Physical-Layer-Enhanced Wireless Secret Key Exchange

Çağatay Çapar
Univ. of Massachusetts Amherst
ccapar@ecs.umass.edu

Murtaza Zafer
IBM Research
mzafer@us.ibm.com

Dennis Goeckel
Univ. of Massachusetts
Amherst
goeckel@ecs.umass.edu

Don Towsley
Univ. of Massachusetts
Amherst
towsley@cs.umass.edu

Dakshi Agrawal
IBM Research
agrawal@us.ibm.com

ABSTRACT

We address the problem of efficiently establishing a shared secret key over an open wireless channel in the presence of an active (jamming) adversary. A commonly employed technique in practice for key sharing is the cryptographic elliptic-curve Diffie-Hellman (DH) protocol; however, its communication cost in a jammed environment is very high. Hence, we employ novel physical-layer techniques to enhance the performance of the DH protocol in a wireless setting. Specifically, we propose a protocol that exploits the randomness inherent to the wireless environment by testing rapidly for time-frequency bands where success can be obtained, essentially probing for those bands that are favorable to the communicating parties and unfavorable to the adversary. The proposed protocol is significantly more energy and time efficient than the standard DH approach, and addresses a number of deficiencies of previous protocols that also attempt to break the circular dependency that arises when bootstrapping secure wireless communications.

Categories and Subject Descriptors

[]

General Terms

Keywords

1. INTRODUCTION

Efficient secure wireless communication between two nodes typically requires them to share a secret key for bootstrapping the channel and enabling higher-layer security mechanisms. For example, spread-spectrum techniques are widely employed for establishing a secure wireless channel, but they require a shared secret spreading code between the legitimate wireless nodes [16, 14]. Establishing pairwise shared keys between the nodes in a network can be achieved by either pre-distribution using trusted offline mechanisms or by on-demand exchange in the field. While the former approach is simple, it is inefficient and incurs a high security management overhead, and is non-scalable for large networks. Thus, for large dynamic wireless networks with changing memberships and a significant overhead of centralized shared key distribution, on-demand key exchange is necessary and a shared secret key needs to be established by the two nodes over the open wireless medium. In this paper, we focus on the specific problem of secret key exchange over a wireless channel.

Secret key exchange is a fundamental problem in security and has been studied in the information-theoretic [9, 1, 2] and the cryptographic literature [10, 4]. The main challenge in the wireless case is the broadcast nature of open wireless transmission, which provides ample opportunity for eavesdropping and signal jamming to the adversary. While many works on wireless security consider a passive eavesdropping adversary [8, 5], we maintain that the consideration of an active adversary (e.g., hostile jammer), rather than a passive eavesdropper, is critical to the problem. Secret key exchange takes place between two nodes who do not share any prior secrets. Therefore, a secure anti-jamming channel has not yet been established because forming this channel itself requires a secret key [14]. Hence, the legitimate nodes have to use the publicly available open wireless channel for the secret key exchange protocol which makes it highly vulnerable to communication disruption. As a result, exchange of message bits necessary for establishing a secret key may incur a high communication cost per bit to overcome jamming attacks. Based on this insight, we propose a secret key exchange method which aims to minimize the number of bits that pay such a high communication cost. In particular, we consider first exchanging a short secret code across the channel at a (potentially) high communication cost, which is then in turn used as a spreading/hopping code to reduce the cost per bit for the longer messages that follow to establish a secure key.

We refer to our proposed method for secret key exchange as the Physical-Layer-Enhanced Key Exchange (PEK) Method. The PEK Method is summarized in Figure 1. Alice and Bob are two nodes with no prior secrets and want to establish a symmetric key in the presence of a potentially jamming adversary Eve by exchanging public messages using a cryptographic method (e.g. Elliptic-curve Diffie-Hellman). Therefore, two long public messages DH_A and DH_B have to be exchanged by Alice and Bob, which will have a high

This research was sponsored by the U.S. Army Research Laboratory and the U.K. Ministry of Defence and was accomplished under Agreement Number W911NF-06-3-0001. The views and conclusions contained in this document are those of the author(s) and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Army Research Laboratory, the U.S. Government, the U.K. Ministry of Defence or the U.K. Government. The U.S. and U.K. Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.
University of Massachusetts Amherst, Computer Science, Technical Report: UM-CS-2010-032.

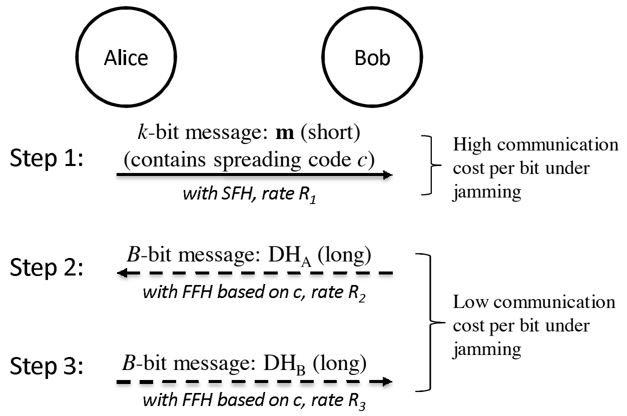

Alice
Bob
Step 1:
k-bit message: $\mathbf{m}$ (short)
(contains spreading code c)
with SFH, rate R_1
High communication cost per bit under jamming
Step 2:
B-bit message: $\mathrm{DH_A}$ (long)
with FFH based on c, rate R_2
Low communication cost per bit under jamming
Step 3:
B-bit message: $\mathrm{DH_B}$ (long)
with FFH based on c, rate R_3

Figure 1: Summary of the Physical-Layer-Enhanced Key Exchange (PEK) Method to do secret key establishment over wireless channels under a possibly jamming adversary. Alice and Bob are nodes with no prior symmetric secrets and they want to establish a secret key by exchanging public messages, DH_A, DH_B, according to Diffie-Hellman key exchange. In Step 1 Alice sends a short message containing a randomly generated spreading code. In Steps 2 and 3, public messages are exchanged using spread spectrum based on this code. Spread spectrum guards the message transmission against jamming as long as the spreading code is known by Alice and Bob only. This method relies on the fact that wireless channels are random and packet losses are common, so it is possible to deliver a short secret code even under a jamming and eavesdropping adversary. Thus, a typical scenario is where Step 1 is repeated until Alice finally receives a valid message from Bob.

communication cost under a jamming adversary without a pre-established efficient channel. In PEK, Alice generates a random spreading code and transmits a short message containing this (ephemeral) spreading code in plain (Step 1 in Figure 1). We consider Fast Frequency Hopping (FFH) as the spread spectrum method employed. After receiving the spreading code, Bob transmits his public message DH_B using this code and thus temporarily guards his communication against possible jamming, and Alice does the same for her public message, DH_A. But it is possible that Eve also receives the spreading code. In that case, Eve can jam the spread spectrum system Bob is employing in Step 2. Therefore, the first stage of PEK has to be such that Alice communicates the short ephemeral code to Bob that Eve is not able to decipher, and this is achieved using the inherent wireless channel randomness. Thus, our method can be viewed as information-theoretic key exchange (the short ephemeral spreading code) assisting computational-security based key exchange through which we establish the long code necessary for enabling a long-term spread spectrum channel.

To accomplish the first step in Figure 1, we take advantage of a key aspect of wireless communications: the time- and frequency-dependent channel quality caused by multipath fading, which makes both the legitimate nodes' and adversary's actions subject to randomness to be exploited. In particular, whenever Alice sends a signal, Bob and Eve will receive the signal through a random channel and during the transmission of a number of packets they will suffer random and independent packet losses. Therefore, it is possible to send a message that is received by Bob and missed by Eve. This is similar to the idea in [17] proposed to address security *maintenance*, where two wireless devices observe their regular link layer retransmissions and regularly update their shared secret key by hashing it with data packets that has been aired only once. This method relies on the fact that packet losses by an adversary is inevitable and in the long run an adversary will miss a packet that has been received by the legitimate node; hence the key will (eventually) be securely updated. In addition to random packet losses suffered by an eavesdropper, a jamming adversary's action will also be subject to randomness due to fading. The jamming signal is sent through a randomly faded channel, and jamming may be unsuccessful when the channel is in deep fade.

