Evaluating Performance of Cryptographic Protocols in the Wireless Environment

Çağatay Çapar, Elizabeth A. Quaglia, Murtaza Zafer, Kenneth Paterson, Dennis Goeckel, and Don Towsley

***Abstract*—We provide a toolbox which allows us to accurately analyze the performance of cryptographic protocols in wireless environments, where protocols are subject to sophisticated adversarial attacks including jamming, message insertion and message deletion. As an application of our toolbox, we consider the fundamental problem of how to securely establish a key over a wireless channel in the presence of an adversary. We are able to efficiently and accurately quantify the efficiency of existing families of key exchange protocol, showing, for example, that an ID-based approach can offer an almost 10-fold improvement in energy consumption when compared to a traditional PKI-based protocol. We then introduce an entirely new class of key exchange protocol designed specifically for the wireless environment. Here, we combine traditional cryptographic methods with physical-layer techniques, including the use of "ephemeral" spreading codes, cooperative jamming, and role-switching. Using the toolbox, we demonstrate that the new approach offers further significant performance advantages over traditional designs.**

I. Introduction

With the wide proliferation of wireless communication, securing information sent over wireless channels is important and has rightfully received significant research attention. However, the design and evaluation of security protocols for wireless communication systems has mainly focused on traditional metrics of security, such as resilience against various attacks, key-size versus computational complexity tradeoffs, etc. No doubt, such metrics are important security considerations, but from a complementary viewpoint it is also equally important to understand the efficiency of a security protocol measured in terms of its communication or other costs.

Inherently, communication over wireless channels is probabilistic in nature due to random errors caused by signal fading, multipath/shadowing and noise, and due to potential adversarial attacks such as signal jamming. Therefore, evaluating the end-to-end performance of a security protocol becomes especially difficult when considering a wireless setting. To give an example, suppose that a security protocol requires the exchange of certain messages over an open wireless channel that is subject to adversarial attacks that can cause multiple re-trials and protocol restarts. These messages and the signals of the adversary will both be subject to random signal fading; thus, the logical flow of the security protocol will be probabilistic in nature. Evaluating the end-to-end performance is thus non-trivial.

Thus, the above observation motivates the question of whether efficient analysis tools exist that can be used to quickly and accurately characterize the performance of cryptographic protocols operating on wireless channels. In this paper, *we focus on the development of such tools and then consider their application to the fundamental problem of how to efficiently establish a shared secret key over an open wireless channel in the presence of an active adversary.*

In particular, motivated by protocols for the key establishment problem, we observe that cryptographic protocols employed over wireless channels can be regarded as a type of dynamic probabilistic system. This allows us to provide a systematic method for their analysis as follows. First, the standard flow diagram for a cryptographic protocol is augmented by adding costs (energy, delay) and probabilities to its branches, where the latter are derived from channel models. Second, the flow diagram is simplified using a standard collection of equivalences which then leads to a "transfer function" for the protocol. Lastly, this transfer function then enables studying virtually any metric of interest, such as the expected cost (e.g. energy, delay) of the protocol, moments of these costs, or the probability that these costs exceed some threshold.

A. Wireless Key Establishment

We apply our analysis techniques to a detailed consideration of the problem of key establishment in the presence of an active adversary. Bootstrapping security over a wireless channel requires first establishing a jam-resilient communication channel, since otherwise open air transmissions are highly susceptible to disruption attacks such as signal jamming. A state-of-the-art approach that is generally employed in this setting is to use spread-spectrum communications techniques, which limits an adversary's ability to jam network nodes' signals without expending large amounts of energy [1]. However, establishing a spread-spectrum channel requires the concerned parties to already share or securely establish a cryptographic key, enabling them to select a 'private' spread spectrum channel which is unknown to the adversary. In turn, this requires any pair of network nodes who might wish to establish jam-resilient wireless communication to either have available

C. Capar, D. Goeckel and D. Towsley are with the University of Massachusetts Amherst, Amherst, MA, 01003 USA. Email: {ccapar,goeckel}@ecs.umass.edu, towsley@cs.umass.edu. M. Zafer is with IBM T.J. Watson Research Center, NY, 10532 USA. Email: mzafer@us.ibm.com. E. A. Quaglia and K. Paterson are with Royal Holloway, University of London, UK. Email: {e.a.quaglia,kenny.paterson}@rhul.ac.uk

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.

a pre-established key, or to run a key establishment protocol over an 'open' channel prior to switching to a spread spectrum channel determined by the agreed key.

Consider first the case of using pre-established keys. If we consider the setting where we have a large numbers of network nodes who may wish to establish secure communications and where node compromise is a realistic threat – for example in an emergency scenario or in a military environment – then having a single, system-wide pre-established key is not a viable solution, since compromise of a single node then compromises the whole network. On the other hand, having a unique pre-shared key per possible pair of communicating parties is not a good solution either, since it does not scale well and is inflexible once deployed. Intermediate solutions, such as those proposed in [2], [3], scale better, but may still require substantial key storage at the nodes. Clearly this provides a highly flexible means of bootstrapping a secure wireless channel between two wireless devices, making it very attractive in many emergency and military scenarios. The problem, however, is that the messages exchanged during key establishment are subject to active adversarial attacks. Because of this (and because of the inherently noisy communications environment), the protocol participants may be forced to repeat steps, or even re-start the protocol from scratch, many times before a session key is successfully established. This implies that establishing a private spread spectrum communications channel may incur significant energy costs, quickly draining battery energy for example. At the outset, it is not clear which protocols minimise energy consumption, or indeed what trade-offs between security and cost might be possible. Quantifying these is essential in selecting the best candidate protocol for a wireless environment. Nor is it clear that current classes of protocol for key exchange, designed mostly with wired networks in mind, are even well-suited to deployment in wireless networks, or whether alternative protocols designed specifically for the wireless environment might perform better.

This fundamental problem of key establishment over open wireless channels was recently highlighted in [4], where the authors used an off-the-shelf key exchange protocol combined with the technique of *uncoordinated frequency hopping (UFH)* for exchanging the messages of the protocol. However, this paper does not provide any systematic means to analyze the performance of the protocol in terms of its communication cost under various active attacks. From a communications perspective, the UFH technique proposed in [4] has limitations when considering a sophisticated follower-jammer in which case this technique can be highly inefficient.

Another technique for key establishment over wireless channels was recently proposed in [5] in which the receiver stored wideband signals and worked backwards to decipher the messages. This scheme has two limitations: one, the process of storing wideband signals can be easily disrupted by a jammer by saturating the front-end receiver filter of the wireless node (this is why current military radios employ frequency-hopping), and second, the original message transmission employs successively weaker signal spreading which can be easily jammed thereby disrupting the backward decoding process.

Another line of research in [6], [7], [8] considered the use of wireless fading to establish a secret key; the main idea here is to exploit the property that a wireless fading coefficient between two nodes is reciprocal, random and spatially independent, from which a secret key can be extracted. However, it was argued recently in [9] that communication cost per bit of secret key established is an important performance measure in wireless environments and, from this viewpoint, extracting a secret key from fading coefficients is highly inefficient under active adversarial attacks. Thus, this technique is best suited only for limited scenarios of passive adversary and low external noise.

B. Our Contributions

What emerges from our study is a *a quantitative analysis method for cryptographic protocols operating over wireless channels* which, when applied to the example of key exchange in a jamming environment, immediately demonstrates that traditional measures of security – such as forward security, or whether the authentication provided is implicit or explicit – are *in*sufficient to identify which are the 'best' protocols according to suitable measures of efficiency and security. The analysis method is generally applicable, easily accessible to the community, and has the potential to provide a standard methodology by which cryptographic protocols for wireless channels can be evaluated.

The analysis method allows us to *quantitatively compare a number of 'classical' approaches to key exchange in a jamming environment*. For example we show that, contrary to what might be expected from [4], explicit authentication of individual protocol messages via digital signatures is not the most energy-efficient approach in the face of a jamming adversary. More extreme, we show that if we are prepared to give up on forward security for our safe channel, then the very simple SOK protocol [10] is difficult to beat in terms of its communication costs and anti-jamming properties.[1] Thus, from the perspective of 'classical' key exchange, our paper has something quite new to say: the consideration of jamming adversaries is a game-changer when it comes to deciding how to select a key exchange protocol.

And, finally, we examine the problem from a different direction: we show how *state-of-the-art key exchange methods* and *physical layer communications techniques* can be combined to thwart the jamming adversary, whilst maintaining *communications and computational efficiency* for the legitimate network nodes. In particular, we augment the 'classical' approach

[1] A justifiably skeptical reader might question the abandonment of forward security here. However, once a secure spread-spectrum channel is established, it is a trivial matter to efficiently arrange for forward security by simply running an unauthenticated Diffie-Hellman key exchange over the channel. An adversary without the spreading code would have to record the entire channel bandwidth to even be able to extract the Diffie-Hellman messages if the spread-spectrum code is later compromised. And the recording of wideband channels is challenging, particularly in an environment with significant electromagnetic interference from radars and jamming.

by the use of a publicly-announced (but randomly chosen) spreading code – we refer to this as *physical layer augmented key exchange*. Informally, the main benefit of using such a code is that an adversary who is either not prepared to spend the energy to continuously monitor the communications path between legitimate nodes or experiences significant multipath fading may miss this announcement, and will not then be able to efficiently jam the subsequent messages exchanged between the nodes.

C. Paper Organization

Section II presents the adversary and wireless communications models, including how to calculate probabilities for message transmission success/failure for various types of communication systems operating over wireless channels. Section III details the analysis method, while Sections IV through VI describe the application of the analysis technique to secure key establishment in a jamming environment. Finally, Section VII presents the conclusions.

II. System Model

The analysis tools presented in this paper can be applied to generic protocols, both cryptographic and others, operating over general wireless communication channels. However, to motivate the analysis tools and clarify the presentation, we provide in this section a basic adversary and system model for the wireless network scenario. In addition, we provide a tutorial treatment illustrating how to extract parameters from the wireless channel model that are required as inputs to the probabilistic algorithm analyses later in the paper.

