Mobility Assisted Secret Key Generation Using Wireless Link Signatures

Junxing Zhang, Sneha K. Kasera
Email: {junxing, kasera}@cs.utah.edu

Neal Patwari
Email: npatwari@ece.utah.edu

***Abstract*—We propose an approach where wireless devices, interested in establishing a secret key, sample the channel impulse response (CIR) space in a physical area to collect and combine uncorrelated CIR measurements to generate the secret key. We study the impact of mobility patterns in obtaining uncorrelated measurements. Using extensive measurements in both indoor and outdoor settings, we find that (i) when movement step size is larger than one foot the measured CIRs are mostly uncorrelated, and (ii) more diffusion in the mobility results in less correlation in the measured CIRs. We develop efficient mechanisms to encode CIRs and reconcile the differences in the bits extracted between the two devices. Our results show that our scheme generates very high entropy secret bits and that too at a high bit rate. The secret bits, that we generate using our approach, also pass the 8 randomness tests of the NIST test suite.**

I. Introduction

Growing work shows physical layer characteristics of wireless links such as multipath properties are different at different locations, and can be considered to be signatures of wireless links. For example, Fig. 1 shows two channel impulse response(CIR) [1], [2] magnitude measurements taken at different locations. From the figure, the two measurements are quite different due to varied channel conditions at the two locations. The fact that these link signatures can be measured almost symmetrically between two ends of a wireless link [3], but cannot be measured from another distinct location has led researchers to suggest using these for secret key establishment [4], [5], [6]. Secret key extraction from link characteristics has the potential to provide an inexpensive alternative to quantum cryptography [7], [8].

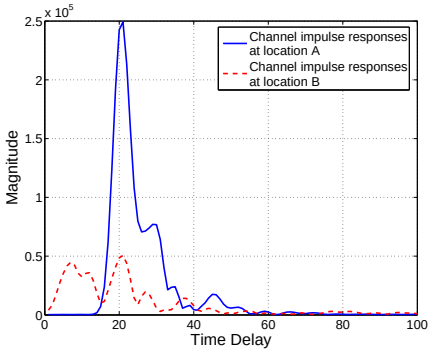


Fig. 1. Multipath properties at two locations.

However, an adversary can be at one of the genuine wireless endpoints' locations and measure the same link signature. Once the adversary steals some signature measurements it has a good chance to determine the key generated with the link signature measurements. We call the attack launched in this manner the *location locking attack*. To avoid this problem, existing work [5] has relied upon the movement in the environment or the movement of the devices exchanging the keys to perturb the wireless channel in an unpredictable manner. Such unpredictable channel is expected to produce unpredictable secret keys. Very interestingly, Jana et al. [9] showed that in static scenarios, an adversary can actually cause predictable movement in the environment and thus fool the endpoints to extract *deterministic* secret keys that it can extract itself. Alternatively, instead of depending on movement in the environment, devices can themselves move to cause variations in the wireless channel that gets translated into secret keys. However, the device must continue to move during key extraction. In this paper, we propose a different approach. Instead of extracting keys from the temporal variations in the channel, the wireless devices measure the wireless link signatures at different unpredictable locations and combine these measurements to produce strong secret keys. We use the CIR as our wireless link signature[1]. Essentially, in our approach, the wireless devices sample the *CIR space* in a physical area to collect uncorrelated CIR measurements.

To understand our approach at a high level, consider two devices Alice (A) and Bob (B), as shown in Figure 2. Assume that these devices are mobile and are at different locations at different times. Let X_i be the CIR measurement of the link between A and B when they are at any pair (denoted i) of specific locations. Let X_i be measured accurately only by devices A and B and no other device that is not at the location of A or B or very close by. The two devices, A and B, measure the CIR at different location pairs. Both A and B use a previously agreed upon and publicly known function f of these measurements, $f(X_1, X_2, X_3, \ldots X_n)$, to compute the shared key. An important assumption here is that an adversary can at best be at some of these locations where the CIR is measured but not all and hence will not be able to compute the secret key if n is reasonably large. As long as the movement of Alice and Bob is not fully retraceable, the change of location does not need to happen at short time intervals. Note that by using the samples in the CIR space, we do not preclude

[1]In this paper, our CIR is actually a vector of 25 channel impulse responses measured over time.

the benefits of channel variations caused by movement in the physical environment or that of A and B. Our novel use of spatial sampling will significantly strengthen any existing technique [5], [10] and make them more robust.

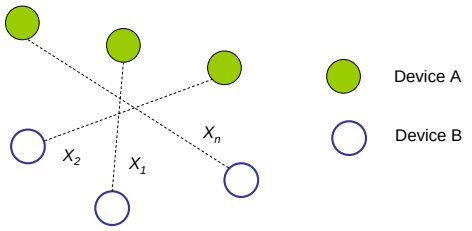


Fig. 2. Wireless devices A and B are shown to be measuring X_is at different locations.

One of the important requirements for generating strong keys is to pick X_is that are uncorrelated with each other. However, the correlation among X_is depends on the multipath characteristics of the physical environment, the step size of movement, and in general the mobility model. We investigate three mobility models - random walk, Brownian motion, and Levy walk, in this paper. We also develop efficient mechanisms to encode CIRs and reconcile the differences in the bits extracted between A and B. Specifically, we propose a new Jigsaw encoding scheme that keeps the mismatch rate in reciprocal measurements, at A and B, low even when CIRs are quantized with increasing bit numbers. We adopt Reed-Solomon forward error correction to reconcile the bits that do not match at A and B, and also analyze the computational complexity of this process. Using extensive measurements in both indoor and outdoor settings, we find that when movement step size is larger than one foot the measured link signatures are mostly uncorrelated. When using step sizes drawn from continuous uniform, Levy, and Gaussian distributions in the adopted three mobility models, we find more diffusion in the model results in less correlation in the measured link signatures. We also find that our scheme generates very high entropy secret bits and that too at a high bit rate. The secret bits, that we generate using our approach, also pass the 8 randomness tests of the NIST test suite [11] that we conduct.