In PEK, when Alice sends the message containing the spreading code in Step 1, she waits for an answer on the corresponding spread spectrum band. She may fail to receive a reply either because Bob was not able to decode her message or Bob's reply is jammed by Eve because Eve also received the code. Therefore, Step 1 is repeated until Alice can deliver a message to Bob which Eve fails to decode. We employ a (publicly known) slow frequency-hopped scheme in Step 1 of PEK. Packets will get through when the random fading of the selected frequency band allows signal-to-noise-plus-jamming at the receiver to be high enough, so the transmitter hops until such happens. One way to think about this approach is as probing the frequency spectrum for that band for which the Alice-Bob channel is good, and the Eve-Bob and Alice-Eve channels are poor. The cost of these repeated trials is reasonable because of the short length of the ephemeral spreading code in the Step 1, and, we argue that, the return in terms of improved efficiency for Steps 2 and 3 in Figure 1 will exceed the extra cost brought by Step 1.

When PEK is used for key exchange, a major weakness becomes apparent. In particular, an eavesdropper near Alice will receive packets with a much lower probability of error than Bob, and thus it will take significant time for Bob to receive a packet that Eve does not. Even more concerning is that the position of Eve cannot be assumed to be known, and thus the system is dominated by the concern of a near eavesdropper. We address this weakness by performing *cooperative jamming*, where a second antenna on the transmitter generates noise to reduce the signal-to-noise ratio at Eve. Although this will also degrade the signal for Bob, we show that the overall impact is a significant improvement in the probability that Bob receives a packet that Eve does not.

The circular dependency between forming an anti-jamming channel and availability of a secret key was noted in [14]. To address this circular dependency problem, in [14, 15], a method called Uncoordinated Frequency Hopping (UFH) is proposed where the transmitter and receiver hop randomly on different frequency bands and exchange messages when they share a common hop which is not shared by the adversary. The shared cryptographic messages are then utilized to generate a secret key. Stated differently, [14, 15] introduce artificial randomness to arrive at a channel that has a low-probability occurrence of a "good" channel between Alice and Bob in the face of jamming. In contrast, [6] introduces a scheme where the transmitter employs a successively weaker key over time and the receiver records the output of the channel. At the end of the transmission, the receiver deciphers the weaker keys and then works back to de-spread the longer signal from its recorded signal. The scheme in [14, 15] suffers from a high communication cost and a sophisticated follower-jammer concern, while [6] has concerns of recording wideband signals and jamming of progressively weaker keys which would prevent backward de-spreading. Also, the front-end filter receiving the wideband signal can be easily saturated by an attacker by jamming with full power on a specific frequency band.

PEK differs from the solutions in [14, 15, 6] in the sense that it tries to establish an anti-jamming channel first before doing the cryptographic message exchanges. Secondly, in contrast to the approaches of UFH and [6], which could also be employed on a wideband wired channel, PEK makes use of the unique properties of the wireless channel.

In summary, we address the secret key exchange problem by proposing the Physical-Layer-Enhanced Key Exchange Method which has the following properties: 1) Secret key exchange messages are transmitted over an ephemeral spread spectrum channel, thus reducing the number of bits transmitted over an open channel 2) It uses the natural randomness present in wireless channels thus avoiding the communication cost present in UFH [14, 15] to create randomness 3) It uses physical layer tools like cooperative jamming in order to combat easy eavesdropping advantage brought by the wireless channel. 4) Its communication cost is flexible to jamming intensity enabling a lower cost under less severe jamming or no-jammer case.

Finally, we argue that physical-layer concerns which are naturally orthogonal to the cryptographic key exchange methods can be of significant importance to the real performance over wireless channels. On the other side, a clear understanding of the physical layer can open the way to find solutions applicable specifically to the wireless environment.

The rest of the paper is organized as follows: Section 2 introduces the wireless communication model. Section 3 develops the idea of creating a secret by using packet losses and presents the physical layer tools we propose to improve its performance. Section 4 describes the proposed secret key exchange method under signal jamming and performance analysis is presented under several attacker assumptions. Section 5 is the conclusion.

2. COMMUNICATION MODEL

In this section, we introduce the wireless communication model. We consider a frequency-hopped communication system where the allocated frequency range is divided into smaller frequency bands. Fading experienced by a signal depends on which frequency band or bands the signal occupies, as shown in Figure 6. Here, a slowly fading frequency-selective fading channel [12] is assumed, where bands with sufficient frequency separation experience *independent* fading. For example, in an urban outdoor environment, two frequency bands with more than 100kHz separation can be assumed to be independently faded [3]. In the wideband systems being assumed for future secure communication systems [13], there will be many such independently fading bands. The frequency response of the channel is assumed to be static over a packet, but to vary from one packet to another, which is the standard quasi-static fading model [16].

Now, consider a signal sent with a slow frequency hopped (SFH) system. In an SFH system, the transmitter dwells in a given frequency band for the transmission of a number of bits before hopping to a different band. Given the quasi-static model, the signal is multiplied by a single fading factor during the time the system dwells in a given frequency band. A signal carrying a message consists of a number of physical-layer symbols. Let $x_{A,i}$ be the ith (complex) symbol of an M-symbol message sent by Alice on a given hop of an SFH system. The received symbols at Bob and Eve, $y_{B,i}$ and $y_{E,i}$, respectively, are

$$\begin{aligned} y_{B,i} =& h_{AB}\sqrt{\frac{E_s}{d_{AB}^{\alpha}}}x_{A,i}+n_{B,i}, \\ y_{E,i} =& h_{AE}\sqrt{\frac{E_s}{d_{AE}^{\alpha}}}x_{A,i}+n_{E,i}, \quad i=1,2,\cdots,M. \end{aligned} \tag{1}$$

Here, E_s is the symbol energy, α is the path-loss exponent, d_{XY} is the distance between nodes X and Y. $n_{X,i}$ is the ith complex zero-mean Gaussian noise symbol at node X with $E[|n_{X,i}|^2]=N_0$, $i=1,2,\cdots,M$. h_{XY} is the (complex) fading coefficient between nodes X and Y for that hop. We assume Rayleigh fading with $E[|h_{XY}|^2]=1$, which implies h_{XY} is a complex Gaussian random variable with zero mean and independent components; hence, $|h_{XY}|^2$ is exponentially distributed with mean 1. The fading of channels between different sender-receiver pairs is assumed independent.

A packet is lost if the received signal-to-noise-and-interference ratio (SINR) is below a certain threshold, γ. This assumption is reasonable for modern codes that demonstrate a threshold effect: there is a critical SINR below which the code tends to experience very high codeword error rates and above which there is a sharp decrease to very low error rates [7, pg. 882]. The SINR threshold for successful communication, γ, is determined by the rate at which the symbols are sent. The frequency band allocated and pulse shaping dictate the number of symbols that can be aired per second. Therefore, the rate at which bits are transmitted (bits per second) is proportional to the rate R bits/symbol. Information theoretical results show that the relation between this rate and the corresponding SINR threshold is given by

$$R=\log_2(1+\gamma). \tag{2}$$

A basic trade-off in choosing the communication rate is apparent: a lower rate allows signals to be more easily decoded at the expense of longer message delays and greater energy costs.

Here, we perform some basic analyses on the above model to support succeeding sections. The probability of a correct packet reception at B and E, respectively, is given by:

$$\begin{aligned} P_{\text{rcv}}^{(A\to B)} =& P\left(\frac{|h_{AB}|^2 E_s/d_{AB}^{\alpha}}{N_0}>\gamma\right) \\ =& \exp\left(-\gamma\frac{N_0}{E_s}d_{AB}^{\alpha}\right) \end{aligned} \tag{3}$$

and,

$$P_{\text{rcv}}^{(A\to E)}{=}\exp\left(-\gamma\frac{N_0}{E_s}d_{AE}^{\alpha}\right). \tag{4}$$

With other parameters fixed, the success of decoding depends on the degree of fading of a given sender-receiver channel.