A. Adversary Model

Consider three entities, Alice, Bob and Eve, where Alice and Bob are legitimate nodes that want to communicate over an *open wireless channel* while Eve is an adversary that wants to eavesdrop/disrupt this process. By an "open" wireless channel, we mean a wireless channel whose parameters (e.g., frequency, power, channel encoding, etc.) are known publicly. We assume that Eve can listen to messages exchanged between Alice and Bob, and mount attacks only through the wireless channel; i.e., besides wireless signal transmission/reception there is no other mechanism available for Eve. Such other mechanisms are clearly beyond the scope of this paper since our focus is on modeling attacks specific to the wireless channel. Eve can passively eavesdrop as well as actively disrupt the key exchange process. In particular, Eve can transmit her own data, fake messages and/or a random noisy signal; however, all of Eve's transmissions occur through the wireless channel and are subject to the natural signal fading phenomenon caused by reflections in the environment that is incurred by the legitimate nodes. Thus, while Eve can control her own transmissions, she cannot control the fading characteristics of the wireless channel. As a result, every strategy of Eve is subject to probabilistic success which can be high or low depending on the physical parameters such as location, transmit power, etc.

We assume that Eve has bounded transmission power (i.e. an energy expenditure of some finite β per unit time), as this is a basic hardware limitation of all radio transmitters. However, we will assume that there is no limitation on the total energy expended by Eve; in other words, whereas Alice and Bob may be battery-operated wireless nodes for which limiting energy expenditure is paramount, Eve can be plugged into a wall outlet and thus have no concerns about battery lifetime.

Finally, Eve can be located anywhere, but we assume that there is a non-zero distance between Eve and the legitimate nodes; in other words, there is a "safe range" around Alice and Bob in which Eve cannot be located. At first glance such an assumption seems like a weakness relative to attacker models that employ an "adversary everywhere" approach, but it simply is to avoid the singularity when Eve is located exactly at Alice or Bob's location, which cannot occur in practice. For many of our envisioned applications (e.g., communication between two dismounted soldiers), it is clear that an adversary extremely close to Alice or Bob can be eliminated by physical means. And, perhaps more pragmatically, it will be clear from the numerical results that the main conclusions of this paper on both the utility of the analysis methodology and the comparison of key exchange schemes are accentuated as the safe range becomes smaller.

B. Wireless Communication Model

We assume a network where Alice and Bob are at fixed locations a distance d_{AB} apart. Per above, Alice and Bob are each surrounded by a safe range where Eve is known not to be located. The radius of this region, r_s, is an important parameter, which is directly related to the amount of jamming power incurred. Eve can be anywhere outside the safe ranges. Since Eve's disruption is maximized when she lies on the line between Alice and Bob, we assume a linear network model with all three nodes on a single line as given in Fig. 1.

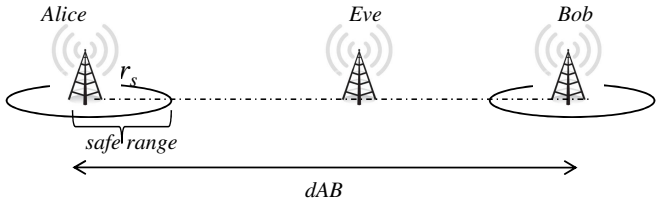

Fig. 1. Description of the wireless setting where Alice, Bob and Eve are located on a line. Alice and Bob are the legitimate nodes with a distance d_{AB} apart. Both have some certain safe range with a radius r_s surrounding them. Eve can be located anywhere outside safe ranges, however it is optimal for Eve to be located on the line to deliver maximum jamming power.

In this section, we introduce the communications models that will be used to generate inputs for the analyses of protocols. While the calculations are based on a careful study of the physical layer, much of the detail is reserved for Appendix A so as to not interrupt the main flow of the paper.

There are two effects that impact the received power in the wireless environment: large scale path-loss caused by the distance between the transmitter and receiver, and small scale

multipath fading caused by signal reflections, as described in detail below.

First, consider path-loss. When a transmitter transmits, the signal spreads out in space, and hence the power density decreases the further one is from the transmitter. Hence, the energy received without any other impairments at a node B from transmission by a node A is given by:

$$E_{rcv} = E_{send}/d_{AB}^{\alpha} \tag{1}$$

where, d_{AB} is the distance between node A and node B, and α is the "path-loss exponent", which generally ranges from 2 to 4 depending on the environment.

However, the primary impairment impacting the signal in wireless communication systems is multipath fading, which is caused when the transmitted signal reflects off objects in the environment. The difference in the length of the paths followed by these reflections relative to the direct signal and, hence, their relative phase, causes the reflections and main signal to add destructively or constructively at various points in space. This causes the actual received signal power to vary randomly around that in (1). Conceptually, what happens is the same as when one throws two rocks in the pond: the interference of the two sets of waves cause a spatial pattern of peaks and troughs. A receiver essentially takes a sample of this spatial pattern at its location that can be above or below the average that would be expected at that point. The random variation in received power is also time-dependent. This is due to the signal taking different paths in time caused by movement of the receiver and/or other objects in the environment. Hence, the actual received signal energy is a random value that depends on location, time, and (as described next in detail) the frequency of the transmitted signal.

Important for a basic understanding of the design of protocols for wireless channels is an understanding of how the multipath fading affects narrowband and wideband communication systems. A narrowband system is one that occupies only a narrow frequency range, an example of which were early analog cell phones of bandwidth 30 kHz. A wideband system is one whose signal occupies a much larger frequency range, an example of which would be more recent spread spectrum cell phones with bandwidths on the order of 1 MHz. The spatial fading pattern described above also depends on the frequency of operation. The fading pattern is correlated for two frequencies located close enough to one another. For example, in a typical urban outdoor environment, it takes roughly a 100 kHz separation between two frequencies before the spatial multipath fading patterns at those two frequencies can be assumed to be independent [11]. Hence, a narrowband system will be impacted on by essentially a single pattern corresponding to the frequency it is operated around. In contrast, wideband systems, if designed properly, can achieve the average of a large collection of fading patterns. By the law of large numbers, each instantiation then approaches its average, and the fading is essentially eliminated.

The above rough description of the phenomenon allows the derivation of the probability that a packet is received under each of the narrowband and wideband models. Modern communication systems demonstrate a threshold effect in the received signal-to-interference-plus-noise ratio (SINR): a packet is received with high probability if the SINR is above a given threshold γ, and the packet is lost if the received SINR is below γ.

As described in Appendix A, for a narrowband system the random power variation caused by multipath fading is modeled as an exponential random variable. Let $P_{\text{rcv}}^{(B\leftarrow A)}$ be the probability that a packet sent from A is received at B; then, from Appendix A,

$$P_{\text{rcv}}^{(B\leftarrow A)} \quad = \quad \exp\left(-\gamma \frac{N_0}{E_A} d_{AB}^{\alpha}\right) \tag{2}$$

for a transmission from A to B on a narrowband channel, where N_0 is a parameter proportional to the power of the thermal noise in the receiver.

In this work, we are also interested in this probability where an active adversary is also present sending a jamming signal over the narrowband channel. The jamming power received from E is also subject to fading and hence modeled as an exponential random variable, but with a different parameter. The probability of successful reception under jamming is the found as

$$P_{\text{rcv}}^{(B\leftarrow A)}|\text{Jamming} \quad = \quad \frac{\exp\left(-\gamma \frac{N_0}{E_A} d_{AB}^{\alpha}\right)}{1+\gamma \frac{E_A/d_{AB}^{\alpha}}{\beta/d_{BE}^{\alpha}}} \tag{3}$$

Notice that, as expected, it becomes more likely to receive when the attacker's transmit energy β is lower or her distance to B, d_{BE}, is larger. Details of this calculation are given in Appendix A.

The probability values for the wideband channel are found next. Per above, a well-designed system for a wideband channel will mitigate the multipath fading through averaging; hence, the packet is either received or not based on the path-loss incurred on the transmission (1): hence, for a wideband wireless channel:

$$P_{\text{rcv}}^{(B\leftarrow A)} = \begin{cases} 1, & \text{if } \dfrac{E_A/d_{AB}^{\alpha}}{N_0} \geq \gamma, \\ 0, & \text{if } \dfrac{E_A/d_{AB}^{\alpha}}{N_0} < \gamma. \end{cases} \tag{4}$$

III. Analysis Method

A. Introduction

A cryptographic protocol over a wireless channel can be regarded as a dynamical system, which probabilistically passes through a number of stages. As a simple example, consider the simple key exchange protocol described in Fig. 7. The protocol has three steps during which A and B exchange messages. They then switch to communicating on a spread spectrum communication channel that is determined by the session key K_{AB} established by the protocol. Consider the corresponding flow diagram in Fig. 8. From any given state, the next step depends on the occurrence of some random event based on the wireless channel. Moreover, each step has an associated

cost to Alice and Bob, which causes the overall cost of the key exchange to be random.

Before presenting a detailed step-by-step approach on how to evaluate the performance of dynamical systems on wireless channels, first we give a brief discussion of the underlying theory. A dynamic probabilistic system can be modeled as a Markov process. The stages in the system correspond to states in a Markov process. The branches connecting the stages of the protocol correspond to the state transitions of this Markov process, and each state transition has an associated probability. In our case, however, each transition incurs a cost (e.g. energy or delay) to Alice and Bob. Our aim is to study the Markov process and find the distribution of its overall cost from the initial state to final state. To accomplish this, we employ linear system theory tools used to evaluate dynamic systems in operations research and decision analysis [12]. In this method, the Markov state diagram is mapped to a signal flow diagram where states become nodes and branches that represent state transitions become systems that the signal passes through. The biggest advantage of mapping the Markov process to a linear system is that the rich set of tools available for linear system analysis then becomes available to us. In particular, once the Markov process is mapped to a signal flow diagram, reduction methods can be used to find the transfer function between the input and output signals. This transfer function is the moment-generating function (mgf) of the random cost, and it will have great utility in answering questions of interest.

Definition 1: Let X be a real-valued random variable. The *moment-generating function* of X is defined as

$$M_X(s) = \mathrm{E}(e^{sX}), \tag{5}$$

where $\mathrm{E}(\cdot)$ corresponds to expectation.