The rest of this paper is organized as follows. In the next section, we provide estimations of the size of the channel impulse secret key space. In Section III we define the adversary model. In Section IV we present a mobility assisted secret key establishment protocol and propose a new encoding scheme and discrepancy reconciliation method that work with the protocol. Finally, in Section V, we evaluate the protocol and examine the impact of mobility and different mobility models on signature randomness and key extraction. The related work are given in Section VI and conclusions are drawn in Section VII.

II. Channel Impulse Response Secret Space

Before developing our sampling and key extraction strategies, we first obtain an estimate of the size of CIR secret space. This number tell us how many unique channel CIR measurements are possible in a given physical environment. For obtaining this number, we use the mutual information between the CIR measurement and the location where the measurement is taken. The details of the methodology to obtain the mutual information from measurement data are described in [12] and not a contribution of this paper. We apply the methodology to obtain mutual information to three data sets that we collect. These data sets essentially contain the CIR measurements and the location information in three different environments. Our measurement campaign for obtaining these data sets will be described in Section V. Here, we only present the estimates of the CIR secret space for the three environments.

Dataset 1	34
Dataset 2	29
Dataset 3	27

TABLE I
Size of signature space in different environment

Table I shows the CIR secret space estimates for the three data sets in bits. An estimated size of 34 bits means that 2^{34} unique CIRs can be obtained from this physical environment. Now, 34 bits by itself is not a very large number and most computers can very quickly try out all the 2^{34} possibilities. However, when we use multiple 34 bit values in the function f, we can obtain much longer keys. For example, if we organize the 20 uncorrelated measurements (ideally, these should be 20 independent measurements) and permute them[2], we will increase the shared secret size by bits equivalent to $20!$. This is because to break the secret an attacker will have to try $20!$ permutations in the worst case. Using Stirling's approximation [13], $20!$ corresponds to about 61 bits thus increasing the shared secret space to roughly 61+34 = 95 bits. We can increase the size even more by increasing n.

III. Adversary Model

We consider an adversary that can overhear all the communication between the two devices A and B. Our adversary can also be in some of the locations where the transmitter or the receiver has been in the past or will be in the future, but the adversary does not know or cannot access all the locations visited by the transmitter and the receiver. We assume that the adversary cannot cause a person-in-the-middle attack. Essentially, we do not address the issue of the authentication of the endpoints (A and/or B) in this paper. We expect our secret key extraction scheme to be used in conjunction with some of the fingerprinting-based authentication being developed elsewhere (e.g., [14], [15], [9]).

Our adversary is also not interested in causing any Denial-of-Service attacks. Some existing approaches propose to secure or hide the location where signature measurements are taken. For example, in [6] the authors suggest to define a

[2]This means that $f(X_1, X_2, X_3, \ldots X_n) = X_1 \| X_2 \| X_3 \| \ldots \| X_n$.

threat region about the legitimate user and physically guarantee that an attacker is not within this region. We do not make any such assumptions.

IV. Mobility Assisted Key Establishment

In this section, we first describe the our simple secret key establishment protocol between A and B that counters the threat that may arise from the adversary described in Section III. Next, we present the important building blocks of the protocol including CIR quantization, bit extraction, and encoding. Finally, we explain how to use Reed-Solomon (RS) error-correcting code to reconcile bit mismatch between the two communicating parties and ensure information (key bits) confidentiality at the same time.

A. Key Establishment Protocol

Our key establishment protocol between A and B is divided into three phases. Fig. 3 shows the message exchange of the protocol. In the first phase, called SIGGEN (short for signature generation), A and B exchange SIGGEN and SIGACK messages to allow them to measure a sufficient number of reciprocal CIR. Note that due to hardware differences and the differences in time instances at which the channel measurement is performed at A and B, the measured CIR is not perfectly reciprocal. We will address this imperfect reciprocity below. Between each pair of SIGGEN and SIGACK message exchange, A and B individually, or both move to a new location.

In the second SIGCHK (short for signature check) phase, upon receiving the SIGCHK message from A, B quantizes all CIR it has measured and removes any duplicates. He then encodes the remaining quantized CIRs to produce both message symbols and parity symbols. Next, in the SIGFEC (short for signature forward error correction) phase, B sends only the parity symbols to A in multiple SIGFEC messages. Upon receiving all the SIGFEC messages, A quantizes the corresponding CIRs that she had measured and encodes them to produce message symbols. B informs A about which CIR measurements to use. This is done with the help of sequence numbers. A then combines her message symbols with parity symbols she receives from B to obtain a bit stream that is identical to that of B. In the final KEYGEN (short for key generation) phase, A and B generate a new secret key with the reconciled bit streams and verify that they indeed have the same key through a simple challenge response exchange.