Fast frequency hopping (FFH) [18], where the system hops multiple times during the transmission of a single symbol, is also considered. The symbol is split across K bands, $k = 1, 2, \cdots, K$, each with different fading coefficients, $h_{AB}^{(k)}$; hence the received signal energy at Bob will be

$$\frac{1}{K}\Big(\sum_{k=1}^{K}|h_{AB}^{(k)}|^2\Big)\frac{E_s}{d_{AB}^{\alpha}}. \tag{5}$$

The probability of successful reception with FFH is then

$$P_{\text{rcv}}^{(A\to B)}{=}P\left(\frac{1}{K}\Big(\sum_{k=1}^{K}|h_{AB}^{(k)}|^2\Big)\frac{E_s/N_0}{d_{AB}^{\alpha}}>\gamma\right). \tag{6}$$

For large K, $\frac{1}{K}\sum_{k=1}^{K}|h_{AB}^{(k)}|^2$ approaches its expected value. In fact, the probability of success with K-fold diversity converges rapidly to its limiting value [16]. In other words, it is reasonable to assume that

$$P_{\text{rcv}}^{(A\to B)}=\begin{cases}1, & \text{if } \dfrac{E_s/d_{AB}^{\alpha}}{N_0}\geq\gamma,\\ 0, & \text{if } \dfrac{E_s/d_{AB}^{\alpha}}{N_0}<\gamma\end{cases} \tag{7}$$

for an FFH system.

3. FORMING SECRECY BY PACKET LOSS

Secret information can be created between two wireless nodes by making use of packet losses an eavesdropper suffers. This is the basic idea employed in PEK (Figure 1) to share the information about a secret anti-jamming channel between Alice and Bob. In this section, we analyze the performance of secret generation by packet loss, using the physical layer basics presented in the previous section. We show that creating a secret by making use of packet losses in fact can perform poorly when the adversary has a relative advantage over the intended receiver and propose solutions to overcome it using cooperative techniques. Consider Alice sending a packet to Bob in the presence of Eve, like in Step 1 of PEK. Let S be the event that "the message sent by Alice is received only by Bob", i.e., it qualifies as a *secret message*. Then,

$$P(S)=P_{\text{rcv}}^{A\to B}(1-P_{\text{rcv}}^{A\to E}), \tag{8}$$

The probability terms here are calculated as in (3) and (4). In the following, we evaluate this value under two different attacker scenarios.

3.1 Passive Adversary and Cooperative Jamming

Assume a scenario where Alice and Bob are a fixed distance apart and the location of the passive eavesdropper, Eve, is varied on the line between Alice and Bob. For each location Eve may occupy, we calculate the probability of delivering a message to Bob only, $P(S)$, using (8).

$$P(S)=\exp\left(-\gamma\frac{N_0}{E_s}d_{AB}^{\alpha}\right)\left(1-\exp\left(-\gamma\frac{N_0}{E_s}d_{AE}^{\alpha}\right)\right) \tag{9}$$

The plot is given in Figure 2 (dashed curve). The immediate observation is that delivering a secret packet is most difficult when Eve is close to Alice. In that case, Eve has a large average SINR advantage over Bob due to path-loss, which enables her to receive most of the packets. Hence, creating secrecy by packet loss is inefficient when the attacker is close to the sender while the intended receiver is located further away. We refer to this case as the *near-far problem*. Therefore, although in principle packet losses are inevitable and a secret message will be delivered eventually, Eve's relative advantage can cause inefficiency in terms of energy and delay. Moreover, the attacker's location is typically unknown and performance is dictated by the worst-case scenario.

The solution we propose to the near-far problem is *cooperative jamming*, where a second antenna on the sender side (or a closely located helper node) helps by transmitting noise into the air. In a near-far case, Eve receives a strong signal but also suffers strong noise. On the other hand, Bob is not as affected by this artificial noise as Eve because he is far away. Hence, our solution works by leveling out the difference between Bob and Eve's SINR values. Note that due to several improvements they offer [11], even small handheld devices are more commonly equipped with at least a second antenna. Using a second antenna as a cooperative jammer in a multi-antenna system requires very limited additional complexity.

When Alice sends her message in the presence of cooperative jamming, the packet receive probability at Bob becomes:

$$P_{\text{rcv}}^{(A\to B)}{=}\exp\left(-\gamma\frac{N_0}{E_A}d_{AB}^{\alpha}\right)\frac{1}{K_C}, \tag{10}$$

where

$$K_C\triangleq\left(\gamma\frac{E_C}{E_A}+1\right). \tag{11}$$

Similarly,

$$P_{\text{rcv}}^{(A\to E)}{=}\exp\left(-\gamma\frac{N_0}{E_A}d_{AE}^{\alpha}\right)\frac{1}{K_C}. \tag{12}$$

Here E_C and E_A are the transmit energy of the helper node (or the second antenna) denoted by C, and Alice, respectively. These values are set such that $E_s = E_C + E_A$. Because C and A are very closely located, we assume $d_{AE} = d_{CE}$ and $d_{AB} = d_{CB}$. Notice that, when cooperative jamming is introduced, the receive probabilities are scaled by a factor K_C, which depends on the relative signal powers of the cooperative and adversary nodes. When cooperative jamming is not employed, $K_C = 1$, and the probabilities become equal to the values in (3) and (4).

In Figure 2, we plot the probability that Alice's message is secret when cooperative jamming is employed. Notice that, compared to the plot with no cooperative jamming, the near-far (worst-case) performance is significantly improved at the expense of possibly reduced performance when Eve is located at other places, namely when Eve is closer to the receiver side. Eve's location is typically unknown in a real scenario; hence, worst-case performance is often the primary concern. For a given network, whether cooperative jamming is useful or not can be decided by attacker assumptions. Obviously, if the assumed locations of Eve already excludes points where Eve could get significant advantage in receiving packets, cooperative jamming would be unnecessary. Also, as shown in Figure 2, the relative benefit of cooperative jamming also depends on the the transmit power of nodes which determines the received SINR values.

3.2 Jamming Adversary and Role-Switching

The situation above is changed when the adversary is capable of not only eavesdropping but also disrupting the message transfer at the same time, i.e., when a jamming adversary is present. Note that jamming and listening at the same time, i.e., having full duplex communication is in general hard to achieve because it requires the isolation of the transmit signal from the receiver antenna. In this case, $P_{\rm rcv}^{A\to E}$ is the same while $P_{\rm rcv}^{A\to B}$ will be reduced due to jamming.

When Alice sends a signal in the presence of a jamming adversary, the receive probability of Eve stays the same as in (10); for Bob it becomes:

$$P_{\rm rcv}^{(A\to B)}{=}\exp\left(-\gamma\frac{N_0}{E_A}d_{AB}^{\alpha}\right)\frac{1}{K_C K_E}, \qquad (13)$$

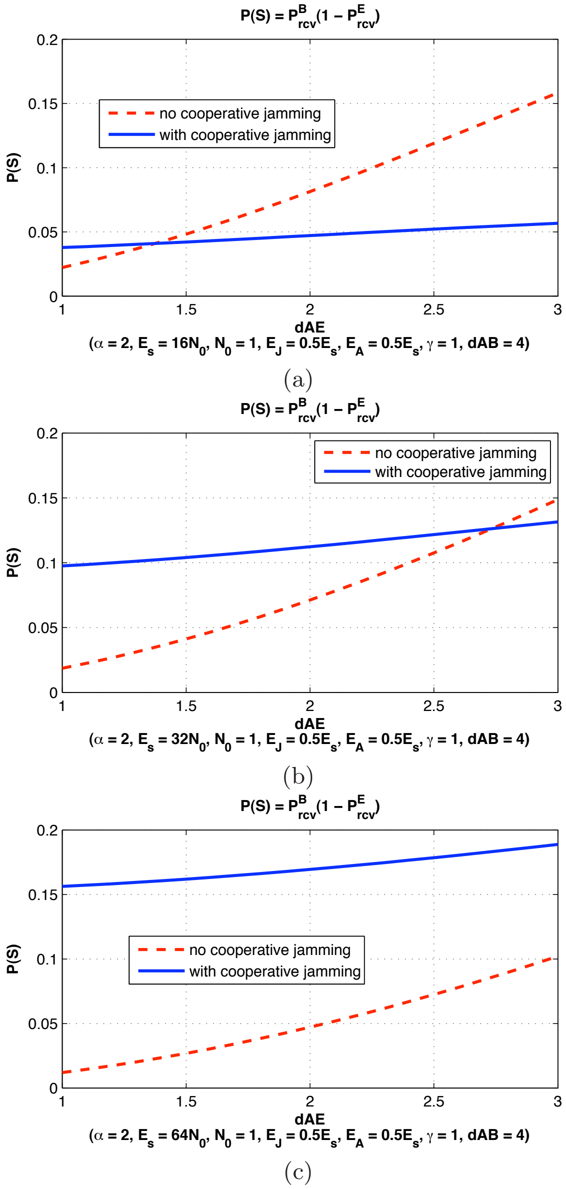
where

$$K_E \triangleq \gamma\frac{\beta/d_{BE}^{\alpha}}{E_A/d_{AB}^{\alpha}}+1.$$

Here β is the attacker's transmit (jamming) power, and K_C is as given in (11).

