R. Yarlagadda, "Unconventional cryptographic keying variable management," *IEEE Transactions on Communications*, vol. 43, no. 1, pp. 3–6, 1995.

[9] A. A. Hassan, W. E. Stark, J. E. Hershey, and S. Chennakeshu, "Cryptographic key agreement for mobile radio," *Digital Signal Processing*, vol. 6, no. 4, pp. 207–212, 1996.

[10] M. A. Tope and J. C. McEachen, "Unconditionally secure communications over fading channels," in *Proceedings of the Communications for Network-Centric Operations: Creating the Information Force (Milcom '01)*, vol. 1, pp. 54–58, October 2001.

[11] T. Aono, K. Higuchi, T. Ohira, B. Komiyama, and H. Sasaoka, "Wireless secret key generation exploiting reactance-domain scalar response of multipath fading channels," *IEEE Transactions on Antennas and Propagation*, vol. 53, no. 11, pp. 3776–3784, 2005.

[12] L. Zang, X. Wenyuan, R. Miller, and W. Trappe, "Securing wireless systems via lower layer enforcements," in *Proceedings of the 5th ACM Workshop on Wireless Security (WiSE '06)*, pp. 33–42, ACM, September 2006.

[13] C. Ye, A. Reznik, and Y. Shah, "Extracting secrecy from jointly Gaussian random variables," in *Proceedings of the IEEE International Symposium on Information Theory (ISIT '06)*, pp. 2593–2597, July 2006.

[14] B. Azimi-Sadjadi, A. Kiayias, A. Mercado, and B. Yener, "Robust key generation from signal envelopes in wireless networks," in *Proceedings of the 14th ACM Conference on Computer and Communications Security (CCS '07)*, pp. 401–410, November 2007.

[15] C. Ye, A. Reznik, G. Sternberg, and Y. Shah, "On the secrecy capabilities of ITU channels," in *Proceedings of the IEEE 66th Vehicular Technology Conference (VTC '07-Fall)*, pp. 2030–2034, October 2007.

[16] A. Sayeed and A. Perrig, "Secure wireless communications: secret keys through multipath," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '08)*, pp. 3013–3016, April 2008.

[17] S. Nitinawarat, "Secret key generation for correlated Gaussian sources," in *Proceedings of the IEEE International Symposium on Information Theory (ISIT '08)*, pp. 702–706, July 2008.

[18] M. Bloch, J. Barros, M. R. D. Rodrigues, and S. W. McLaughlin, "Wireless information-theoretic security," *IEEE Transactions on Information Theory*, vol. 54, no. 6, pp. 2515–2534, 2008.

[19] X. Tang, R. Liu, P. Spasojević, and H. V. Poor, "Secret-key sharing based on layered broadcast coding over fading channels," in *Proceedings of the IEEE International Symposium on Information Theory (ISIT '09)*, pp. 2762–2766, July 2009.

[20] S. T.-B. Hamida, J.-B. Pierrot, and C. Castelluccia, "An adaptive quantization algorithm for secret key generation using radio channel measurements," in *Proceedings of the 3rd International Conference on New Technologies, Mobility and Security (NTMS '09)*, pp. 1–5, December 2009.

[21] J. Zhang, S. K. Kasera, and N. Patwari, "Mobility assisted secret key generation using wireless link signatures," in *Proceedings*

of the IEEE International Conference on Computer Communications (IEEE INFOCOM '10), pp. 1–5, March 2010.

[22] S. Xiao, W. Gong, and D. Towsley, "Secure wireless communication with dynamic secrets," in *Proceedings of the IEEE International Conference on Computer Communications (IEEE INFOCOM '10)*, pp. 1–9, March 2010.

[23] S. Gollakota and D. Katabi, "Physical layer wireless security made fast and channel independent," in *Proceedings of the 30th IEEE International Conference on Computer Communications (IEEE INFOCOM '11)*, pp. 1125–1133, IEEE, Shanghai, China, April 2011.

[24] M. A. Zafer, D. Agrawal, and M. Srivatsa, "A note on information-theoretic secret key exchange over wireless channels," in *Proceedings of the 47th Annual Allerton Conference on Communication, Control, and Computing (Allerton '09)*, pp. 754–761, October 2009.

[25] J. Wallace, "Secure physical layer key generation schemes: performance and information theoretic limits," in *Proceedings of the IEEE International Conference on Communications (ICC '09)*, pp. 1–5, June 2009.

[26] J. W. Wallace, C. Chen, and M. A. Jensen, "Key generation exploiting MIMO channel evolution: algorithms and theoretical limits," in *Proceedings of the 3rd European Conference on Antennas and Propagation (EuCAP '09)*, pp. 1499–1503, March 2009.

[27] S. C. Draper, A. M. Sayeed, and T.-H. Chou, "Minimum energy per bit for secret key acquisition over multipath wireless channels," in *Proceedings of the IEEE International Symposium on Information Theory (ISIT '09)*, pp. 2296–2300, July 2009.