We describe the mechanisms to quantize link signatures and extract bits and to perform encoding and error correction in the following subsections. The estimate of the CIR secret space (as in Section II) and the size of the required secret key determine how many CIR measurements are necessary for secret key extraction. To convert the bit stream obtained from the CIR measurements, we utilize a key compression function. This compression function uses the 2-universal hash family to perform *Privacy Amplification* [16]. Privacy amplification minimizes the possible correlation among input bits of the bit stream and compresses the raw bits to the chosen key size with

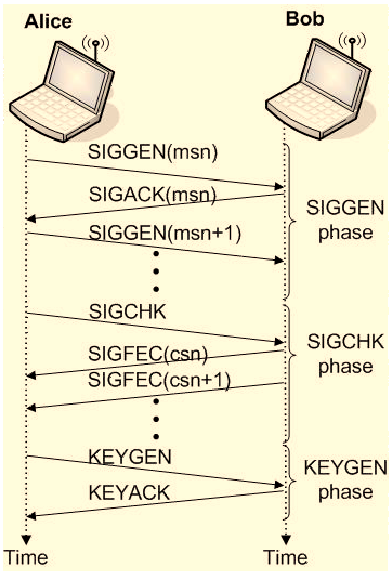


Fig. 3. Mobility Assisted Key Establishment. The sequence numbers used by different messages are denoted as msn and csn.

a target function. We use SHA-256, SHA-384, and SHA-512 as the target function to produce keys of 256, 384, and 512 bits. For a given fading environment and protocol configuration, not every measured CIR can be reconciled, so our protocol cannot ensure successful secret key establishment all the time and hence is opportunistic. We use a Bloom filter[3] [17] for finding duplicates among quantized CIRs in an efficient manner in the SIGCHK phase above.

B. Quantization and Bit Extraction

Because CIRs are continuous random variables, we must quantize them in order to use them for secret key generation. In this paper, we adopt the widely-used uniform quantization [18] to quantize CIR measurements. In order to quantize CIR into integer vectors that can be easily converted to binary bits, we first normalize each CIR with its maximum element value. Channel impulse response normalization is also avoids the impact of the intentional manipulation of the transmitting power by an attacker or to filter out the effect of the slow temporal changes in the average signal power. Next, to quantize the normalized CIR to 2^q discrete values with equal intervals, we multiply these values with 2^q and then round them to the nearest integers in the range of $[0, 2^q - 1]$. We simply convert integers in the resulting vector to their binary representation to extract the initial bits that we use later for secret key generation.

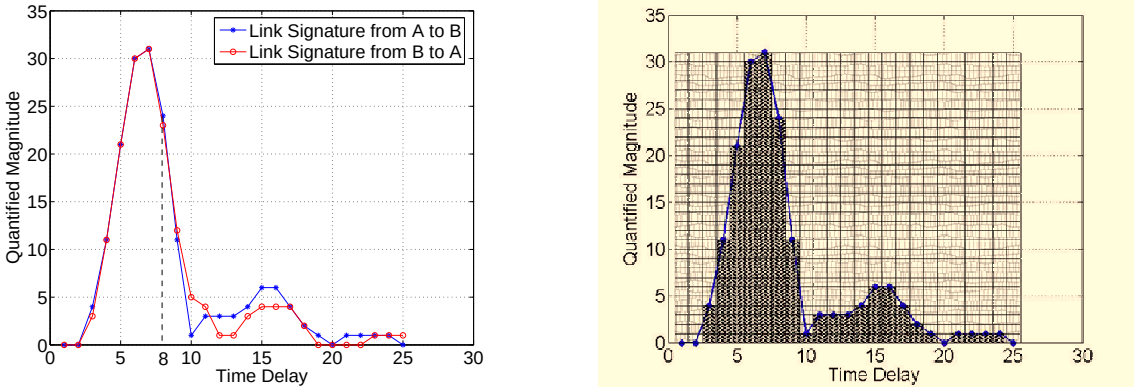
C. Jigsaw Encoding

Although uniform quantization is simple and easy to implement, we find when increasing the quantization bit number q from 1 to 8, the rate of the discrepant elements in the quantized CIRs, that are not measured the same at A and

[3]A Bloom filter is a space-efficient probabilistic data structure that is often used to test memberships in a set.

Quan Bits	1	2	3	4	5	6	7	8
UniQuan Mean	0.0064	0.0908	0.1952	0.3660	0.4824	0.6476	0.7560	0.8448
JigEnc Mean	0.0032	0.0232	0.0307	0.0374	0.0372	0.0379	0.0380	0.0380
JigEnc Std	0.0123	0.0196	0.0236	0.0228	0.0231	0.0230	0.0231	0.0232

TABLE II
DISCREPANCY RATE IN RECIPROCAL LINK SIGNATURE MEASUREMENTS



(a) Uniform Quantization (b) Jigsaw Encoding

Fig. 4. Comparison of the uniform quantization and the jigsaw encoding, showing (a) a pair of reciprocal link signatures that are uniformly quantized, and (b) one signature of the pair that is uniformly quantized and then encoded with the Jigsaw scheme. The dark colored patterns represent random numbers from the one-map and the light colored patterns stand for numbers from the zero-map.

B, grows dramatically. Table II lists the discrepancy rate in a sample bi-directional measurement set. The row "UniQuan Mean" shows the rates with only uniform quantization. In this row the rate increases from 0.0064 to 0.8448. Even quantized with only 3 bits, there are 19.52% of elements in each pair of reciprocal CIRs that do not agree with each other. Because reciprocal measurements should be very similar, these results suggest that the simple uniform quantization cannot preserve reciprocity and even increase the discrepancy rate in quantized CIRs.