The effect of a jamming adversary is to scale the receive probability of the intended receiver by a factor K_E, which is a function of the relative locations and transmit powers of the legitimate and adversary nodes. In Figure 3, we plot the probability of a message being secret under a jamming adversary. The immediate observation is that there is an additional weak performance point, namely when the adversary is close to the receiver side. This happens because, in that case, the receiver is inundated with the attacker's noise and not able to receive the sender's message most of the time. A solution to improve performance in the jamming adversary case is to utilize the *asymmetry* in secret message transfer. Namely, instead of one node always being the receiver, the nodes can *switch* sender-receiver roles after a certain number of trials. The effect of role-switching will roughly be to choose the advantaged side as the sender.

For the jamming adversary case, Figure 4 shows the expected number of trials until a secret message is transmitted. When nodes employ neither cooperative jamming nor role-switching, performance becomes worst when Eve is close to either side. Cooperative jamming improves the performance in the near-Eve case but the far-Eve case is still a problem. The best performance is achieved when nodes switch roles after a certain number of trials in addition to using cooperative jamming. Note that, cooperative jammer power and the number of trials after which roles are exchanged can be tuned to find the optimal curve. These plots simply serve to illustrate how these methods can improve secret message



(a)

(b)

(c)

Figure 2: Probability of delivering a secret message from Alice to Bob, $P(S)$, in the presence of a passive eavesdropper Eve. Alice and Bob are d_{AB} units apart. $P(S)$ is plotted against varying attacker location, d_{AE}. $P(S)$ is smallest (i.e., poor performance) when Eve is closest to the sender side. This minimum $P(S)$ value is increased by employing cooperative jamming where a second antenna on Alice generates noise to counteract Eve's location advantage. Note that the total transmit energy is kept the same when cooperative jamming is employed. Performance is plotted for three SINR values. The relative benefit of cooperative jamming depends on the attacker location and the received SINR by both the legitimate node and the eavesdropper.

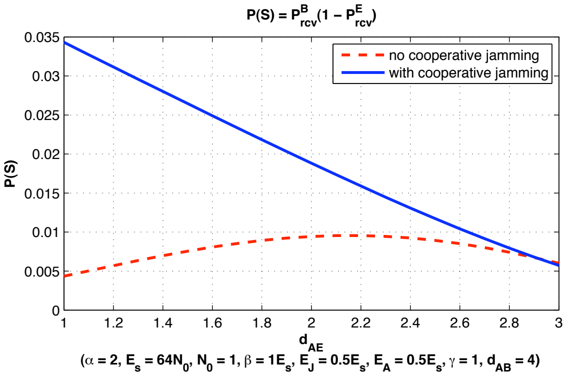


Figure 3: Probability of delivering a secret message from Alice to Bob, $P(S)$, in the presence of a jamming and eavesdropping adversary Eve. $P(S)$ is smallest when Eve is closest to either of the sides. Cooperative jamming improves performance for case the case when Eve is closer to sender side, but cannot improve when Eve is closer to the receiver side, where the receiver is inundated with the attacker's jamming power.

transfer.

Finally, we note that the physical layer analysis and techniques we presented in this section, which we will employ to support our approach to secret key exchange, also apply directly to the key refresh scheme in [17] to improve performance, particularly in the case of a persistent adversary that may follow a node around.

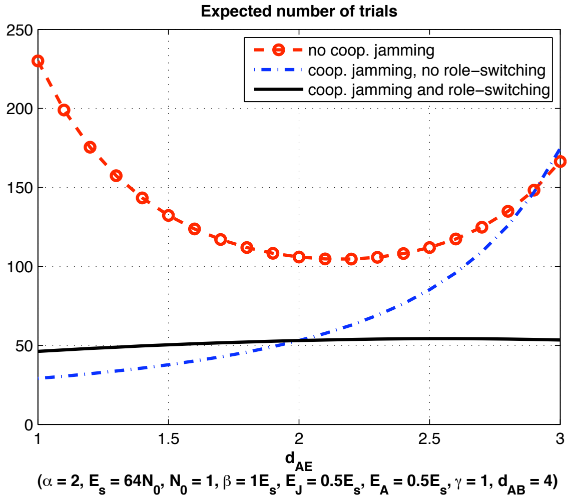


Figure 4: Expected number of message transmissions to deliver a secret message from Alice to Bob under a jamming adversary Eve. Secret message delivery takes the longest when Eve is closest to A or B and cooperative jamming alone cannot improve minimum performance. When Alice and Bob switch sender-receiver roles after a certain number of trials, they avoid the receiver being constantly jammed in the case where attacker is too close to the receiver. The number of trials needed to transfer a secret message is minimized when both cooperative jamming and role-switching are employed.

4. PHYSICAL-LAYER-ENHANCED KEY EXCHANGE METHOD

Background and Motivation

Building upon the physical layer techniques presented in the previous sections, we now consider the main focus of this paper, namely, the problem of secret key exchange over a wireless channel between two nodes (Alice and Bob) in the presence of an adversary (Eve). We consider secret key exchange utilizing the Elliptic Curve Cryptography based Diffie-Hellman (DH) protocol [10] and adapt it to the wireless context.

To begin, we briefly explain the Diffie-Hellman key exchange protocol and other associated assumptions. First, in terms of pre-established security infrastructure, we assume that each node has a public-private key and that a certification authority (CA) provides certificates to bind node identities to their respective keys. While each node holds its own public-private key and the credentials of the CA, it does not have the valid public keys of other network nodes; this is a valid practical assumption especially for large networks with dynamically changing network membership, where it would be practically ineffective for every node to pre-store credentials (or shared secrets) with other network nodes. Also, we assume that no other (secure) channels exist between A and B, including non-wireless, for them to exchange a shared secret during key exchange. Clearly, having such channels would preclude the need for secret key exchange over an open wireless channel, but it will then shift the problem towards enabling and maintaining such other secure channels.

Diffie-Hellman protocol for key exchange requires two message transfers: one from A to B, and the other from B to A, which are denoted as DH_A and DH_B, respectively. Message DH_A typically consists of PK_A (public-key of A), CA's signature of A's credentials, Sig_{CA}, the credential r_A of the DH protocol (A's random number), A's identity and a time-stamp t_A (to avoid a replay attack), and finally A's signature on this message, Sig_A. Assuming a 128-bit security throughout, these components are roughly of the following sizes in bits: $|PK_A| = 256$, $|Sig_{CA}| = 256$, $|r_A| = 256$, $|Sig_A| = 256$, plus bits needed for encoding $A's$ identity and time-stamp t_A (which depends on the network size, protocol implementation, etc. and can be neglected for the scope of this paper). Thus, $|DH_A| = 1024$ bits, and similarly, the message size from B to A would be $|DH_B| = 1024$ bits. Note, the above size is directly related to the security level chosen, and if the security level is increased from 128-bit to 256-bit, each term will get doubled and messages sizes will be 2048 bits.

Using computational security assumptions, the DH protocol enables exchange of a secret key even in the presence of an adversary; however, as we show next, under a jamming adversary a direct DH protocol suffers from significant inefficiency and communication cost. Since typical wireless devices are resource constrained, such an inefficiency could be potentially exploited by an adversary to disrupt the DH protocol in its basic form.

Consider the communication cost of establishing a secret key using the DH method in the presence of a jamming adversary. We know that Alice and Bob will have to exchange public messages DH_A and DH_B, respectively. Suppose that each message is of length B bits and is sent at a rate of R bits/symbol; so, each message requires the transmission of B/R physical-layer symbols. Let us assume that the adversary forces signal jamming with power β which results in a certain success probability for message transmission. Under the basic protocol, Alice keeps sending her public message DH_A until it is successfully received by Bob, at which point Bob repeats the process and transmits his public message DH_B. Assume these messages are sent with a Slow Frequency-Hopped (SFH) system using a publicly-known hopping pattern. The total number of symbols that needs to be transmitted is a random variable, N, with expected value:

$$E(N){=}E(N_1) + E(N_2),$$

where

$$E(N_1){=}\sum_{m=0}^{\infty}(B/R)(m+1)(1-p_B)^m p_B,$$
$$E(N_2){=}\sum_{n=0}^{\infty}(B/R)(n+1)(1-p_A)^n p_A. \qquad (14)$$

Here, m, n denote the number of failed attempts in delivering DH_A, DH_B, respectively. N_1, N_2 are the number of symbols transmitted by Alice and Bob, respectively, and $p_B = P_{rcv}^{(A\to B)}$ and $p_A = P_{rcv}^{(B\to A)}$ are as given in (13) with $K_C = 1$.