We sometimes are interested in the joint behavior of two random costs X, Y (e.g., energy and delay). The *moment-generating function* of X, Y is defined as

$$M_{XY}(s,t) = \mathrm{E}(e^{sX}e^{tY}) \tag{6}$$

Note that the marginal mgf can be easily found by the following relation: $M_X(s) = M_{X,Y}(s,0)$

The mgf of a random variable completely characterizes its distribution and the exact probability density function can be found by an inverse transform. However, often a statistic of the cost is of more interest than the complete distribution. The moment-generating function, as one would expect from its name, readily supplies such; in fact the cross of the m^{th} moment of X and the n^{th} moment of Y is given by:

$$\mathrm{E}(X^m Y^n) = \frac{\partial M_{XY}(s,t)}{\partial s^m \partial t^n}\Big|_{s=0,t=0}, \tag{7}$$

and the expected value of X (of Y) is just the special case with $m = 1$ and $n = 0$ ($m = 0$ and $n = 1$).

Often it is also of interest to know how likely it is that the cost will exceed a certain amount; for example, a radio might have some initialization period during which the protocol must be executed, and hence it is of interest to know the probability that the protocol is not executed within that period. One would find this by calculating the so-called "tail probability," which is readily bounded with the mgf through the Chernoff bound:

Chernoff Bound

Let X be a random variable, and $M_X(s)$ its moment-generating function. Then

$$P(X \geq c) \leq \min_{s \geq 0}(e^{-sc}M_X(s)), \forall c \in \mathbb{R} \tag{8}$$

Markov Process	Linear System
State Diagram	Signal Flow Diagram
State	Node
Initial State	Input Node
Final State	Output Node
State transition from S_i to S_j	Signal passing through a linear system
Branch with prob. p_{ij}, cost X, Y units	Linear system with transfer function $h(s) = p_{ij}e^{sX}e^{tY}$
End-to-end cost distribution	Overall impulse response
Moment-generating function of cost distribution	Transfer function of the whole system

TABLE I
MAPPING A MARKOV PROCESS TO A LINEAR SYSTEM

B. Methodology

In this section, we demonstrate how to obtain the mgf of the random cost of interest. First, consider the state diagram for an arbitrary Markov process as given in Fig. 2. To map this state diagram to a signal flow graph, each state is replaced with a small black circle to represent a node. The transition from state i to state j is labeled with $p_{ij}e^{sX}e^{tY}$, where p_{ij} is the probability of transitioning to state j from state i, X is the cost of that transition in the first metric of interest, and Y is the cost of that transition in the second metric of interest. This mapping is also summarized in Table I. The probability of the transition can be found from the wireless system calculations of Section II-B, whereas the costs will be clear from the protocol description.

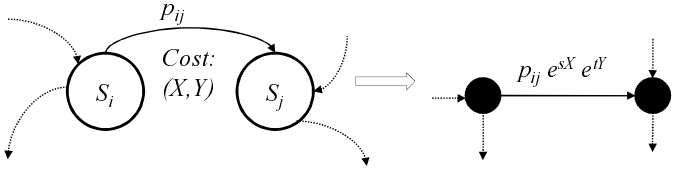

Fig. 2. The Markov state diagram is mapped to a signal flow diagram. Each state is mapped to a node, each state-transition is mapped to a branch between nodes.

Next, the system flow diagram is systematically reduced through a number of transformations as shown in Figure 3 until a single branch from the initial state to the final state is obtained. The label of that branch corresponds to the joint mgf of X and Y. A simple example in the next section will make this straightforward procedure clear.

C. Example

Consider a simple wireless communication algorithm where Alice wants to send a message to Bob. We are interested in the

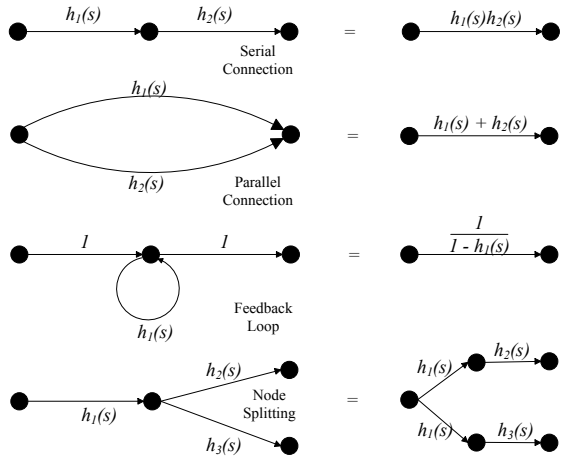

Fig. 3. Basic equivalences in signal flow diagrams. These equivalences can be used to reduce any signal flow diagram to a single branch containing the input and output nodes.

total energy and time (delay). Alice first probes the channel. If the channel is "good", which happens with probability p_G, she transmits with parameters chosen for the good channel, where the message costs an (energy, delay) value of (N_1, D_1) per transmission. Alice keeps repeating the message until it is delivered, where for each trial she is successful with probability p_1. Similarly, if the channel is "bad", she does the same with a cost of (N_2, D_2) per trial with success probability of p_2. We model this communication example as a Markov process with four states as given in Fig. 4. The initial state is where Alice probes the channel, two intermediate states are the cases of the channel being good or bad, and the final state is where Bob received the message correctly.

As shown in Fig. 4, this Markov process is mapped to a signal flow diagram. The transfer function of this signal flow diagram is found using the reduction methods of Figure 3, and the transfer function is found to be:

$$M_{XY}(s,t) = \frac{p_G p_1 e^{(1+N_1)s} e^{(1+D_1)t}}{1-(1-p_1)e^{N_1 s}e^{D_1 t}} + \frac{p_B p_2 e^{(1+N_2)s} e^{(1+D_2)t}}{1-(1-p_2)e^{N_2 s}e^{D_2 t}},$$

where X and Y are the random variables denoting the overall costs in terms of energy and delay, respectively. Suppose we would like to find the expected value of the energy Alice spends to deliver the message. Then

$$\begin{aligned} \mathrm{E}(X) &= \frac{\partial M_{XY}(s,t)}{\partial s}\Big|_{s=0,t=0} \\ &= \frac{\partial}{\partial s}\left(\frac{p_G p_1 e^{(1+N_1)s}}{1-(1-p_1)e^{N_1 s}} + \frac{p_B p_2 e^{(1+N_2)s}}{1-(1-p_2)e^{N_2 s}}\right)\Big|_{s=0} \\ &= p_G(1+\frac{N_1}{p_1}) + p_B(1+\frac{N_2}{p_2}) \end{aligned}$$

For this example, the above value can be verified easily by inspection. For the second cost metric Y, we find the bound on the tail probability using the Chernoff bound in (8). Suppose we are interested in the probability that delivering a message to Bob takes longer than a threshold of c unit time.

$$\begin{aligned} P(Y \geq c) &\leq \min_{t\geq 0}(e^{-tc}M_Y(t)) \\ &= \min_{t\geq 0} e^{-tc}\Big[\frac{p_G p_1 e^{(1+D_1)t}}{1-(1-p_1)e^{D_1 t}} + \frac{p_B p_2 e^{(1+D_2)t}}{1-(1-p_2)e^{D_2 t}}\Big] \end{aligned}$$

We next illustrate how the physical model is used to calculate the parameters for numerical evaluation. The probability and cost values in the signal flow diagram in Fig. 4 are found by the wireless communication model given in Section II. Suppose when Alice has a good channel, she sends her message with a transmit power of $E_A = 1$mJ/s, where transmission happens at a rate such that each message takes $D_1 = 10$ms to transmit. Therefore, the sending of each message costs $N_1 = 10\mu$J. Suppose the distance between Alice and Bob is $d_{AB} = 20$, where the path-loss exponent of the medium is $\alpha = 2$. Suppose the thermal noise power is $N_0 = 1\mu$J/s, and the SINR threshold for successful decoding is $\gamma = 0.5$. Then the value p_1 in the analysis is given by

$$p_1 = P_{\mathrm{rcv}}^{(A\to B)} = \exp\left(-\gamma \frac{N_0}{E_A} d_{AB}^{\alpha}\right) \approx 0.82$$

With similar calculations, we find for the bad channel, $p_2 \approx 0.37, N_2 = 40\mu$J, $D_2 = 20$ms. Finally, assume $p_G = 0.3$ and $p_B = 0.7$, and probing the channel costs energy and delay of 1μJ and 1ms, respectively.

The expected energy cost is then

$$\mathrm{E}(X) = p_G(1+\frac{N_1}{p_1}) + p_B(1+\frac{N_2}{p_2}) \approx 80.3\mu\text{J} \qquad (9)$$

Next, we calculate the upper bound on the tail probability for the case that sending a message to Bob takes more than $c = 250$ms.

$$P(Y \geq c) \quad \leq \quad \min_{t\geq 0}(e^{-tc}M_Y(t)) \approx 0.047, \qquad (10)$$

where the upper bound is found by a numerical linear search in t. Hence, the message is delivered within 250ms with more than 95% confidence.