Fig. 4(a) shows a pair of reciprocal link signatures that are uniformly quantized. It appears the two link signatures are very similar, whereas they have 14 out of 25 elements (56%) that do not agree. The high discrepancy rate results from the fact that each element is represented only by a single quantized value. For example, the elements at the delay 8 are 23 and 24. They agree on the first 23 units and differ only at the last unit. Because they are represented with the sum of these units, their similarity is hidden. To solve this problem, we propose to further encode each uniformly quantized value with multiple values. We call the new encoding scheme *Jigsaw Encoding*. In this scheme, we make use of two random number maps that are shared between the two parties. Each map is a matrix of 2^q rows and L columns where q is the quantization bit number and L is the link signature length. The matrix elements are random numbers of m bits with the first bit as the sign bit. All random numbers in the first map has the sign bit of one so it is called one-map. All random numbers in the second map has the sign bit of zero so it is called zero-map. We patch the two maps together to form a jigsaw map. We define the joint points of the jigsaw map by quantized element values of the CIR. For example, the A-to-B link signature in Fig. 4(a) is encoded with the new method in Fig. 4(b). Because the element at the delay 8 is 24 in this signature, the method encodes this value in a column of random numbers with the first 24 numbers chosen from the one-map (depicted with dark colored patterns) and the last 7 numbers (since signature elements are quantized in $[0, 31]$) from the zero-map (depicted with light colored patterns). The numbers are chosen according to their positions in the 8^{th} columns of the two maps. If we apply the same method to the B-to-A signature in Fig. 4(a), at the time delay 8 they will have 30 numbers agree (23 from the one-map and 7 from the zero-map) and only 1 number discrepant. Therefore, the additional Jigsaw encoding helps to expose reciprocal similarity greatly. The big improvement is illustrated by the discrepancy rates at the second row of Table II. This improvement is achieved by replacing a single quantized element value with an array of random symbols. The original value is encoded as the partitioning index that divides the array symbols into groups from the two maps. We utilize random symbols instead of some constant value for two reasons. First, we will encode random numbers in the Jigsaw map further in the RS scheme described below which has a requirement of the number of bits per symbol. Second, because the RS scheme treats input symbols as the coefficients of a polynomial to generate output symbols, if all input symbols only have two constant values, they would compromise the error-correction strength of the scheme.

D. RS Error Correction

We adopt the RS forward error correction (FEC) scheme [19] to reconcile any discrepancies in reciprocal mea-

surements of CIR. FEC works by sending redundant data in addition to messages so that the receiver can detect and correct errors within some bound. Our adoption of FEC is a little different from its conventional use. We send only redundant data (parity symbols) because the other party already has its own encoded CIR, although with some errors.

The RS code works by determining a degree $k-1$ polynomial and treating every k symbols of the input message as the coefficients of the polynomial. It encodes the coefficients by evaluating the polynomial at various points. Each output codeword has p symbols including k input symbols followed by $2 \times t$ parity symbols. t is the error-correction capability of the (p,k) RS code. This relation is described in Eq. 1[4]. In addition, the codeword length p is determined by the symbol bit-number m as shown in Eq. 2.

$$t = \frac{p-k}{2} \quad (1)$$

$$p = 2^m - 1 \quad (2)$$

$$\epsilon = \frac{t}{k} \quad (3)$$

In our adoption of the RS code, the sender transmits the parity symbols of codewords in the link layer payload. Because link layer error control (i.e. retransmission of erroneous packets) is provided in wireless networks, we assume these symbols are always received correctly at the other party. Therefore, the inconsistent symbols can only appear in the input message portion (the reciprocal signature) of the codewords. This situation leads to the definition of the link signature discrepancy rate ϵ in Eq. 3.

$$t = \frac{\epsilon \times (2^m - 1)}{1 + 2 \times \epsilon} \quad (4)$$

$$\Gamma = 2^{q \times \left\lceil \frac{k-t}{2^q} \right\rceil} \approx 2^{q \times \frac{1-\epsilon}{2 \times \epsilon + 1} \times 2^{m-q}} \quad (5)$$

Note that the use of FEC also has security implications. There are predominantly two concerns with the use of FEC. First, it might be possible for a third party to use the parity symbols and correct its own CIR to match the actual CIR between A and B. Second, the parity symbols might themselves give away information on the actual bits of the CIR. We address the first problem by constraining the error-correction capability t, for a given symbol size m, to limit the reciprocal rate to ϵ, as governed by Eq. 4. If the measured CIR of the attacker has more errors than the reciprocal discrepancy ϵ, the public parity symbols will not be able to turn the attacker's CIR into the legitimate CIR. To address the second problem, we make the discovery of the coded link signature using brute force and public parity symbols computationally infeasible. The reciprocal CIRs can have up to t inconsistent symbols among a total of k symbols. This means that the two parties share at least $k-t$ message symbols in order to convert one signature into the other. Without this shared information, an attacker will have to find the joint points of $\left\lceil \frac{k-t}{2^q} \right\rceil$ columns (q is the quantization bit number) in the jigsaw map to obtain $k-t$ correct symbols. Given 2^q possible values each joint point can take, to have all correct joint points at the same time, the computational complexity Γ would be as large as defined in Eq. 5. For $m = 10$ and $q = 5$, it is larger than 2^{133}. For $m = 10$ and $q = 1, 2$, it is in the order of 2^{427}. It should be noted that due to the exponential decrease of $\frac{1}{2^q}$ the complexity drops very fast with larger quantization bit numbers, and because its maximum value is 0.5, symbols must have more than 8 bits to obtain a complexity larger than 2^{128}.

[4]For the convenience of analytical reduction, we ignore the requirement for k to be an odd integer here. The approximation can cause k off by a value smaller than 2.

V. Protocol Evaluation

In this section, we evaluate the proposed protocol in three steps. First, we assess the impact of device mobility on measured link signatures. Then, we investigate its impact on key generation. Finally, we evaluate the quality of the generated keys.