We calculate the value of $E(N)$ for an illustrative scenario where Alice and Bob are a fixed distance apart (taken as 10 units) and Eve is located on the line between them; clearly, jamming is most severe when the attacker is close to Alice or Bob. To establish a 128-bit secret key, Alice and Bob need to exchange $B = 1024$ bits in each direction. For a rate $R = 0.2$ bits/symbol, this requires $B/R \approx$ 5k symbols for each transmission of a public message. Figure 5 shows how poorly key exchange performs under severe jamming by dramatically increasing the expected number of symbols transmitted, hence wasting energy and time. In the worst case, the secret key exchange requires more than 100k symbol transmissions on average. Finally, since an attacker location is generally unknown, it makes the worst-case performance the primary concern.

The analysis and results above suggest that secret key exchange under a jamming adversary can incur a very high communication cost in terms of the number of physical-layer symbols that need to be transmitted. In an ad hoc network scenario, where nodes need to conserve energy, or when secret key exchange needs to be done in a small amount of time, this cost can be intolerable; therefore, in the following, we propose a method to address these issues.

Physical-Layer-Enhanced Key Exchange Method (PEK)

It is intuitive that the main reason behind the high communication cost of a direct DH protocol is that Alice and Bob lack an efficient channel to guard their communication against jamming, and as we discussed earlier establishing such a channel using spread spectrum requires a secret key. From Section 3, we see that channel fading in conjunction with cooperative jamming provides a mechanism for generating an information-theoretic secret key, but an important realization is that the packet loss technique of Section 3 is efficient only for *short* key exchanges (i.e. short packet transmissions where the process can be repeated to exploit the frequency-time fading characteristics). This brings us to the main idea behind our proposed Physical-Layer-Enhanced Key Exchange Method (PEK) – namely, *first establish an ephemeral channel using a short key exchange based on a physical layer technique, and then exchange the long DH messages over the ephemeral channel to finally establish the long secret key.*

The PEK method is summarized in Figure 1, where, in Step 1, Alice randomly generates a spreading code for use in Steps 2 and 3 and sends this code over a frequency hopped system with a publicly known hopping pattern. Upon decoding Alice's message from Step 1, Bob extracts the ephemeral spreading code from it and sends his public message DH_B on a frequency hopped pattern based on this code, and finally Alice does the same for her public message DH_A; thus

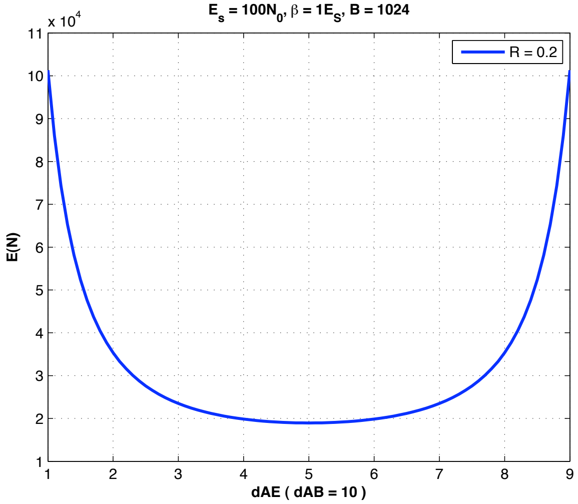
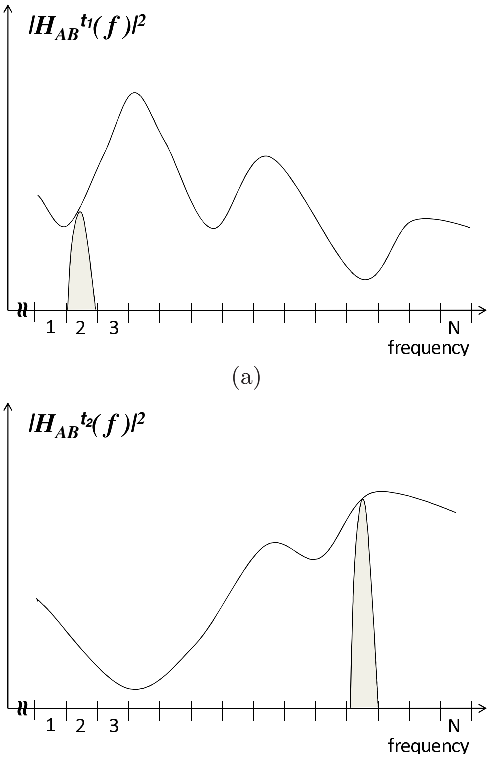


Figure 5: Expected number of symbols, $E(N)$, required to be transmitted to perform secret key exchange without an efficient anti-jamming channel. Under a jamming adversary, Alice and Bob exchange public messages, DH_A, DH_B, respectively, according to Diffie-Hellman protocol. Both messages are sent with a publicly known SFH system. Alice and Bob try to send their respective DH-messages repeatedly until random fading allows a good channel under jamming. As the location of Eve becomes closer to one of the sides, jamming becomes more severe and the communication cost of secret key exchange increases significantly. Each DH-message is $B = 1024$ bits long and $R = 0.2$ bits/symbol is selected to minimize $E(N)$. Each trial of sending a DH-message costs $B/R \approx 5k$ symbols. In the worst case, secret key exchange requires more than 100k symbols on the average.



(a)

(b)

Figure 6: Frequency response of the channel between Alice and Bob as a function of frequency at two time instants t_1 and t_2. The frequency range allocated to the communication system consists of N frequency bands. Each message is transmitted on a single band in a slowly frequency hopped system. In Step 1 of PEK, Alice searches for a good channel by probing different frequency bands. In above, Alice's signal passes through a stronger channel at t_2 compared to t_1.

the DH messages are carried on an efficient anti-jamming channel. This the basic principle behind PEK method and we now delve into details quantifying performance results under different adversarial assumptions.

Clearly, the important requirement of this method is that the short spreading code sent in Step 1 is known to Alice and Bob only. Otherwise, Eve can easily jam frequency hopped transmission and prevent public messages from being transferred. By sending short spreading codes which are hopped in frequency and time, Alice is essentially probing a time and frequency slot where the adversary has a poor channel to the legitimate pair (see Section 2) and Alice-Bob channel is relatively better. Figure 6 provides a schematic sketch of the time varying frequency dependent fading and the corresponding bands picked up. Thus, a typical process will have Alice probe the bands until a *secret* spreading code is conveyed to Bob. After that, the pair exchange their DH messages at a low cost using Fast Frequency Hopping.

Figure 7 summarizes how PEK works in terms of Alice's actions. After sending a randomly generated short spreading code in Step 1, Alice starts listening on the corresponding band. If Bob was not able to decode the message containing the spreading code, he is not able to reply and Alice sends another spreading code. This repeats until Bob is able to decode Alice's message to obtain the spreading code and replies with his DH-message, DH_B. If Alice can decode DH_B, she replies with her DH-message, DH_A. On the other hand, if Eve also receives the spreading code in Step 1, she is forced to jam the band to prevent Alice from decoding Bob's possible reply, in which case the steps repeat again. The tradeoff then is in the exchange of a large number of short messages (short code) and fewer long DH messages.

The rates Alice and Bob choose for sending their messages determine the number of physical-layer symbols required per message and the probability that these messages are successfully decoded. In Step 1, Alice picks rate R_1 bits/symbol. Therefore, the message sent by Alice in Step 1 requires transmitting k/R_1 symbols. Alice sends this message with an SFH system and, hence, the probability that a node can receive this message is given in (3) and (4). In Step 2, (and in the case where he decodes Alice's message) Bob picks rate R_2 bits/symbol. Sending a B-bit DH_B message, therefore, takes B/R_2 symbols. Since these symbols are sent on an FFH system, Bob picks R_2 such that the received SINR at Alice is enough to decode his message as long as the FFH transmission is not jammed (see (7)). Step 3 is very similar to Step 2 where Alice picks the rate R_3 for her FFH transmission of DH_A. Usually the SINR assumptions are symmetric for Bob and Alice and thus $R_3 = R_2$.