IV. Key Exchange Protocols

A key exchange protocol provides a mechanism by which two parties A and B can generate a common secret key (or session key) while communicating over an insecure channel. Many different security models and security definitions for key exchange protocols have been developed by the cryptographic research community (see for example [13], [14], [15], [16]). A consensus has now emerged around a few essential security properties. *Session-key security* refers to the property that the compromise of one (or many) session key(s) should not affect the security of other session keys. *Forward security* refers to the property that past session keys are not compromised even if the long-term keys of the parties are. The prevention of *unknown key-share attacks* and resilience to *key-compromise impersonation attacks* are also considered important, if secondary, goals. For our analysis, an important characteristic will be whether the messages in the protocol are *explicitly*

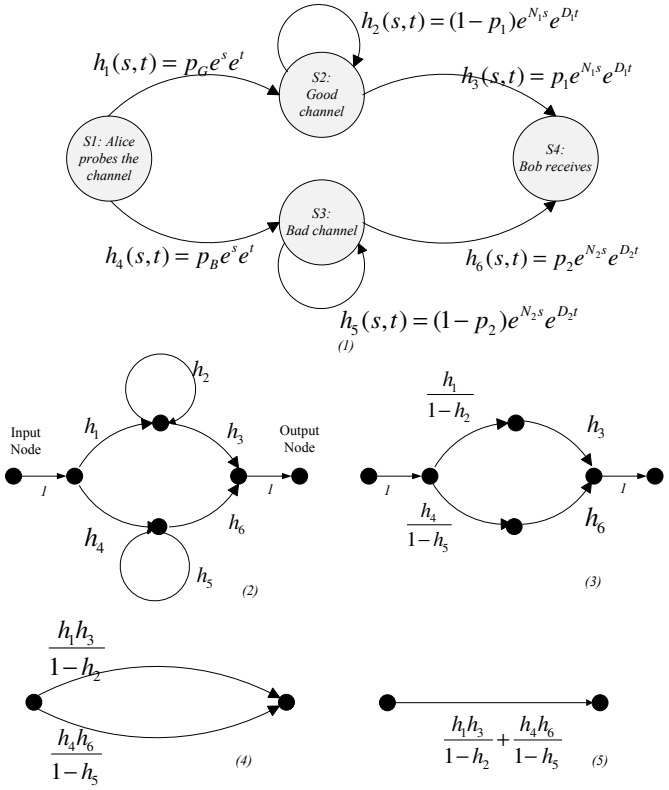

Fig. 4. An example which demonstrates the analysis tool. The cost of a simple wireless communication algorithm is analyzed. The algorithm is described as a Markov process with four states. For each state transition, there is an associated probability and a cost value. For the analysis, the state diagram is mapped to a signal flow graph using the conversions in Table I. Then the signal flow graph is reduced to an equivalent circuit to obtain an analytical closed-form expression for the transfer function (mgf of the cost) using the rules in Fig. 3.

authenticated or not. In the former case, each message is accompanied by a digital signature and a time-stamp, giving a proof of origin and freshness. In the latter case, parties in the protocol do not have such guarantees, but still obtain an *implicit* key authentication property: each party is assured that only the other *could* generate the same session key.

We are interested in evaluating the performance of different types of key exchange protocol in wireless environments, in the face of a sophisticated active adversary. We will consider protocols in the traditional PKI setting, where each node is equipped with a public/private key pair, as well as protocols in the identity-based setting. There, each node is associated with an identifier (such as a network address) from which its public key can be derived, while a trusted authority (TA) will pre-provision each node with the private key corresponding to its identifier.

In the remainder of this paper, we will consider four representative protocols, each of a different type. To maintain a level playing field for fair comparison of the different protocol types, we will assume that all nodes have their public key certified by a single CA in the PKI setting, and that this CA's public key is hard-coded into the network nodes. We will also assume that certificates consist solely of the CA's signature on the nodes' public keys. (In reality, certificates are much larger and more complicated data structures than this.) Similarly, we will assume that, for the identity-based setting, each node obtains its private key from a single TA and that this TA's public parameters are hard-coded into the network nodes. In addition, we assume that, since the networks we study are large and dynamic, each node holds its public key (or identifier) and corresponding private key, but does not know the public keys (or identifiers) of other network nodes. Pre-empting our later analysis, we focus on minimising the message sizes, using state-of-the-art cryptographic schemes to do so. Finally, we take 128 bits as the target security level for all our key exchange protocols. This is appropriate for the protection of, for example, classified information in tactical military networks, but may be higher than is required in a commercial sensor network, say. Since we wish to prioritize communications efficiency above computational performance, this target security level implies that we will make extensive use of elliptic curve and pairing-based cryptographic techniques. Details of these are beyond the scope of this paper, but we include sufficient references to enable the interested reader to verify our parameter choices.

We start by considering a classical PKI-based Diffie-Hellman key-exchange protocol (Fig. 5, left). This protocol is *forward secure* and the protocol messages are *explicitly authenticated*. In this solution, A sends B a message of the form $M_A = (pk_A, c_A, DH_A, t_A, \sigma_A)$, where pk_A is A's public key (256 bits), c_A is the certificate on her public key (256 bits), DH_A is the Diffie-Hellman value (256 bits), t_A is a time-stamp (32 bits) and σ_A is a signature on the whole message, for authentication (512 bits). Here we use a BLS signature [17] for the certificate. This allows us to minimise the signature size and exploits the fact that we do not need to transmit the CA's public key. (For the BLS scheme, at the 128-bit security level, we need to use a BN curve [18], asymmetric pairings and sextic twists. This gives a signature size of 256 bits but a CA public key size of 512 bits.) We use the ECDSA scheme for the signature σ_A, in order to minimise the sum of the size of this signature and the corresponding public key (totalling 768 bits at the 128-bit security level) whilst avoiding the relatively expensive pairing calculations that would be needed in the BLS scheme[2]. We assume that the Diffie-Hellman exchange takes place over a pre-agreed elliptic curve group whose elements can be represented by 256 bits, using point compression. Similarly, B sends message M_B to A of the same form; A and B can then create their session key by applying a key derivation function to the shared Diffie-Hellman value and their public keys. The total cost per message is 1312 bits. We note the requirement of this protocol that the protocol participants have synchronized clocks or access to a coordinated time service; this is a non-trivial issue

[2]We note that, at the 128-bit security level, a pairing calculation costs on the order of 10-20 times an elliptic curve point multiplication [19]; so, while the cost of a pairing computation is not prohibitive, it is useful to minimise their number when selecting a protocol.

in practice.

We next consider an analogous protocol in the identity-based setting. We consider a *forward secure* identity-based Diffie-Hellman key-exchange (see Fig. 5, right), whose messages are *explicitly authenticated*. Here, A sends B a message of the form $M_A = (id_A, DH_A, t_A, \sigma_A)$, where id_A is her identifier (32 bits), DH_A is the Diffie-Hellman value (256 bits), t_A is the time-stamp (32 bits) and σ_A is an identity-based signature on id_A, DH_A and t_A (512 bits). Here, the main advantage of the identity-based approach over the PKI-based approach can be seen: there is no longer any need for the nodes to exchange certificates and public keys; instead an exchange of short identifiers suffices. To instantiate the identity-based signatures, we use the BLMQ scheme [20] over BN curves. At the 128-bit security level, signatures in this scheme consist of an element of $\mathbb{Z}_p^*$, where p has 256 bits, together with an element of a 256-bit elliptic curve group. B sends a similar message M_B to A. Here the exchanged messages are each 832 bits long. Again, this protocol requires synchronized clocks.

Our third protocol, the SCK-2 protocol from [21], is an *implicitly authenticated*, *forward secure* identity-based protocol. Here A sends B a message of the form $M_A = (id_A, DH_A)$ and B sends a similar message M_B to A (see Fig. 6, left). The session key is calculated by combining the Diffie-Hellman private and public values, the identities and the private keys in a particular way that is detailed in [21]. Here, in fact, there are a number of possible protocol designs that we could have selected, with [19] providing a useful survey of the available alternatives. We have selected the SCK-2 protocol because of its low bandwidth consumption (here, the protocol messages are only 288 bits each at the 128-bit security level) and its proven security properties [21], [19].

Our fourth protocol, the SOK protocol [10], is also *implicitly authenticated*, but sacrifices the forward security property in order to reduce bandwidth to a minimum. In the basic version of this protocol A simply sends her identifier id_A to B and B sends his identifier id_B to A; the two parties then combine their respective identifiers and private keys in a specific way to obtain a shared session key. We augment this basic protocol with 32-bit nonces (see Fig. 6, right), with these nonces being included in the key derivation step, in order to prevent the agreed key from being a static value. The exchanged messages are still very short, just 64 bits each. To instantiate this protocol efficiently at the 128-bit security level, we use BN curves and asymmetric pairings, equipping each party with two private key components, one in each group input to the pairing operation and using an ordering on node identifiers to determine in which order hashed identifiers/private key components are input to the pairing operation. These modifications do not affect the bandwidth consumption of the protocol. Basic versions of this protocol (without nonces) were proven secure in [22], [23]. In Table II we summarize the properties of the protocols considered so far.

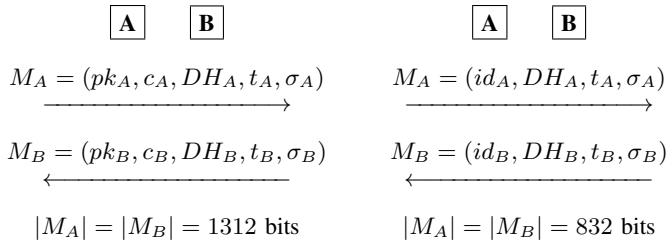

Fig. 5. Protocol 1: PKI-based Diffie-Hellman (left); Protocol 2: Identity-based Diffie-Hellman (right)

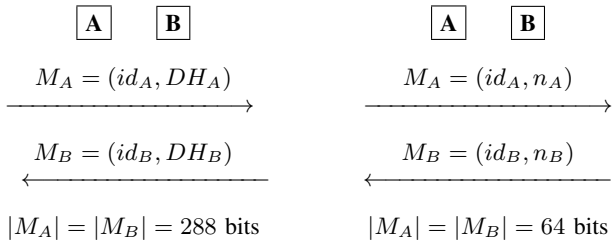

Fig. 6. Protocol 3: Forward secure, implicitly authenticated, identity-based (left); Protocol 4: SOK (right)

V. Evaluation of Key Exchange Protocols

A. Cost Analysis

The key exchange process of interest is summarized in Fig. 7 as a three-step process. For the purpose of cost analysis, we divide the protocols in Table II into two classes: (1) protocols with explicitly-authenticated messages, and (2) protocols with implicitly-authenticated messages. The reason for this is that the flow diagrams for protocols in each class are identical, although, with different costs associated with the branches.

1) Protocols with explicitly-authenticated messages: The flow chart common to both Protocol 1 and Protocol 2 is given in Fig. 8. Key exchange ends when A and B switch to communicating on a spread spectrum channel which is determined by the established session key. Although other metrics will also be considered below, we assume that E is focused on making A and B expend as much energy as possible. Hence, during the transmission of M_A (Step 1), E transmits noise to jam the communication, since transmitting a fake message does not help as the messages are explicitly authenticated; therefore, the goal of E is to minimize the probability of reception of M_A by B. For Step 2, Eve listens to the channel, and if she detects the transmission of M_B, she again transmits jamming noise.