A. Mobility Models

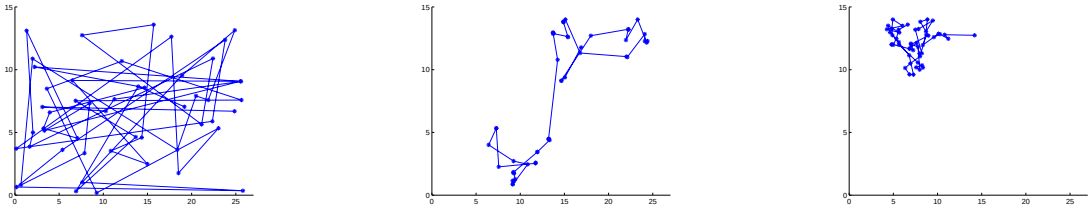
To study the impact of device mobility, we use three mobility models. These three models are among the most popular mobility models used for evaluating wireless research. Fig. 5 shows sample trails of these three models. Fig. 5(a) exhibits a trail of the *Random Walk* model [20]. It is created by starting from a random location in the specified area, walking toward another randomly selected location in the area in each step, and repeating. Fig. 5(b) displays a trajectory of the *Levy Walk* model [21]. This model is also created with a sequence of steps. Each step is defined by a uniformly distributed direction and a step size drawn from a Levy distribution as defined in the following Eq. 6. The exponent $\alpha = 0.8$ in this example.

$$f_{X_i}(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{-itx - |ct|^{\alpha}} dt \quad (6)$$

When $\alpha = 2$ in the Eq. 6 the Levy distribution reduces to a Gaussian distribution. Using this distribution for the step size, we can create a *Brownian Motion* model as demonstrated by Fig. 5(c). This model depicts the random movement of particles suspended in a liquid or gas. It can be seen from Fig. 5 that the three mobility models demonstrate decreasing diffusion, so they form a valuable suite to study the impact of device mobility. Although researchers have established the similarity between the Levy Walk pattern and the outdoor human mobility, it will be valuable to study the other two simpler models. Moreover, it is also known the Levy Walk model does not account for social constraints and geographic restrictions [21].

B. Measurement Campaign

We collect five sets of measurements for this study. The first two sets are collected at multiple discrete locations along the trails generated by the three mobility models. We take the first set of measurements in a large lobby of an engineering building on the University of Utah campus (*Indoor Trail* set). We collect the second set of measurements on a flat square



(a) Random Walk (b) Levy Walk (c) Brownian Motion

Fig. 5. Three Mobility Models

outside of the building (*Outdoor Trail* set). We collect the next two sets of measurements on two grids of different scales. The first grid measures 30 by 30 foot with a grid line distance of 1 foot, and the second grid is 14 by 26 inch with a grid line distance of 2 inch. Measurements are taken at every cross point of the grids. We refer to measurements in the first set as *Grid* measurements and those in the second set as *Fine Grid* measurements. The last set of measurements is taken at one fixed indoor location (*Stationary* set).

We obtain all measurements with a Direct Sequence Spread Spectrum (DS-SS) transmitter and receiver. The transmitter sends a DS-SS signal with a 40 MHz chip rate, a 1024 code length, and a center frequency of 2443 MHz. These signals are essentially unmodulated pseudo noise which span the US ISM band. The receiver correlates the received signal with he known DS-SS signal to estimate the CIR. We rely on trail measurements to accurately reflect the relationships between measurements at the various points in space determined by the mobility models. Since we can sample only a limited number of trails, it is possible that some features disclosed by trail measurements are not universal to the movement pattern. Therefore, we always use grid measurements to verify results obtained from trail measurements. Because grid measurements cover grid areas with limited precision, we can generate numerous 'quantized' trails for testing.

C. Impact of Mobility on Link Signatures

In this subsection, we investigate how various mobility models affect correlation among measured CIRs. For this purpose, we use the correlation coefficient between pairs of CIRs, X_i and X_j, defined as

$$\rho_{X_i,X_j} = \frac{E((X_i - \mu_X)(X_j - \mu_X))}{\sigma_X^2}$$

However, instead of using just a single value given by the above equation, we plot the histogram of all the values $(X_i - \mu_X)(X_j - \mu_X)/\sigma_X^2$ to study their distribution.

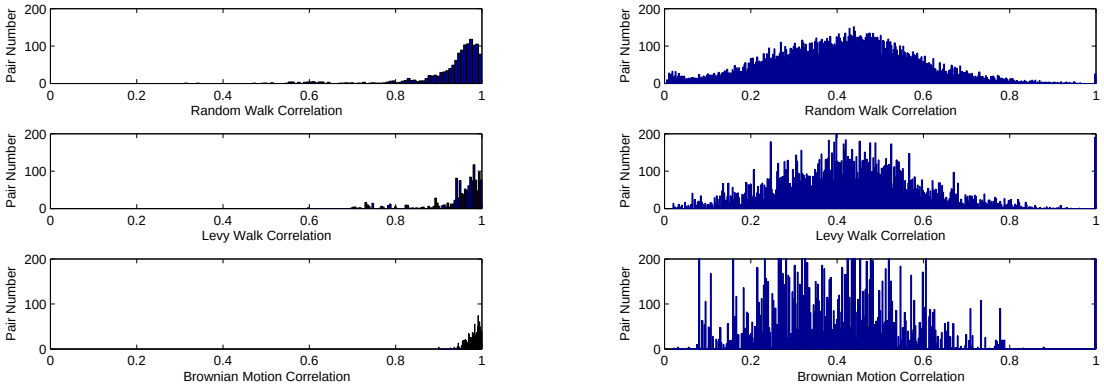
Fig. 6 and Fig. 7 show that the more diffusive a mobility model is the less correlated its measurements become. In Fig. 6(a), the correlation among the CIRs of the Random Walk model lie in [0.6, 1], while that of the Levy Walk model and the Brownian Motion model lie in [0.7, 1] and [0.9,1], respectively. Surprisingly, Fig. 6(b) does not show the similar trend in correlation. In comparison to the two subfigures of Fig. 6, we can notice the striking difference in their correlation distributions. The correlation values of the Fine Grid measurements predominantly concentrate to the right of 0.6, whereas the correlation values of the Grid measurements are mostly centered at 0.4. This disparity suggests most Fine Grid measurements are correlated but most Grid measurements are not. The lack of a trend in Fig. 6(b) can be explained by measurements occurring at coarsely-quantized mobile trail points, rather than the actual trail points.