The relative benefit of the PEK method depends on how easy it is to send a short secret message from Alice to Bob that enables the formation of an ephemeral secret FFH channel. PEK will improve key exchange efficiency if the cost of establishing this channel is small compared to the cost of DH message exchanges *without* such an efficient channel, like the case plotted in Figure 5. In the following, we analyze the performance of PEK under different scenarios moving from a simple to a more sophisticated attacker model.

4.1 Passive Adversary Case

When the adversary is *known* to be passive, there is no disruption of message transmissions and the only threat is of eavesdropping. Such a scenario is identical to the classical wired eavesdropping case for which the standard Diffie-Hellman protocol suffices and it is *not necessary* to employ any anti-jamming advantage. The PEK method in this case incurs an additional cost in Step 1, which may not be required; however, an important point to note is that this additional exchange of bits is very small because delivering the spreading code in Step 1 will take a small number of trials under a passive adversary. Thus, while PEK is designed for active adversarial cases, its performance is minimally affected in the passive adversary case.

It is important to distinguish between the case where it is known that the adversary is passive where the standard DH approach is sufficient, and the case where there is potentially a jamming adversary, but there is no active jamming during a certain key exchange session. In a real network scenario, an attacker may not always be actively jamming or may not even be present at a given time; thus, security protocols will need to be cognizant of this. While being efficient for the passive adversary case, PEK is flexible to different jamming scenarios. In particular, the probing of bands in Step 1 also serves as a check on whether there is intense jamming or

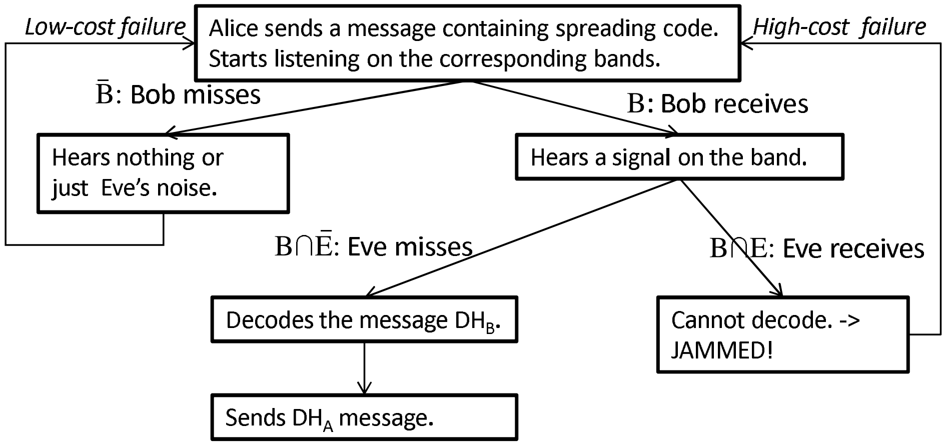

Low-cost failure
Alice sends a message containing spreading code. Starts listening on the corresponding bands.
High-cost failure
B̄: Bob misses
B: Bob receives
Hears nothing or just Eve's noise.
Hears a signal on the band.
B∩Ē: Eve misses
B∩E: Eve receives
Decodes the message DH_B.
Cannot decode. -> JAMMED!
Sends DH_A message.

Figure 7: Flow chart of Alice's actions to perform secret key exchange according to PEK (Figure 1).

not. If there are no jammers around, Steps 1 and 2 will be completed rather easily, hence enabling a lower cost in a low-threat environment.

4.2 Jamming Adversary Case

A jamming adversary is one that not only eavesdrops but also can disrupt communication by transmitting artificial noise. Unlike the legitimate nodes, we assume that a jamming adversary can both listen and jam at the same time (i.e., full-duplex) and evaluate PEK in terms of how much transmit energy is spent by the legitimate nodes. We employ cooperative jamming during the transmission of the message in Step 1 (see Section 3.1) where energy is equally divided for transmission of Alice's message and cooperative jamming. Note that an attacker may also employ a second antenna; however, the additional gain in PEK would be limited. In Step 1 of PEK, Eve can use multi-antenna for additional advantage in receiving the spreading code from Alice and to project away from cooperative jammer's noise, i.e., separate the signal from artificial noise. However, as Alice probes different bands for a good channel, she is essentially also changing the projection of her message onto Eve's antennas, hence making it difficult for Eve to suppress the noise she suffers.

As can be seen in Figure 7, when Alice sends the k-bit message (where k is small, chosen as 32 bits) containing the short code and starts listening, one of four events can happen:

1. $\bar{B} \cap \bar{E}$: *Failure (low-cost)*: Both Bob and Eve are not able to decode Alice's message in Step 1 and hence both miss the spreading code. Alice hears nothing in Step 2 on the FFH system based on the spreading code. The cost is k/R_1 symbols transmitted by Alice.

2. $\bar{B} \cap E$: *Failure (low-cost)*: Only Eve is able to decode Alice's message and hence receive the spreading code. In Step 2, Alice hears potential noise on the FFH system due to Eve's jamming. The cost is k/R_1 symbols symbols transmitted by Alice.

3. $B \cap E$: *Failure (high-cost)*: Both Bob and Eve decode Alice's message and receive the spreading code. In Step 2, Bob replies with B-bit DH message, DH_B. In worst case, Alice cannot decode this message due to jamming. The cost is $(k/R_1 + B/R_2)$ symbols sent by Alice and Bob.

4. $S = B \cap \bar{E}$: *Success*: Only Bob decodes Alice's message and receives the spreading code. In Step 2, Bob replies with B-bit DH message. Alice decodes this message. The cost is $(k/R_1 + B/R_2)$ symbols sent by Alice and Bob. Alice proceeds to Step 3.

Each of these four events has a probability obtained in a manner similar to (13); e.g. $P(\bar{B} \cap E) = (1 - P_{\text{rcv}}^{(A\rightarrow B)}) P_{\text{rcv}}^{(A\rightarrow E)}$.

Alice repeats the above steps until the event $B \cap \bar{E}$ occurs. Whenever Bob misses the spreading code, this results in a failure but with a low cost since k is a small number. However, when both Bob and Eve receive the code, this results in a failure with a high cost. In a typical scenario, Alice receives Bob's DH-message after a series of failed attempts to either deliver the spreading code to Bob ($\bar{B} \cap E$ or $\bar{B} \cap \bar{E}$) or to hear Bob's response in jamming ($E \cap B$). Figure 8 shows how this process works on the time-frequency plane. Let N_1 be the random variable denoting the number of symbols transmitted starting with Alice's first transmission until she successfully decodes Bob's DH-message. Then,

$$E(N_1) =$$
$$\sum_{m=0} \sum_{n=0} \left((m+1)(\frac{k}{R_1} + \frac{B}{R_2}) + n\frac{k}{R_1} \right) \binom{m+n}{m} p_{B \cap E}^m p_{\bar{B}}^n p_S,$$

where m denotes the number of high-cost failures (i.e., $B \cap E$) and n denotes the number of low-cost failures (i.e., $\bar{B} = (\bar{B} \cap E) \cup (\bar{B} \cap \bar{E})$). p_X denotes the probability of event X.