Following the approach in Section III, the flow chart for a given protocol is converted to its corresponding signal flow diagram and then simplified to obtain the transfer function. For the protocols characterized in Fig. 8, the simplified flow diagram is given in Fig. 9. The diagram is completed by finding the transfer functions for the branches $h_1(s,t), \cdots, h_4(s,t)$, which is equivalent to calculating each branch's associated probabilities and costs. Only two probabilities are required in

	Message size (bits)	Authentication	Forward security
Protocol 1	1312	Explicit	Yes
Protocol 2	832	Explicit	Yes
Protocol 3	288	Implicit	Yes
Protocol 4	64	Implicit	No

TABLE II
COMPARISON OF KEY EXCHANGE PROTOCOLS.

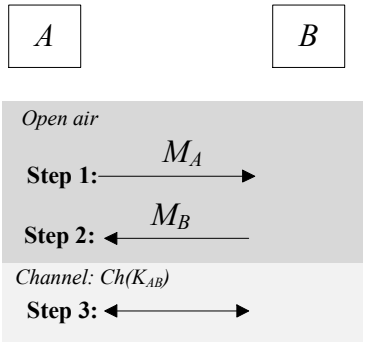

Fig. 7. Description of the key exchange protocol over a wireless channel. A and B exchange messages M_A, M_B over the publicly known wireless channel referred to as open air. Then they switch to a spread-spectrum channel given by the session key K_{AB}, where they can communicate efficiently and free from jamming attacks as long as K_{AB} is secret.

this diagram: p_{BA} and p_{AB}, the probability of Bob receiving Alice's transmitted message M_A, and the probability of Alice receiving Bob's transmitted signal M_B, respectively, which can be found using the communication model in Section II.

The costs in energy and delay for each branch are then calculated as follows. Let N_A and N_B be the transmit energy spent for transmitting the messages M_A and M_B, respectively. The energy expended on the branches in the system flow diagram are then given by $N_1 = N_3 = N_A$ and $N_2 = N_4 = N_B$. Likewise, if D_A and D_B are the time it takes to send messages M_A and M_B, respectively, the delay cost of the branches h_1, h_2 are $D_1 = D_A$, and $D_2 = D_B$. The branch h_3 corresponds to the case where Alice sends her message but Bob misses and keeps silent. Per above, in that case we assume Eve will also remain silent, so Alice will quickly sense that there is no message on the channel and switch to transmit mode and send M_A again, hence implying $D_3 = N_A$, and, likewise, $D_4 = D_B$. Note that, on the other hand, if Eve was focused on maximizing the delay of the protocol, she would choose to send a fake message during Step 2 even when Bob misses M_A to cause Alice to waste time. This would increase the third branch's cost to $D_3 = (D_A + D_B)$.

The overall transfer function is now obtained using the reduction given in Fig. 9. Let X, Y be the random variables denoting the total cost in transmit energy and delay, respectively. The moment-generating function of the cost is then:

$$M_{XY}(s,t) = \frac{h_1 h_2}{1-(h_1 h_4 + h_3)}, \quad \text{where}$$

$$\begin{aligned} h_1(s,t) &= p_{BA} e^{(sN_A + tD_A)}, \\ h_2(s,t) &= p_{AB} e^{(sN_B + tD_B)}, \\ h_3(s,t) &= (1-p_{BA}) e^{(sN_A + tD_A)}, \\ h_4(s,t) &= (1-p_{AB}) e^{(sN_B + tD_B)}. \end{aligned} \tag{11}$$

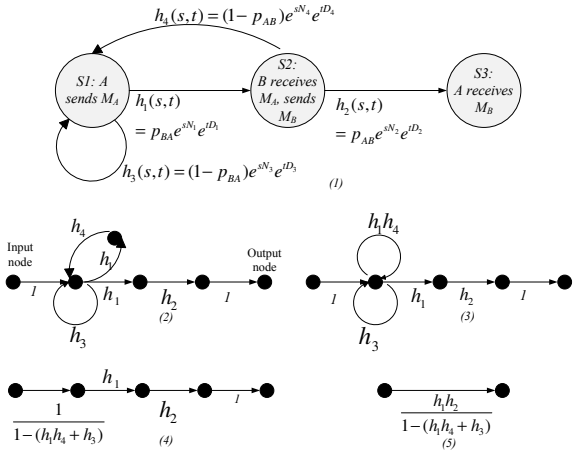

Fig. 9. Markov state diagram for Protocols 1 and 2, and the corresponding signal flow diagram reduced to a single branch step by step. The state diagram corresponds to the flow chart in Fig. 8, simplified to three states. The transfer function of the final signal flow diagram gives the mgf of the cost.

2) Protocols with implicitly-authenticated messages: A differentiating feature of protocols employing implicit authentication is that they do not allow a node to immediately discard a fake message successfully inserted by an adversary. The flow chart for Protocols 3 and 4 is given in Fig. 10. When Bob receives a message M_E from Eve, he will reply with his message M_B and will attempt to compute a key K_{BE}; however, since Eve does not possess the correct keying materials she will not be able to compute the same key and the process will fail, forcing Bob to return to the open air channel to listen. While not compromising security, this will incur a transmit energy cost for Bob, which was not present in explicitly-authenticated protocols. While Bob is replying to Eve's message, Eve may also send a fake message to Alice, which also causes her to generate some key K_{AE} and move to the corresponding spread spectrum channel; again, the process fails and Alice returns to the start of the protocol. While protocol re-start is more likely in implicitly authenticated protocols, the advantage is that the message sizes are shorter and the cost incurred is lower in each cycle. Thus, we can see a tradeoff in cost incurred by a protocol versus its other properties.

We simplify the flow chart to a signal flow diagram as given in Fig. 11. The additional state S_4 is necessary for the case that Bob receives Eve's message. Three probabilities are required on the branches in the diagram: p_{BA}, p_{BE}, and p_{AB}. The probabilities p_{BA}, p_{BE} are calculated using the insight that during Step 1, Eve will insert a legitimate message instead of random noise. The calculation of p_{AB}, however, remains the same as in the explicitly-authenticated case.

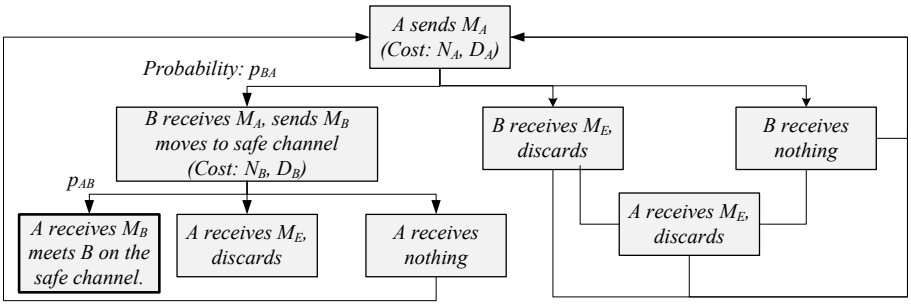

Fig. 8. The flow chart for the key exchange described in Fig. 7 for Protocols 1 and 2. In these two protocols explicitly-authenticated messages are exchanged. The protocol is initiated by A sending M_A, which costs an (energy, delay) value of (N_A, D_A). B may receive this message, receive a message M_E from an attacker, or may not be able to decode the signal at all, each with a certain probability. The protocol is completed after a random series of retrials until A, B successfully receive each other's messages and start communicating over the channel $Ch(K_{AB})$.

Borrowing from Fig. 9, the cost only needs to be calculated for the branches connecting S_1 and S_4. The branches labeled g_1, g_2 have energy cost values of $N_{g1} = N_A$ and $N_{g2} = N_B$. To calculate the costs in terms of delay, we assume that when Bob receives M_E, he will reply, so Alice will spend time to receive this message. Hence the branch g_2 has a delay of $D_{g2} = D_B$, and, likewise, the delay for g_1 is $D_{g1} = D_A$. The self-loop branch g_3 is identical to the case in the explicitly-authenticated protocol.

The moment-generating function for the end-to-end cost is given as $M_{XY}(s,t) = \frac{h_1 h_2}{1-(h_1 h_4 + h_3)}$, where:

$$\begin{aligned} h_1(s,t) &= p_{BA} e^{(sN_A + tD_A)}, \\ h_2(s,t) &= p_{AB} e^{(sN_B + tD_B)}, \\ h_3(s,t) &= (1 - p_{BA} - p_{BE}) e^{(sN_A + tD_A)} \\ &\quad + p_{BE} e^{\left(s(N_A+N_B) + t(D_A+D_B)\right)}, \\ h_4(s,t) &= (1 - p_{AB}) e^{(sN_B + tD_B)}. \end{aligned} \tag{12}$$

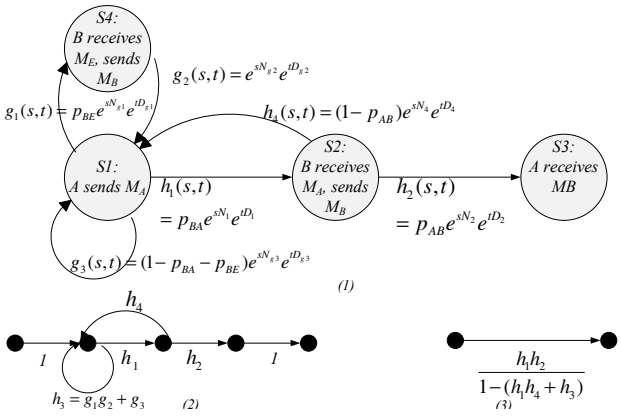

Fig. 11. Markov state diagram for Protocols 3 and 4, and the corresponding signal flow diagram reduced to a single branch step by step. The state diagram corresponds to the flow chart in Fig. 10

B. Numerical Results

The number of physical layer symbols transmitted can be converted to units of the cost metric using the specifications of the wireless devices. For our calculations, we assume nodes send messages with a transmit energy of 10dBm, which is equal to 10mJ/s, and the symbols are transmitted at a rate of 1Msymbols/sec. Hence sending each physical layer symbol costs $10^{-2}\mu$J, and takes 1μs.