The distance between measurement points in the Grid set is 1 foot. We suspect that this coarse grid quantization has "homogenized" the three mobility models, and thus the correlation distributions in Fig. 6(b) does not show any differences in the three mobility models. On the other hand, because the line distance in the Fine Grid is only 2 inch, the approximation leads to smaller errors, and thus the different impact of the three models is preserved and reflected on Fig. 6(a). Further, close observation of correlation distributions in Fig. 6(b) shows some model differences. The correlation values of the Random Walk model spread wider than values of the other two models. The Levy Walk model also has a somewhat wider distribution than the Brownian Motion model. The Brownian Motion model also shows more spikes with value close to one than the other two models.

When we turn our attention to Fig. 7 we see clearer indications of model impact. In Indoor Trail and Outdoor Trail sets all measurements are taken at the exact locations of the model trails, so there is no loss of precision. Because indoor environments provide rich multipath environments compared to outdoor environments, at a given spatial separation, indoor measurements will be less correlated. The higher correlation in outdoor environment measurements obscures the effects of movement.

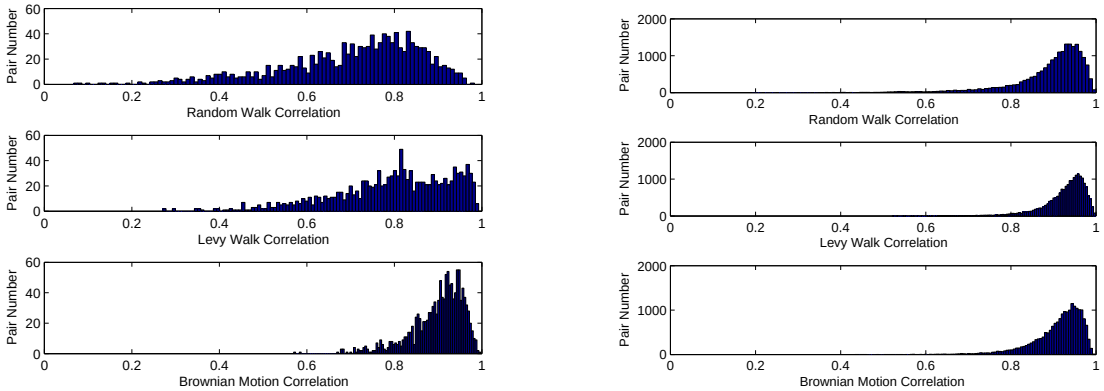
D. Impact of Mobility on Key Generation

In this subsection, we evaluate how mobility models impact the generation of unique quantized CIRs (that we call binary signatures). Fig. 8 compares the unique binary signature rates in two data sets: the Stationary set where all signatures are collected at the same location, and the Grid set where signatures are taken while device can be moved to any location of a grid. Our first discovery is that signatures measured without moving indeed have a large number of duplicates. When quantized with one bit the unique signature rate is



(a) Fine Grid measurements (b) Grid measurements

Fig. 6. Impact of mobility on signature correlations in the two grid measurement sets.



(a) Indoor Trail measurements (b) Outdoor Trail measurements

Fig. 7. Impact of mobility on signature correlations in the two trail measurement sets.

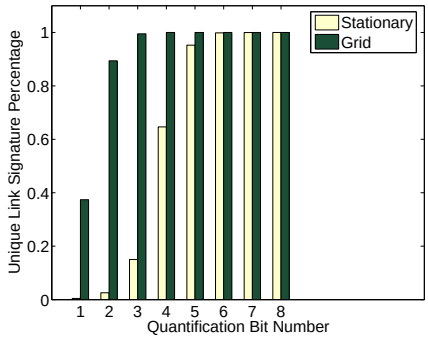


Fig. 8. Impact of mobility adoption on unique binary signatures.

as low as 0.44%. Even when quantized with five bits, there are 5% of duplicate signatures in the Stationary data set. These results certainly show the need for removing duplicates. According to Eq. 5 the computation for an attacker to acquire a legitimate link signature using brute force and public parity symbols drops very fast with larger quantization bit numbers. Thus, the duplicate rates at small quantization numbers play an important role. We also observe that the adoption of device mobility greatly improve the unique signature percentage with the average improvement of 31% as illustrated by the rates of the Grid measurements at different quantization bit values.

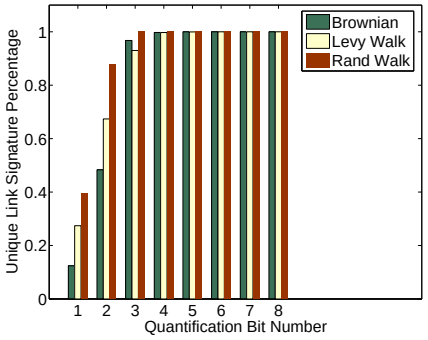


Fig. 9. Impact of mobility models on unique binary signatures.

Fig. 9 shows the impact of different mobility models on the unique signature rate. When comparing this figure with Fig. 8, it is clear that any movement pattern would help lower the duplicate rate and improve signature randomness. When quantized with one or two bits the three mobility models demonstrate similar trend in affecting the unique signature

rate as the trend shown in affecting the continuous-valued signature correlation. The Brownian Motion has the least impact in avoiding duplicate signatures, the Levy Walk has the medium impact, and the Random Walk has the largest impact. However, when quantized with more than two bits, the Brownian Motion model suddenly exhibits a little more impact than the Levy Walk model. This change may reflect that the Levy Walk model results in signatures with bigger differences than those of the Brownian Motion but it does not necessarily create more changes than the Brownian Motion. This situation is quite interesting since it does not show up or is not clear in the continuous-valued measurements. The Random Walk model, on the other hand, consistently yields the highest unique signature rate with all quantization bit numbers.