The last step to complete secret key exchange is Step 3, where upon receiving Bob's DH-message, Alice replies with her B-bit DH_A message with rate R_3 on the same FFH pattern that Bob employed to transmit DH_B to her. As noted above, due to similar SINR assumptions, R_3 will typically be the same as R_2. Per (7), DH_A will be received as long as this assumption is valid. With that assumption, DH_A will be delivered in one attempt and will require $N_2 = B/R_3 = B/R_2$ symbols. Therefore, the expected number of total symbols

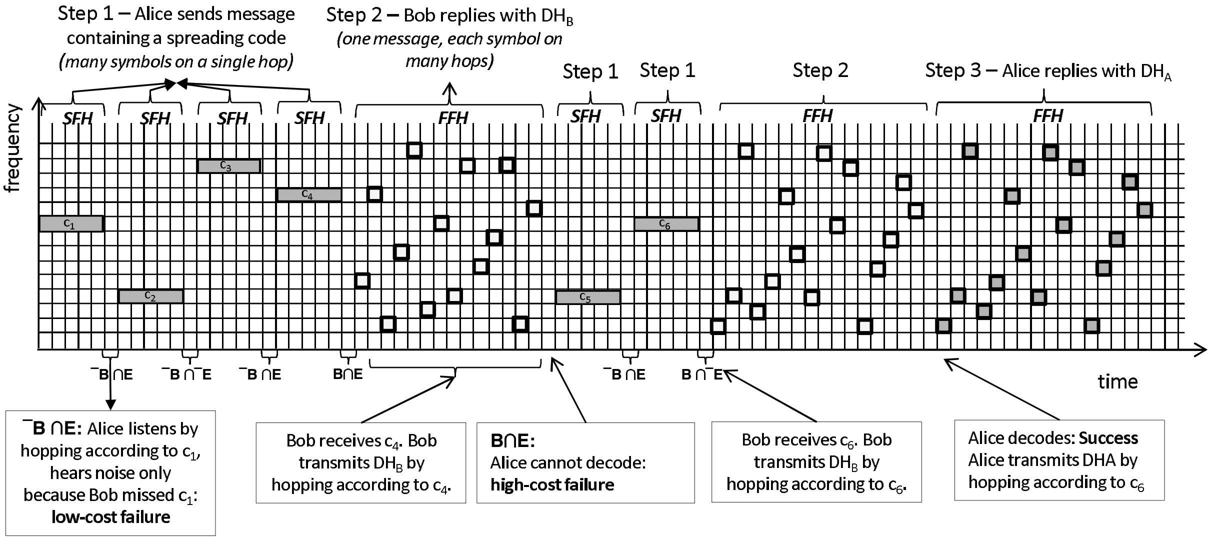

Step 1 – Alice sends message containing a spreading code (many symbols on a single hop)
Step 2 – Bob replies with DH_B (one message, each symbol on many hops)
Step 1
Step 1
Step 2
Step 3 – Alice replies with DH_A
SFH
SFH
SFH
SFH
FFH
SFH
SFH
FFH
FFH
frequency
c_1
c_2
c_3
c_4
c_5
c_6
time
$\bar{B} \cap E$
$\bar{B} \cap \bar{E}$
$\bar{B} \cap E$
$B \cap E$
$\bar{B} \cap E$
$B \cap \bar{E}$
$\bar{B} \cap E$: Alice listens by hopping according to c_1, hears noise only because Bob missed c_1: low-cost failure
Bob receives c_4. Bob transmits DH_B by hopping according to c_4.
$B \cap E$: Alice cannot decode: high-cost failure
Bob receives c_6. Bob transmits DH_B by hopping according to c_6.
Alice decodes: Success Alice transmits DHA by hopping according to c_6

Figure 8: Illustration of how PEK (Figure 1) runs over time and frequency. In Step 1, Alice sends her message with a slow frequency hopped (SFH) system. In SFH, each individual message, which consists of several symbols, are sent over a single band. Messages in Step 2 and 3 are sent with FFH. In FFH, even a single symbol of a message is sent by hopping quickly between several bands. Secret Key Exchange will be successful after a number of failures and will require repeated transmissions of spreading codes c_i by Alice. Before transmitting another spreading code, Alice listens for a small amount of time for a possible reply from Bob. When Alice receives a signal from Bob, she tries to decode this message and if successful replies with her message, DH_A, with FFH.

sent during key establishment by PEK is given by

$$E(N) = E(N_1) + B/R_2. \tag{15}$$

If the expected number of symbols transmitted in PEK (15) is less than the same value in (14), then PEK will spend less transmit energy on the average and thus enhance the efficiency of secret key exchange. In Figure 9, we plot the expected number of symbol transmissions by the legitimate nodes to establish a secret key. In order to reflect the effect of varying jamming and eavesdropping capability of the attacker, we assume Alice and Bob are a fixed distance d_{AB} apart and Eve is located on the line between Alice and Bob in a given range. Worst-case assumption on jamming intensity depends on how large this range is, i.e., how close Eve can get to the either side. Note that Alice and Bob optimize their communication rates depending on this assumption. Plots show that for all the considered ranges Eve might span, PEK requires almost half the transmit energy compared to a direct DH key exchange. Therefore, the extra energy spent in PEK to form a secret ephemeral channel (using the short code) is justified by the reduced cost in exchanging DH-messages in latter steps. Also note that, as discussed in Section 3.2, role-switching can be applied to improve the performance. In particular, after a certain number of failures Alice and Bob can switch their sender-receiver roles in the PEK Method.

4.3 Message Insertion by Attacker

In this section, we consider an attacker which is capable, in addition to eavesdropping and jamming with random noise, of inserting her own messages into the medium. Note that, whenever the attacker transmits her own message, we assume that she will not jam the same channel in parallel, because jamming will disturb the decoding of the fake message and make message insertion irrelevant. We first discuss insertion of fake FFH code and then consider the insertion of fake DH messages.

Insertion of an FFH code

During Step 1 of PEK, when Alice sends the short spreading code, an adversary can insert her own fake spreading code by transmitting at the same time as Alice. Let the codes sent by Alice and Eve be denoted as c_A and c_E, respectively. One of the following four events can happen at Bob's side:

1. $\bar{B}$: Bob cannot decode either of the messages and misses c_A and c_E: This case is identical to a jamming-only attacker where Bob misses Alice's message.

2. B_{AE}: Bob receives both c_A and c_E: This case happens when Bob employs *successive interference cancellation* [16], and extracts two FFH codes, c_A, c_E, but does not know which one is from the legitimate sender. To address this problem, Bob sends his public message DH_B twice, one according to c_A, and then according to c_E. However, the order of these messages matter because if the first message is on c_E, Alice cannot decode but has to keep listening in case there is a message in the second period. This can be addressed by a slight modification of how Bob forms his response when he receives two codes (see Figure 10). Bob divides his message DH_B into two fragments. The first fragment is a very short message containing the first few bits of DH_B. By choosing some arbitrary order, Bob repeats this fragment on the two spreading codes he received.

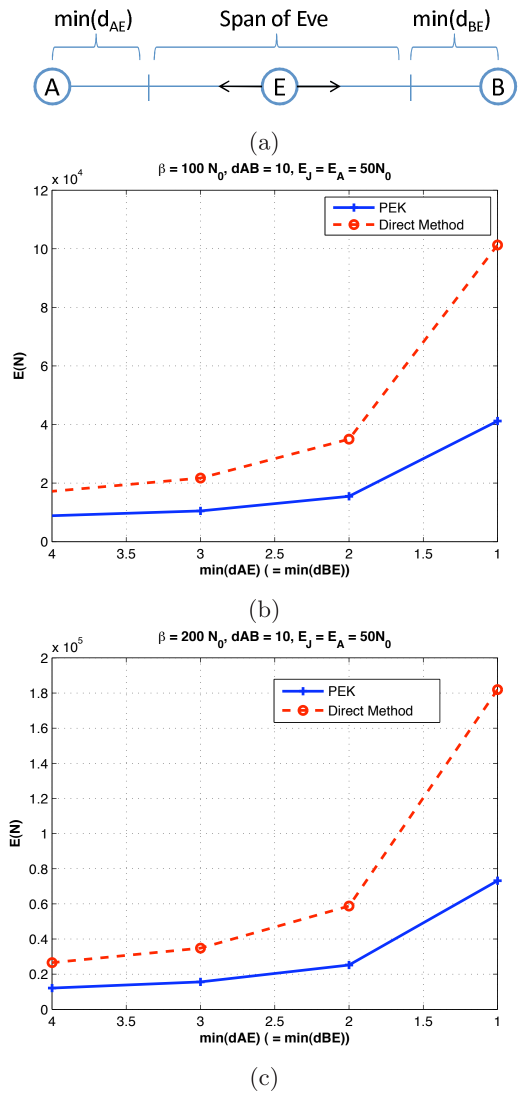


Figure 9: Expected number of legitimate symbols transmitted, $E(N)$, for establishing a secret key. Each point on the horizontal axis corresponds to how much Eve can come close to either side, going from smaller to a larger span of Eve. For each point, the *worst-case* $E(N)$ is given. The transmit energy of Eve, β is equal to signal energy E_s in (a) and $2E_s$ in (b). E_s is shared equally by cooperative jammer and symbol transmission in PEK case. Jamming becomes more severe as the span of Eve gets larger and hence $E(N)$ increases. For all jamming scenarios, PEK requires less number of symbols transmitted and therefore requires less energy compared to the key exchange without an efficient channel.