1) Transmit Energy Cost: Plots are given in Fig. 12 for the expected energy expenditure of Protocols 2 and 3. For these plots, we assume a safe radius $r_s = 1$, and we plot the cost as a function of E's location, which varies from $d_{AE} = 1$ to $d_{AE} = 9$. For each value of d_{AE}, we calculate the expected value of the cost using the mgf's given in Eqs. 11 and 12.

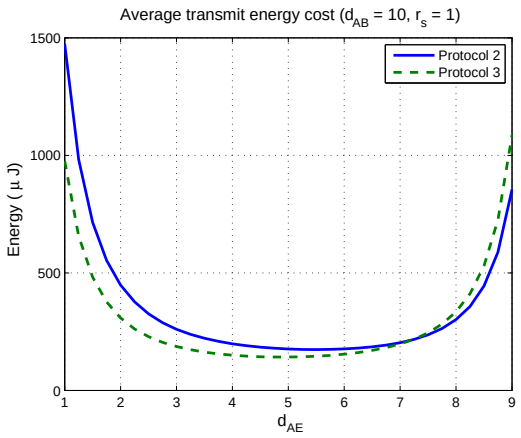

Fig. 12. Cost of key establishment using Protocols 2 and 3. The cost metric is the total transmit energy spent by A and B during the key exchange session. The values are calculated for the case described in Fig. 1. A and B are located with $d_{AB} = 10$ units apart, and E's location is varied on the line from $d_{AE} = 1$ to $d_{AE} = 9$(hence, $d_{BE} = 1$). For each value of d_{AE}, the expected cost is calculated. The probability values are found by the wireless comm. model using (2), by taking path-loss exponent $\alpha = 2$, $N_0 = -20$dBm. The transmit energy per unit time is taken as $E_A = E_B = \beta = 10$dBm for all nodes.

As can be seen, although they provide the same security, Protocols 2 and 3 are quite different both in their cost analysis and cost values. For example, consider $d_{AE} = 1$. Using Protocol 3 results in a transmit energy saving of 33%, i.e., by more than 1.5dB, which is a significant amount by wireless-communication engineering standards. The comparison of

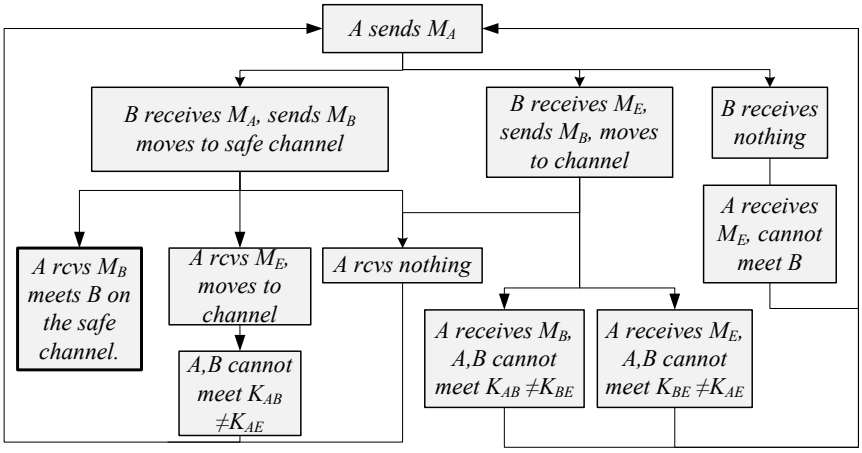

Fig. 10. The flow chart for the key exchange described in Fig. 7 for Protocols 3 and 4. The protocol is initiated by A sending M_A, which costs an (energy, delay) value of (N_A, D_A). B may receive this message, receive a message M_E from an attacker, or may not be able to decode the signal at all, each with a certain probability. In these two protocols implicitly-authenticated messages are exchanged, so the major difference compared to the flow chart in Fig. 8 is that B cannot immediately discard a message M_E, causing B to reply and switch to a wrong channel. This causes extra cost to the system compared to Protocols 1 and 2. The protocol is completed after a random series of retrials until A, B successfully receive each other's messages and start communicating over the channel $Ch(K_{AB})$.

Protocols 2 and 3 illustrates the problem of using message size as the metric to judge efficiency gains. For example, Protocol 3 has a total message size of 288 bits while Protocol 2 has a message size of 832 bits; however, because of the extra cost due to the lack of explicit authentication, the overall cost reduction with Protocol 3 compared to Protocol 2 is not necessarily 65% (as implied by ratio between the message sizes). In fact, for $d_{AE} = 9$, Protocol 2 requires *less* transmit energy on average, by roughly 20%. Aside from showing comparing the protocols, plots in Fig. 12 illustrate the fact that the attacker's location is a very important parameter affecting the total cost of key exchange over wireless channels. The cost grows exponentially when the attacker is very close to either of the nodes, which is natural since the success of an adversary's attacks become higher.

Fig. 12 plots the cost as a function of attacker location for some given r_s. However, the exact location of the attacker is in general unknown to the nodes. Hence, for evaluating a key exchange protocol, what may be more important is the maximum cost over all possible attacker locations outside the safe range, i.e. the worst-case cost. For example, for $r_s = 1$, for Protocol 2 in Fig. 12 the value for $d_{AE} = 1$ represents the worst case cost.

Figs. 13, 14, and 15 compare the worst-case transmit energy costs of key exchange for varying safe radius sizes. As expected, an increased safe range results in decreased cost.

Fig. 13 compares Protocols 2 and 3, for which the numbers for $r_s = 1$ on the plots are the maxima of the curves in Fig. 12. This shows that there is a benefit, in terms of reducing the worst case average energy, to use a protocol with implicitly-authenticated messages in place of one with explicitly-authenticated messages. Figs. 14 and 15 compare the cost for Protocols 1 and 2, and Protocols 3 and 4, respectively. Fig. 14 shows the benefit of switching from PKI-based to ID-based security keeping everything else (security level, protocol type) the same. Figure 15 shows that a major gain in cost reduction is obtained by switching to the SOK protocol, which sacrifices forward security. These figures

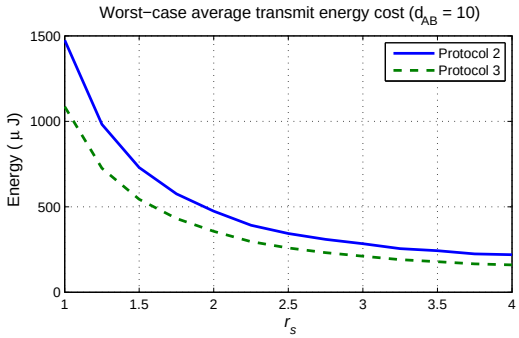

Fig. 13. Worst-case cost values plotted for Protocols 2 and 3. Cost metric is total transmit energy spent by nodes. For a given safe range size r_s, E can be located anywhere outside these ranges. Hence, the place where she causes maximum cost becomes the worst-case cost for that range. Hence for each value of r_s, the maxima of the cost as a function of E's location is calculated and plotted. For example, for $r_s = 1$, the value on the curves can be found by inspection by locating the maxima of the curves in Fig. 12.

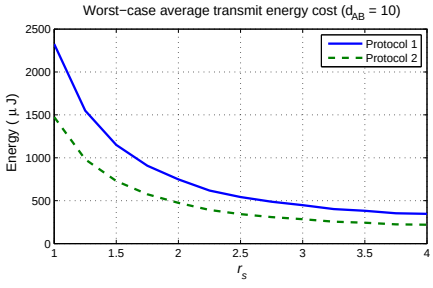

Fig. 14. Worst-case cost values plotted for Protocols 1 and 2. Calculation is the same as the case in Fig. 13.

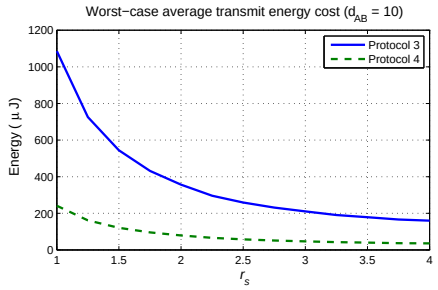

Fig. 15. Worst-case cost values plotted for Protocols 3 and 4. Calculation is the same as the case in Fig. 13.

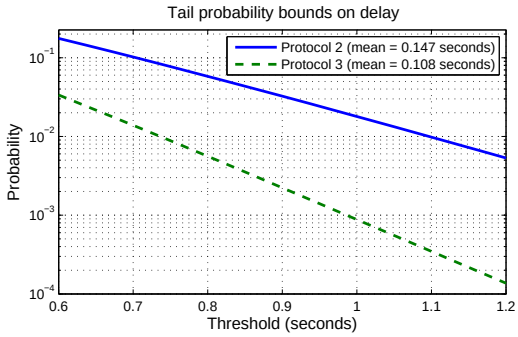

Fig. 16. Bounds on tail probabilities for Protocols 2 and 3 with Eve keeping silent if she senses no message M_B. For each r_s, we calculate the bound for the worst-case point of the expected value. We hasten to note that the curves should be considered individually as making a comparison of protocols based on bounds can be misleading.

compare cryptographic protocols according to metrics that are important from a wireless communication perspective, and provide a complementary viewpoint, which is necessary for system designers in fairly judging the security versus cost benefits of different protocols. *The overall combination of the changes from PKI-based to ID-based, then explicit to implicit, then sacrificing forward security, shows almost a 10-fold improvement in energy consumption*. Again we note that, lack of forward security is a concern only in the case of an attacker capable of the challenging task of recording the entire wideband channel. These results also concretely demonstrate the intuitive approach that achieving small message sizes is crucial to building secure, jamming-resistant key exchange protocols for wireless channels.

Finally, the plot for Protocol 2 in Fig. 12 is reproduced for E's transmit power increased from being equal to A and B's, to four times this value. As expected, the cost values increase with increased transmit power by attacker. However, the behavior of the cost as a function of the attacker's location is unchanged. The plots are omitted here due to space constraints. Note that, both safe range size r_s, and attacker's transmit power are valid parameters as a variable since they are a direct measure of the jamming power received at the nodes. However, safe range size r_s also effects the probability that an attacker successfully eavesdrops a message, which is important for cost evaluation in the scenarios considered in Section VI.