[28] R. Wilson, D. Tse, and R. A. Scholtz, "Channel identification: secret sharing using reciprocity in ultrawideband channels," in *Proceedings of the IEEE International Conference on Ultra-Wideband (ICUWB '07)*, pp. 270–275, September 2007.

[29] M. G. Madiseh, M. L. McGuire, S. W. Neville, and A. A. B. Shirazi, "Secret key extraction in ultra wideband channels for unsynchronized radios," in *Proceedings of the 6th Annual Communication Networks and Services Research Conference (CNSR '08)*, pp. 88–95, May 2008.

[30] M. G. Madiseh, M. L. McGuire, S. S. Neville, L. Cai, and M. Horie, "Secret key generation and agreement in UWB communication channels," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM '08)*, pp. 1842–1846, December 2008.

[31] S. T.-B. Hamida, J.-B. Pierrot, and C. Castelluccia, "Empirical analysis of UWB channel characteristics for secret key generation in indoor environments," in *Proceedings of the IEEE 21st International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC '10)*, pp. 1984–1989, September 2010.

[32] J. Croft, N. Patwari, and S. K. Kasera, "Robust uncorrelated bit extraction methodologies for wireless sensors," in *Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN '10)*, pp. 70–81, ACM, April 2010.

[33] S. T. Ali, V. Sivaraman, and D. Ostry, "Secret key generation rate vs. reconciliation cost using wireless channel characteristics in body area networks," in *Proceedings of the IEEE/IFIP 8th International Conference on Embedded and Ubiquitous Computing (EUC '10)*, pp. 644–650, December 2010.

[34] Q. Wang, H. Su, K. Ren, and K. Kim, "Fast and scalable secret key generation exploiting channel phase randomness in wireless networks," in *Proceedings of the IEEE International Conference on Computer Communications (IEEE INFOCOM '11)*, pp. 1422–1430, April 2011.

[35] K. Ren, H. Su, and Q. Wang, "Secret key generation exploiting channel characteristics in wireless communications," *IEEE Wireless Communications*, vol. 18, no. 4, pp. 6–12, 2011.

[36] C. H. Bennett, G. Brassard, and J. M. Robert, "Privacy amplification by public discussion," *SIAM Journal on Computing*, vol. 17, no. 2, pp. 210–229, 1988.

[37] R. Impagliazzo, L. A. Levin, and M. Luby, "Pseudo-random generation from one-way functions," in *Proceedings of the 21st Annual ACM Symposium on Theory of Computing (STOC '89)*, pp. 12–24, ACM, May 1989.

[38] C. Cachin, "Linking information reconciliation and privacy amplification," *Journal of Cryptology*, vol. 10, no. 2, pp. 97–110, 1997.

[39] E. N. Gilbert, "Gray codes and paths on the n-cube," *Bell Systems Technical Journal*, vol. 37, pp. 815–826, 1958.

[40] G. Brassard and L. Salvail, "Secret-key reconciliation by public discussion," in *Proceedings of the Workshop on the Theory and Application of of Cryptographic Techniques (EUROCRYPT '93)*, pp. 410–423, Springer, Lofthus, Norway, May 1994.

[41] B. Kanukurthi and L. Reyzin, "Key agreement from close secrets over unsecured channels," in *Proceedings of the 28th Annual International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT '09)*, pp. 206–223, Springer, Berlin, Germany, 2009.

[42] A. Srinivas and E. Modiano, "Minimum energy disjoint path routing in wireless Ad-Hoc networks," in *Proceedings of the 9th Annual International Conference on Mobile Computing and Networking (MobiCom '03)*, pp. 122–133, ACM, September 2003.

[43] J. Tang and G. Xue, "Node-disjoint path routing in wireless networks: Tradeoff between path lifetime and total energy," in *Proceedings of the IEEE International Conference on Communications (ICC '04)*, pp. 3812–3816, June 2004.

[44] S. Marti, T. J. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks," in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MOBICOM '00)*, pp. 255–265, August 2000.

[45] S. Buchegger and J. Y. Le Boudec, "Performance analysis of the CONFIDANT protocol (Cooperation of nodes: fairness in dynamic ad-hoc networks)," in *Proceedings of the 3rd ACM International Symposium on Mobile ad Hoc Networking & Computing (MobiHoc '02)*, pp. 226–236, June 2002.

[46] P. Michiardi and R. Molva, "Core: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks," in *Proceedings of the IFIP TC6/TC11 6th Joint Working Conference on Communications and Multimedia Security*, pp. 107–121, Kluwer BV, Deventer, The Netherlands, 2002.

[47] D. C. Rife and R. R. Boorstyn, "Single tone parameter estimation from discrete-time observations," *IEEE Transactions on Information Theory*, vol. IT-20, no. 5, pp. 591–598, 1974.

[48] R. Bagrodia, R. Meyer, M. Takai et al., "Parsec: a parallel simulation environment for complex systems," *Computer*, vol. 31, no. 10, pp. 77–85, 1998.

[49] T. Rappaport, *Wireless Communications: Principles and Practice*, Prentice Hall PTR, Upper Saddle River, NJ, USA, 2001.

[50] NIST, "A statistical test suite for random and pseudorandom number generators for cryptographic applications," 2001.