E. Quality of Key Generation

We evaluate two aspects of the quality of key generation - the quality of keys, and the efficiency of key extraction. We assess the quality of keys using two methods. First, we run 8 tests from the NIST test suite [11] on the secret bits generated using the indoor Levy walk data. Table III shows the results that we obtain from the NIST tests. Because the P-values in all tests are larger than the threshold, 0.01, our secret key bits show good randomness according to the tests. Next, we compute the entropy values of the secret keys. Fig. 10 shows the entropy for different data sets. All entropy values of our keys are very close to 1.0 indicating a high degree of uncertainty. For comparison, we also calculate the entropy values of keys generated using an existing method [5] proposed by Mathur et al. All entropy values from Mathur's method are in the range of $[0.6, 0.8]$. Therefore, our method generates keys with higher entropy in comparison to Mathur's method.

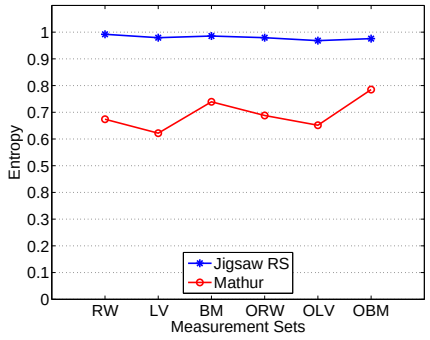


Fig. 10. Entropy comparison between Mathur's method and ours. RW, LV, and BM represent the indoor random walk, levy walk, and Brownian motion measurement sets. ORW, OLV, and OBM are outdoor measurement sets from the respective models

To assess the the efficiency of key extraction, we use a metric called *Secret Bit Rate* that defined as the average number of secret bits extracted from each channel response. Our protocol compresses reconciled signature raw bits to a key size determined by the user. The user needs to estimate this size according to the information entropy of the signature space in the measured environment. Because there is no well recognized way to estimate this size yet, for our evaluation purpose we take a simple and conservative approach to estimate a lower bound of this size. In Fig. 11, we plot the entropy values of the bit stream generated with different quantization bit numbers (per channel response). We find in all six measurement sets, the entropy values monotonously increase with growing quantization bit numbers until they reach a saturation point, after which they fluctuate mildly. It seems before the saturation point, increase in the size of the key would increase entropy while after the saturation point adding more bits will not necessarily results in increased entropy. Based on this observation, we consider the key size generated with quantization bit number immediately before the saturation point as a lower bound of the real key size. Using the lower bound key size we compute the secret bit rates in different measurement sets and plot them in Fig. 12. We also plot the secret bit rates from the existing Mathur's method using the same measurement sets. As illustrated in the figure, while the existing method generates $0.10 - 0.19$ bits per link signature, our method generates $2.59 - 5$ bits per signature, which is more than one order of significance (note the logarithm y axis) higher than the existing method.

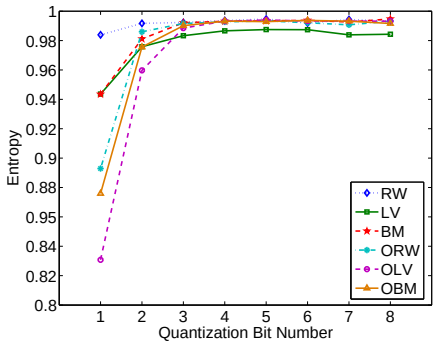


Fig. 11. Entropy comparison between bits generated with different quantization bit numbers.

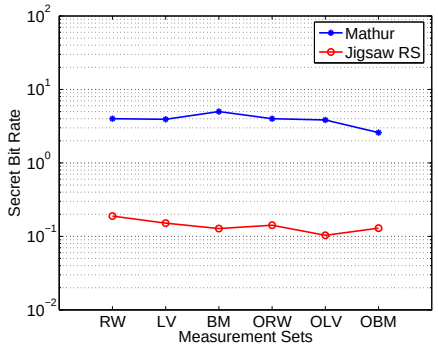


Fig. 12. Secret bit rate comparison between Mathur's method and ours.

Quan Bits	1	2	3	4	5	6	7	8
Frequency	0.81	0.51	0.49	0.48	0.23	0.29	0.18	0.53
Block Frequency	0.78	0.92	0.46	0.64	0.30	0.58	0.61	0.86
Cumulative sums(Fwd)	0.85	0.80	0.73	0.35	0.20	0.38	0.26	0.49
Cumulative sums(Rev)	0.99	0.56	0.38	0.82	0.42	0.28	0.31	0.70
Runs	0.03	0.42	0.99	0.29	0.63	0.76	0.52	0.14
Longest run of ones	0.55	0.97	0.19	0.51	0.53	0.20	0.04	0.61
FFT	0.89	0.89	0.92	0.77	0.98	0.89	0.17	0.07
Approx. Entropy	0.41	0.03	0.17	0.77	0.91	0.90	0.06	0.10
Serial	0.59, 0.72	0.50, 0.75	0.64, 0.37	0.34, 0.37	0.96, 0.88	0.90, 0.98	0.22, 0.39	0.16, 0.10

TABLE III
NIST STATISTICAL TEST P-VALUES OF THE GENERATED KEYS.