2) Delay: For delay, a natural statistic of more interest than the average delay is how likely it is for the key exchange to complete within a certain time duration. In other words, we are interested in the probability that the delay exceeds a certain threshold, which is commonly referred to as the tail probability. For numerical results, again we use the mgf's given in (11) and (12) and use them to find the bounds on the tail probability. Fig. 16 compares the bounds for Protocols 2 and 3.

VI. Physical Layer Augmented Key Exchange

In this section, we use insights from our cost analysis of traditional key exchange protocols to make modifications to the standard cryptographic protocols and improve how message exchanges are performed physically.

A. Protocol Modifications

One observation from the previous analysis is that the large cost arises because all of the messages need to be carried over the open air public channel which is subject to active attacks in which an adversary could force a high cost for sharing a secret key. Here, the modification we propose is based on the idea of reducing the number of bits that are carried over the open channel. We propose the idea of an *ephemeral channel*, which is a spread-spectrum channel temporarily established only for carrying the messages of a key exchange protocol.

The modified protocols are given in Fig. 17. In the first protocol, a message m containing a random number (less than 32 bits) for an ephemeral spread-spectrum channel, is appended to the first protocol message, so the message sent by A becomes (m, M_A). When B receives this message, he replies back on the channel generated by m, $Ch(m)$. Clearly, with this method only one of the messages is carried over the open channel. A second modification to the protocol is the same idea of sending a randomly generated ephemeral spreading code; however, this time as a single message over an open air channel. The protocol is described in Fig. 17 (right). A only sends the message containing the ephemeral spreading code over open air. When B receives this message and extracts the code, he replies with his message M_B over $Ch(m)$, and similarly A replies back with M_A on the same channel. In other words, the whole key exchange protocol is moved to a spread-spectrum channel. Note that, both these methods are *general* modifications that can be applied to any key exchange protocol in Table II.

These two modified protocols aim to carry at least some part of the messages on an ephemeral spread-spectrum channel to avoid signal jamming; however, this benefit comes at the expense that E could eavesdrop the open, short spreading code message and thus learn the channel $Ch(m)$, in which case she can successfully jam the message transfers and the benefit is lost. Clearly, the protocol may require an increased number of retransmissions for the message in Step 1, but the cost of these trials will be reasonable as the message containing the ephemeral channel code is typically short compared to the

messages exchanged by the key exchange protocols, hence the increase in message sizes is negligible. The total cost of key exchange is improved if the benefit obtained in subsequent steps will offset the added cost incurred for transmission of the ephemeral channel code.

The idea of sending a random code for establishing an ephemeral channel is motivated by the fact that the wireless channel quality is random and time-varying, and packet losses are inevitable for any receiver including an attacker.

A point to note is that the modifications proposed above do not significantly degrade the efficiency of the original method in the case when there are no attacks for that particular session. This is because the ephemeral code has a short length and will be delivered quickly under no attacks. Note that this provides a significant advantage versus [4].

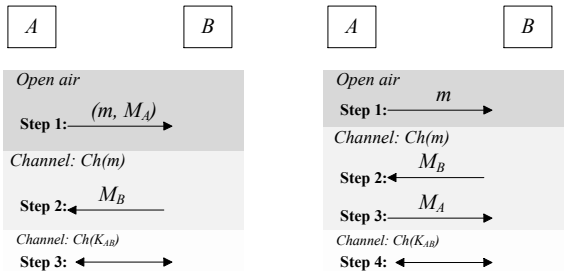

Fig. 17. Modified protocol (left): Sending a random ephemeral spreading code appended to the first message. A sends a randomly generated spreading code and appends it to message M_A. After receiving M_A and extracting the code, B sends M_B over channel $Ch(m)$ instead of open air as in the classic protocol in Fig. 7. Modified protocol (right): Sending a random ephemeral spreading code over the open air. A sends a randomly generated spreading code m and sends it. After receiving m, A, B exchange M_A, M_B over channel $Ch(m)$ instead of open air. In contrast to the classic protocol in Fig. 7, both protocol messages are carried over a spread-spectrum channel.

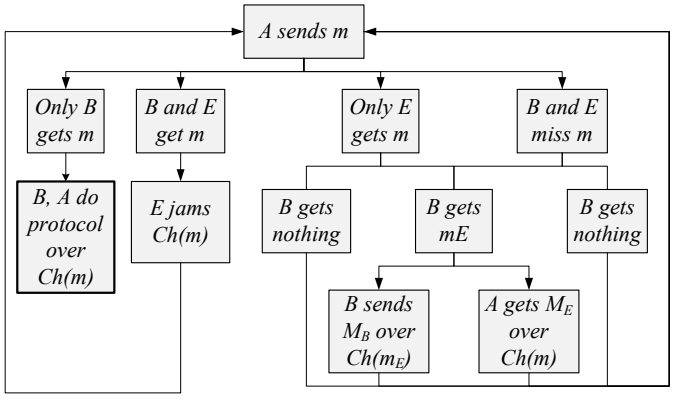

Fig. 18. The flow chart for the key exchange described in Fig. 17 (right) The protocol is initiated by A sending m. E may receive this message or miss it. B may receive m, a message m_E, or may not be able to decode the signal at all. Whenever E captures m she jams the communication between A and B given they were able to switch to $Ch(m)$. If the underlying protocol is Protocol 3 or 4, E can also incur cost by sending a message M_E to A which she replies back. When B receives m_E, he sends the protocol message M_B over a wrong channel $Ch(m_E)$. The protocol is completed after a random series of retrials until m is received by B and missed by E..

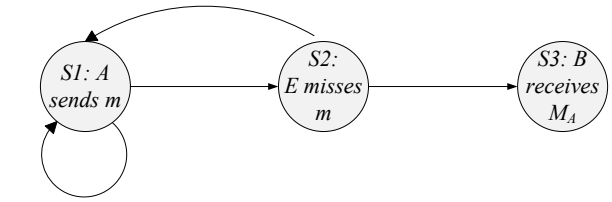

Fig. 19. Markov state diagram corresponding to the protocol described in Fig. 17 (right). The flow chart in Fig. 18 is simplified to a state diagram with three states.

B. Cooperative Jamming and Role-Switching

When the modified protocols with ephemeral spreading codes are employed, 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. Since Eve's location is unknown, the system is dominated by the concern of a near eavesdropper. We address this by utilizing a novel physical layer technique referred to as *cooperative jamming*, where a second antenna on the transmitter (or collaborator) 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.

Compared to the classic protocols, the performance of the modified protocols depends more strongly on which of the two parties initiates the protocol by sending the ephemeral code. Since nodes in general are not able to gauge their channel qualities to the other parties, we propose *role-switching* where nodes take turns initiating the protocol. For an example, consider the modified protocol in Fig. 17 (right). Suppose A sends the message m, and starts listening for a reply on $Ch(m)$. If she cannot receive a message, she reverts to listening for a message from B over open air. In the meanwhile, B has either missed the message m, or his reply over $Ch(m)$ has not been delivered and so he could not receive a message from A. In either case, B realizes this and switches roles, and becomes the sender of the ephemeral code, m, over open air.

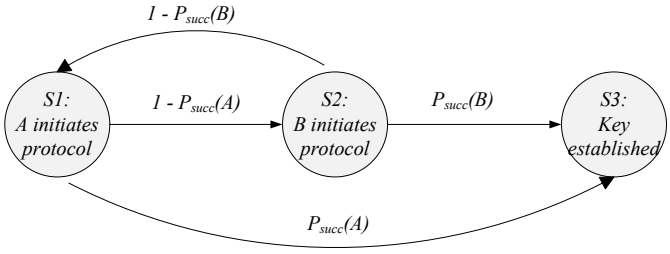

Fig. 20. Markov state diagram for cost evaluation of protocols with role-switching. A, B take turns in initiating the protocol after each failure. Hence, at each trial, either the protocol is successful, or the token changes hands. The probability and cost values for each branch will be adopted from the individual analyses for A and B using a signal flow graph such as in Figs. 9 and 11.

C. Cost Evaluation and Numerical Results

The flow chart for the modified protocol (Fig. 17 (right)) is given in Fig. 18. A very similar flow chart for the protocol in Fig. 17 (left) is omitted. The analysis is similar to previous sections and thus the basics omitted for brevity. However, one important difference here is the consideration of the role-switching described above, as the cost calculation becomes complicated when nodes take turns in initiating the protocol. However, this is easily handled by our analysis method. The Markov state diagram in Fig. 20 is used to find the overall cost with role-switching, where the values on each branch is calculated (not given in the figure) by simplifying the individual Markov diagrams for A and B for the specific protocol.

The expected transmit energy cost for the modified protocols is shown in Fig. 21. For each protocol, we have three plots showing the cost for *Classic*, *ephemeral appended* and *ephemeral on open air*. The plots for classic methods are the same plots from Fig. 15. As can be seen, the modified methods provide important performance gains compared to the classic protocols. These plots show that adding the ephemeral spreading code along with cooperative jamming and role-switching has a significant effect on performance. Hence, this demonstrates the value of integrating more traditional protocol designs with considerations of the underlying physical layer.

From Fig 21, it can be seen that, the modified methods offer better cost savings when the safe ranges are small. This is because a smaller safe range implies more severe jamming, for which the modifications are more suitable. The protocols with ephemeral over open air always requires less transmit energy on average compared to classic, although with savings reduced for less intense jamming. However, the method with ephemeral code appended starts to become more costly than classic after a certain value of r_s.