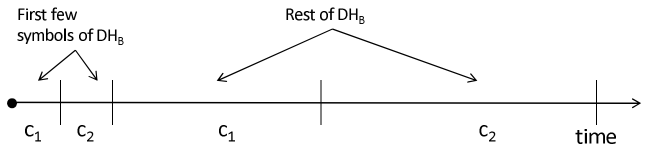


Figure 10: Bob's response in the case where he extracts two FFH codes c_1, c_2. Bob sends his message DH_B on both hopping codes by dividing the message into two fragments. The first fragment is a very short sequence consisting of a few symbols carrying the first few bits of DH_B. This fragment is repeated on c_1 and c_2 in some arbitrary order. The same is done for the second fragment with the same order.

The second fragment, which is a much longer message is repeated with the same order. If only one code is received by Bob, he responds as usual. If Alice cannot decode any signal for the first two short durations, she will stop listening and turn back to Step 1.

3. B_E: Bob receives only c_E. In this case, Bob replies with DH_B on Eve's channel and this incurs a high communication cost.

4. B_A: Bob receives only c_A. This is the same case as the event B in Section 4.2. Bob replies with DH_B and depending on Alice being able to decode this message or not, nodes will either proceed to Step 3 to finish key exchange or will go back to Step 1.

Insertion of a DH message

A second scenario is where the attacker sends her own Diffie-Hellman message in Step 2 of PEK. This can happen in the case where Eve receives the FFH code, c_A, sent by Alice in Step 1. We assume that during key exchange Alice and Bob know each other's identity through public announcement of their device IDs; thus, while Eve can fake identities and send fake DH messages, these messages will fail authentication at Alice due to the presence of certificates from the central authority (CA).

Consider the case where Bob has missed Alice's message and thus does not have c_A. If Eve chooses to send DH_E, Alice will hear Eve's signal on the band and decode DH_E. However, this message will fail authentication. So, Alice will return to Step 1. The additional cost brought by DH-message insertion by the attacker largely depends on how the performance is defined. If the performance metric is the transmit energy spent by the legitimate nodes, this event does not incur extra cost because Alice will not reply an unauthenticated DH-message. In other words, from an energy perspective the receipt of DH_E by Alice qualifies as a low-cost failure for PEK.

If the performance metric is time, then Eve faking a DH-message raises a legitimate concern for the efficiency. Basically, Alice wastes time listening to DH_E – which is a long message – and performing the authentication check. This converts the low-cost failure event $\bar{B} \cap E$ into a high-cost failure event. In the case where Bob also received c_A and sends DH_B, either the messages will fail to get decoded by

Alice, or through successive interference cancellation Alice can decode DH_B along with DH_E. Clearly, only DH_B will pass the authentication check and Alice will proceed to Step 3.

An important consideration regarding the difference between the actions of jamming and message insertion by an adversary is the capability of the legitimate nodes to perform Successive Interference Cancellation (SIC) [16]. By employing SIC, a node may be able to decode messages sent by an adversary and a legitimate node at the same time on the same band. For this reason, it would be practically difficult for an attacker to disturb the otherwise possible decoding of a legitimate message by solely transmitting her own message, even if the received signal power from the attacker is significantly larger. Therefore, if nodes can perform SIC, an adversary essentially risks allowing a node's message to pass through by choosing to send a fake message instead of jamming the band. In that case, Eve's optimal strategy would be to jam rather than transmit her own message. In order to perform SIC, a node needs to record the received signal. In PEK, messages are sent either on SFH or FFH and hence, at any given instant a signal is carried in only one of the frequency bands. Unlike for wideband signals, recording a narrowband signal is an easy requirement.

5. SUMMARY AND CONCLUSIONS

In this paper, we address the problem of secret key exchange over a wireless channel. Secret key exchange requires exchanging long messages and two nodes lack a secure channel to guard their communication against disruption since they do not share any prior symmetric secrets . Therefore, message transfers for key exchange has to be done over a public wireless channel. We argue that an active adversary can exploit this and cause significant cost to key exchange by jamming. We describe a physical-layer communication model and analyze the communication cost of secret key exchange in the presence of a jamming adversary. Then, we propose a method where a short spreading code is first transmitted to form an ephemeral channel to carry the long messages required for key exchange. In order for the spreading code to be secret between nodes, we exploit the time and frequency-dependent randomness in wireless channels which causes inevitable packet losses to a possible attacker. We analytically show that establishing a temporary channel to do key exchange results in increased overall efficiency compared to the case where message exchanges are done without an efficient channel.

We conclude that a clear physical-layer understanding is crucial to better understand unique challenges of the wireless key exchange problem and to find solutions that exploit unique features of the wireless environment.

6. REFERENCES

[1] R. Ahlswede and I. Csiszar. Common randomness in information theory and cryptography. i. secret sharing. *Information Theory, IEEE Transactions on*, 39(4):1121–1132, Jul 1993.

[2] M. Bloch, J. Barros, M. Rodrigues, and S. McLaughlin. Wireless information-theoretic security. *Information Theory, IEEE Transactions on*, 54(6):2515–2534, June 2008.

[3] J. Cavers. *Mobile Channel Characteristics*. Springer, 1 edition, Sept. 2000.

[4] Y. Dodis, J. Katz, and L. Reyzin. Robust fuzzy extractors and authenticated key agreement from close secrets. In *In Advances in CryptologyŮCRYPTO Š06*, pages 232–250. Springer, 2006.

[5] 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 *MobiCom '09: Proceedings of the 15th annual international conference on Mobile computing and networking*, pages 321–332, New York, NY, USA, 2009. ACM.

[6] T. Jin, G. Noubir, and B. Thapa. Zero pre-shared secret key establishment in the presence of jammers. In *MobiHoc '09: Proceedings of the tenth ACM international symposium on Mobile ad hoc networking and computing*, pages 219–228, New York, NY, USA, 2009. ACM.

[7] S. Lin and D. J. Costello. *Error control coding*. Pearson-Prentice Hall, 2 edition, 2004.

[8] S. Mathur, W. Trappe, N. Mandayam, C. Ye, and A. Reznik. Radio-telepathy: extracting a secret key from an unauthenticated wireless channel. In *MobiCom '08: Proceedings of the 14th ACM international conference on Mobile computing and networking*, pages 128–139, New York, NY, USA, 2008. ACM.

[9] U. Maurer. Secret key agreement by public discussion from common information. *Information Theory, IEEE Transactions on*, 39(3):733–742, May 1993.

[10] A. J. Menezes. *Elliptic Curve Public Key Cryptosystems*. Springer, 1st edition, July 1993.

[11] A. Paulraj, D. Gore, R. Nabar, and H. Bolcskei. An overview of mimo communications - a key to gigabit wireless. *Proceedings of the IEEE*, 92(2):198 – 218, feb 2004.

[12] J. Proakis. *Digital Communications*. McGraw-Hill Science/Engineering/Math, 4 edition, Aug. 2000.

[13] D. Stephens, B. Salisbury, and K. Richardson. Jtrs infrastructure architecture and standards. pages 1 –5, oct. 2006.

[14] M. Strasser, S. Capkun, C. Popper, and M. Cagalj. Jamming-resistant key establishment using uncoordinated frequency hopping. In *Security and Privacy, 2008. SP 2008. IEEE Symposium on*, pages 64–78, May 2008.

[15] M. Strasser, C. Pöpper, and S. Čapkun. Efficient uncoordinated fhss anti-jamming communication. In *MobiHoc '09: Proceedings of the tenth ACM international symposium on Mobile ad hoc networking and computing*, pages 207–218, New York, NY, USA, 2009. ACM.

[16] D. Tse and P. Viswanath. *Fundamentals of Wireless Communication*. Cambridge University Press, June 2005.

[17] S. Xiao, W. Gong, and D. Towsley. Secure wireless communication with dynamic secrets. In *INFOCOM 2010, IEEE*, 2010.

[18] R. E. Ziemer, R. L. Peterson, and D. E. Borth. *Introduction to Spread Spectrum Communications*. Prentice Hall, Apr. 1995.