VI. Related Work

There is a large amount of literature on extracting secret key bits from variations in received signal strength (RSS) measurements (e.g., [9], [5], [22], [23]). These existing works depend upon movement of the devices or in the surroundings to cause the variations. They do not consider the variations of RSS in space. Similarly, a few of other existing approaches [5], [10], [24], although use the CIR, extract keys from short term temporal variations of the CIR. We use CIR in our research as well. As in [10], we also use forward error correction for reconciling any mismatch in bits at Alice and Bob. However, our work differs from these existing works in the following significant ways. First, we use movement only to sample the channel impulse response space at different locations and then combine these to generate strong keys. As long as this movement is not fully retraceable, the change of location does not need to happen at short intervals. Second, we examine different mobility models to see which ones are more effective in giving us uncorrelated samples. Third, we propose a new Jigsaw encoding scheme that keeps the mismatch rate in reciprocal measurements, at the two parties interested in establishing a secret, low even when channel responses are quantized with increasing bit numbers. Last, we evaluate our approach using extensive measurements in real environments.

VII. Conclusions

We proposed a new approach for secret key establishment between two wireless devices where the two devices measure the CIRs at different locations and combine these measurements to produce a strong secret key. We studied the impact of three mobility models in obtaining uncorrelated CIRs. We also developed efficient mechanisms to encode CIRs and reconcile the differences in the bits extracted between the two devices. Our evaluations showed that our scheme generated very high entropy secret bits at a high bit rate. In the future, we plan to implement our scheme on the Universal Software Radio Peripherals (USRPs) for real time experimentation, and to build a thorough understanding of the strengths and limitations of our methods.

References

[1] T. S. Rappaport, *Wireless Communications: Principles and Practice.* New Jersey: Prentice-Hall Inc., 1996.

[2] H. Hashemi, "The indoor radio propagation channel," *Proceedings of the IEEE*, vol. 81, no. 7, pp. 943–968, July 1993.

[3] A. Goldsmith, *Wireless communications.* Cambridge Univ Pr, 2005.

[4] A. Sayeed and A. Perrig, "Secure wireless communications: Secret keys through multipath," *Acoustics, Speech and Signal Processing*, pp. 3013–3016, 31 2008-April 4 2008.

[5] S. Mathur, W. Trappe, N. B. Mandayam, C. Ye, and A. Reznik, "Radio-telepathy: extracting a secret key from an unauthenticated wireless channel." in *MOBICOM*. ACM, 2008, pp. 128–139.

[6] Z. Li, W. Xu, R. Miller, and W. Trappe, "Securing wireless systems via lower layer enforcements," in *Proc. 5th ACM Workshop on Wireless Security (WiSe'06)*, Sept. 2006, pp. 33–42.

[7] C. Bennett, F. Bessette, G. Brassard, L. Salvail, and J. Smolin, "Experimental quantum cryptography," *Journal of cryptology*, vol. 5, no. 1, pp. 3–28, 1992.

[8] S. Wiesner, "Conjugate coding," *ACM Sigact News*, vol. 15, no. 1, pp. 78–88, 1983.

[9] S. Jana, S. P. Nandha, M. Clark, S. K. Kasera, N. Patwari, and S. Krishnamurty, "On the effectiveness of secret key extraction using wireless signal strength in real environments," in *Mobicom*, 2009.

[10] C. Ye, A. Reznik, and Y. Shah, "Extracting secrecy from jointly gaussian random variables," in *Information Theory, 2006 IEEE International Symposium on*, July 2006, pp. 2593–2597.

[11] NIST, "A statistical test suite for the validation of random number generators and pseudo random number generators for cryptographic applications," December 2008.

[12] N. Patwari and S. K. Kasera, "Temporal link signature measurements and models for location distinction," submitted to IEEE Transactions on Mobile Computing.

[13] R. B. Paris and D. Kaminsky, *Asymptotics and the Mellin-Barnes Integrals.* Cambridge University Press, 2001.

[14] B. Danev, T. S. Heydt-Benjamin, and S. Capkun, "Physical-layer identification of rfid devices," in *Usenix Security Symposium*, 2009.

[15] V. Brik, S. Banerjee, M. Gruteser, and S. Oh, "Wireless device identification with radiometric signatures," in *Proceedings of the 14th ACM international conference on Mobile computing and networking*, 2008.

[16] R. Impagliazzo, L. A. Levin, and M. Luby, "Pseudo-random generation from one-way functions," in *STOC'89*, 1989.

[17] B. H. Bloom, "Space/time trade-offs in hash coding with allowable errors," *Commun. ACM*, vol. 13, no. 7, pp. 422–426, 1970.

[18] J. Proakis and M. Salehi, *Digital communications.* McGraw-Hill New York, 1995.

[19] S. Wicker and V. Bhargava, *Reed-Solomon codes and their applications.* Wiley-IEEE Press, 1999.

[20] L. Breslau, D. Estrin, K. Fall, S. Floyd, J. Heidemann, A. Helmy, P. Huang, S. McCanne, K. Varadhan, Y. Xu, and H. Yu., "Advances in network simulation (ns)," *IEEE Computer*, 2000.

[21] I. Rhee, M. Shin, S. Hong, K. Lee, and S. Chong, "On the levy-walk nature of human mobility," *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, pp. 924–932, April 2008.

[22] B. Azimi-Sadjadi, A. Kiayias, A. Mercado, and B. Yener, "Robust key generation from signal envelopes in wireless networks," in *CCS*, 2007.

[23] M. A. Tope and J. C. McEachen, "Unconditionally secure communications over fading channels," in *Military Communications Conference (MILCOM 2001)*, vol. 1, Oct. 2001, pp. 54–58.

[24] R. Wilson, D. Tse, and R. Scholtz, "Channel identification: Secret sharing using reciprocity in UWB channels," *IEEE Transactions on Information Forensics and Security*, 2007.