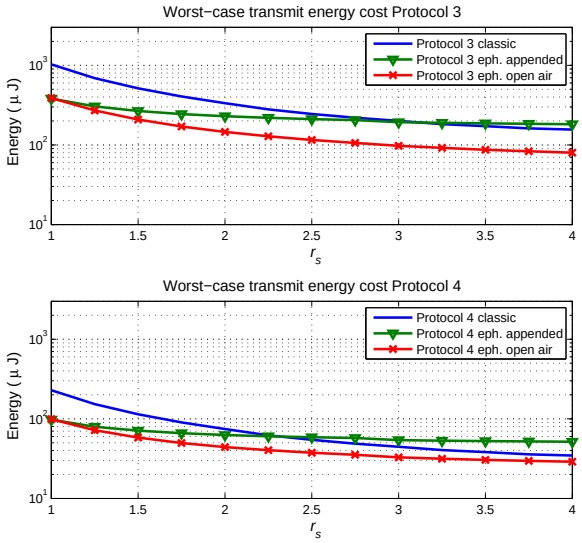

Fig. 21. Cost values for Protocols 3 and 4 are plotted including their modified counterparts. The cost metric is the total transmit energy spent by A and B. For each protocol, three curves are given. The first curve is the classic protocol, where the values are the same as in Fig. 15. The other two curves are for the modified protocols, where an ephemeral spreading code is added to the classic protocols to decrease the cost. The plots show that cost of key exchange over a wireless channel is reduced by modifying the protocols in a way to fit the physical layer properties of the wireless channel. Role-switching is assumed for calculating all of the curves. Cooperative jamming is employed only for the modified protocols. Plots for Protocols 1 and 2 show very similar behavior and are omitted here.

VII. Conclusions

In this paper, we have developed an analysis method in which cryptographic protocols are modeled as dynamic, probabilistic systems in order to assess their performance on wireless channels. We have illustrated the application of our method by analysing a range of traditional protocols for key establishment in a wireless environment with an active adversary. The analysis leads to several counter-intuitive results not suggested by prior approaches. In addition, this analysis led us to a novel approach to the design of key exchange protocols specifically tailored to the wireless environment. When studied using our analysis method, this approach exhibited performance enhancements over traditional key exchange protocols. This shows the value of adopting a design approach that integrates physical layer features with traditional key exchange primitives.

For reasons of space and clarity of presentation, we have focused on communication cost as the principal metric in this paper. It will be evident that our general approach can also be used to study computation costs, protocol execution times, or other metrics of practical relevance. At the same time, the method can also be applied to compute moments and tail probabilities for these metrics. Further, the application of the method is not limited to key exchange protocols, but could also be extended to study more complex classes of protocols, such as protocols for public key management, secure routing protocols for ad hoc networks, or protocols for secure distributed computing. We plan to explore these further applications in our future work. In addition, we plan to study how our approach can be extended to handle mobile nodes and adversaries. In this sense, we regard our work as providing a first step in bridging the gap between communications theory and security.

Appendix A
Physical Layer Models and Derivations

Messages sent through a wireless channel are subject to random signal fading due to multipath. Here, fading is assumed to be frequency-selective [24], where bands with sufficient frequency separation experience *independent* fading. 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* [25].

When we employ a slow frequency hopped (SFH) system, where the transmitter dwells in a given frequency band for the transmission of a number of bits before hopping to a different band, the appropriate model is that of a narrowband system, and the signal for a given packet is multiplied by

a single fading factor. 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 A on a given hop of an SFH system. The received symbols at B, $y_{B,i}$ is

$$y_{B,i} \quad = \quad h_{AB}\sqrt{\frac{E_A}{d_{AB}^{\alpha}}}x_{A,i} + n_{B,i}, \quad i = 1, 2, \cdots, M \quad (13)$$

Here, E_A is the symbol energy as transmitted by A, 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 packet. 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.

$$\text{SINR}_B = E_{rcv}^{(B \leftarrow A)}/N_0, \quad \text{where} \quad (14)$$

$$E_{rcv}^{(B \leftarrow A)} = E_A/d_{AB}^{\alpha}|h_{AB}|^2 \sim \text{Exp}(\lambda_{BA})$$

So, the received energy is an exponential random variable with parameter $\lambda_{BA} = d_{AB}^{\alpha}/E_A$. A packet is lost if the received signal-to-noise-and-interference ratio (SINR) is below a certain threshold, γ.

$$P_{\text{rcv}}^{(B \leftarrow A)} \quad = \quad P(\text{SINR}_B > \gamma) = e^{-\lambda_{BA}\gamma N_0} \quad (15)$$

which yields (2)

The SINR threshold, γ, is determined by the physical layer error control coding rate R bits/symbol. The frequency band allocated and pulse shaping dictate the number of symbols that can be aired per second. Hence, R is directly proportional to the rate in the more familiar unit of bits/s. Information theoretical results show that the relation between R and γ is given by $R = \log_2(1 + \gamma)$.

In the case of a jamming attacker Eve on the narrowband channel, the SINR at B becomes

$$\text{SINR}_B = E_{rcv}^{(B \leftarrow A)}/(N_0 + E_{rcv}^{(B \leftarrow E)}) \quad \text{where} \quad (16)$$

$$E_{rcv}^{(B \leftarrow E)} = \beta/d_{BE}^{\alpha}|h_{BE}|^2 \sim \text{Exp}(\lambda_{BE}) \quad (17)$$

So, the jamming power suffered at B is also an exponential random variable with parameter $\lambda_{BE} = d_{BE}^{\alpha}/\beta$, and statistically independent of $E_{rcv}^{(B \leftarrow A)}$. Then,

$$P_{\text{rcv}}^{(B \leftarrow A)}|\text{Jamming} \quad = \quad P(\text{SINR}_B > \gamma) = \frac{e^{-\lambda_{BA}\gamma N_0}}{1 + \gamma\frac{\lambda_{BA}}{\lambda_{BE}}} \quad (18)$$

Hence, the probability under jamming is as in (3).

For a fast frequency hopping (FFH) system [1], where the sender hops multiple times during the transmission of a single symbol, the appropriate model is that of a wideband system. 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 B will be

$$\frac{1}{K}\Big(\sum_{k=1}^{K}|h_{AB}^{(k)}|^2\Big)\frac{E_A}{d_{AB}^{\alpha}}, \quad \text{with} \quad (19)$$

$$P_{\text{rcv}}^{(A \to B)} \quad = \quad P\left(\frac{1}{K}\Big(\sum_{k=1}^{K}|h_{AB}^{(k)}|^2\Big)\frac{E_A/N_0}{d_{AB}^{\alpha}} > \gamma\right). \quad (20)$$

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 [25]. In other words, it is reasonable to assume probabilities as given in (4) for an FFH system.

REFERENCES

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

[2] C. J. Mitchell and F. Piper, "Key storage in secure networks," *Discrete Applied Mathematics*, vol. 21, no. 3, pp. 215–228, 1988.

[3] L. Eschenauer and V. D. Gligor, "A key-management scheme for distributed sensor networks," in *ACM Conference on Computer and Communications Security*, V. Atluri, Ed. ACM, 2002, pp. 41–47.

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

[5] 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*. New York, NY, USA: ACM, 2009, pp. 219–228.

[6] 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*, ser. CCS '07. New York, NY, USA: ACM, 2007, pp. 401–410. [Online]. Available: http://doi.acm.org/10.1145/1315245.1315295

[7] 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*. New York, NY, USA: ACM, 2009, pp. 321–332.

[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*. New York, NY, USA: ACM, 2008, pp. 128–139.

[9] M. A. Zafer, D. Agrawal, and M. Srivatsa, "A note on information-theoretic secret key exchange over wireless channels," in *Communication, Control, and Computing, 2009. Allerton 2009. 47th Annual Allerton Conference on*, 30 2009-Oct. 2 2009, pp. 754–761.

[10] R. Sakai, K. Ohgishi, and M. Kasahara., "Cryptosystems based on pairing." in *The 2000 Symposium on Cryptography and Information Security*, Okinawa, Japan, 2000, pp. 26–28.

[11] J. Cavers, *Mobile Channel Characteristics*, 1st ed. Springer, Sep. 2000.

[12] R. Howard, *Dynamic Probabilistic Systems*. John Wiley & Sons Inc, Jun. 1971.

[13] A. Menezes, P. C. van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*. CRC Press, 1996.

[14] M. Bellare and P. Rogaway, "Entity authentication and key distribution," in *CRYPTO 1993*, ser. Lecture Notes in Computer Science, D. R. Stinson, Ed., vol. 773. Springer, 1993, pp. 232–249.

[15] R. Canetti and H. Krawczyk, "Analysis of key-exchange protocols and their use for building secure channels," in *EUROCRYPT 2001*, ser. Lecture Notes in Computer Science, B. Pfitzmann, Ed., vol. 2045. Springer, 2001, pp. 453–474.

[16] M. Bellare, D. Pointcheval, and P. Rogaway, "Authenticated key exchange secure against dictionary attacks," in *EUROCRYPT 2000*, ser. Lecture Notes in Computer Science, B. Preneel, Ed., vol. 1807. Springer, 2000, pp. 139–155.

[17] D. Boneh, B. Lynn, and H. Shacham, "Short signatures from the weil pairing," in *ASIACRYPT 2001*, ser. Lecture Notes in Computer Science, C. Boyd, Ed., vol. 2248. Springer, 2001, pp. 514–532.

[18] P. S. L. M. Barreto and M. Naehrig, "Pairing-friendly elliptic curves of prime order," in *Selected Areas in Cryptography 2005*, ser. Lecture Notes in Computer Science, B. Preneel and S. E. Tavares, Eds., vol. 3897. Springer, 2005, pp. 319–331.

[19] L. Chen, Z. Cheng, and N. P. Smart, "Identity-based key agreement protocols from pairings," *Int. J. Inf. Sec.*, vol. 6, no. 4, pp. 213–241, 2007.

[20] P. S. L. M. Barreto, B. Libert, N. McCullagh, and J.-J. Quisquater, "Efficient and provably-secure identity-based signatures and signcryption from bilinear maps," in *ASIACRYPT 2005*, ser. Lecture Notes in Computer Science, B. K. Roy, Ed., vol. 3788. Springer, 2005, pp. 515–532.

[21] L. Chen and C. Kudla, "Identity based authenticated key agreement protocols from pairings," in *CSFW*. IEEE Computer Society, 2003, pp. 219–233.

[22] R. Dupont and A. Enge, "Provably secure non-interactive key distribution based on pairings," *Discrete Applied Mathematics*, vol. 154, no. 2, pp. 270–276, 2006.

[23] K. G. Paterson and S. Srinivasan, "On the relations between non-interactive key distribution, identity-based encryption and trapdoor discrete log groups," *Des. Codes Cryptography*, vol. 52, no. 2, pp. 219–241, 2009.

[24] J. Proakis, *Digital Communications*, 4th ed. McGraw-Hill Science/Engineering/Math, Aug. 2000.

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