This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the Mini-Conference at IEEE INFOCOM 2010

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. The fact that these link signatures can be measured almost symmetrically between two ends of a wireless link [1], but cannot be measured from another distinct location has led researchers to suggest using these for secret key establishment [2]. Secret key extraction from link characteristics has the potential to provide an inexpensive alternative to quantum cryptography [3].

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 [2] 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. [4] 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). 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 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 [2], [5] and make them more robust.

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

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

978-1-4244-5837-0/10/$26.00 ©2010 IEEE

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the Mini-Conference at IEEE INFOCOM 2010

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 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 [6] that we conduct.

II. 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., [4]). Our adversary is also not interested in causing any Denial-of-Service attacks.

III. Mobility Assisted Key Establishment

In this section, we first describe our secret key establishment protocol. Next, we present the important building blocks of the protocol.

A. Key Establishment Protocol

Our key establishment protocol between A and B is divided into three phases. 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 SIGCHK phase, B sends only the parity symbols to A in multiple SIGFEC (short for signature forward error correction) 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.

To convert the bit stream obtained from the CIR measurements, we utilize a key compression function. This function uses the 2-universal hash family to perform *Privacy Amplification* [7]. Privacy amplification minimizes the possible correlation among input bits of the bit stream and compresses the raw bits to the chosen key size with 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.

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 [8] 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. The normalization 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

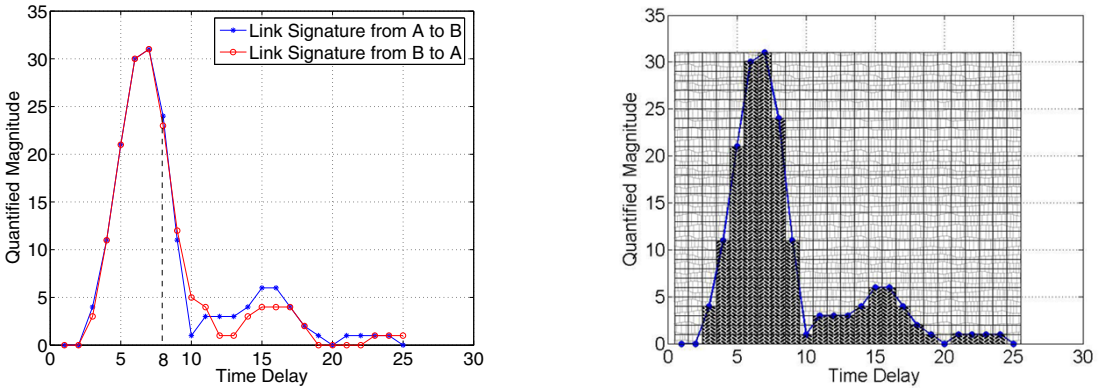
Although uniform quantization is simple and easy to implement, we find when increasing the quantization bit number q from 1 to 8, the rate of the discrepant elements in the quantized CIRs, that are not measured the same at A and B, grows dramatically. Table I lists the discrepancy rate in a sample bi-directional 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. 1(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 full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the Mini-Conference at IEEE INFOCOM 2010

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 I
DISCREPANCY RATE IN RECIPROCAL LINK SIGNATURE MEASUREMENTS



(a) Uniform Quantization (b) Jigsaw Encoding

Fig. 1. 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.

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. 1(a) is encoded with the new method in Fig. 1(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. 1(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 I. We utilize random numbers instead of some constant values in the two public maps 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 [9] to reconcile any discrepancies in reciprocal measurements of CIR.

Each RS 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[2]. 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

[2]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.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the Mini-Conference at IEEE INFOCOM 2010

rate ϵ in Eq. 3.

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

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

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 discrepancy 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 $\lceil \frac{k-t}{2^q} \rceil$ 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}.

IV. Protocol Evaluation

In this section, we evaluate the proposed protocol in two steps. First, we assess the impact of device mobility on measured link signatures. Then, we evaluate the quality of key generation.

A. Measurement Campaign

To study the impact of device mobility, we use three mobility models: *random walk* [10], *Levy walk* [11], and *Brownian motion*. These models dictate trails with decreasing diffusion, so they form a valuable suite to the study. We measure link signatures with a pair of Direct Sequence Spread Spectrum transmitter and receiver as described in [12].

We have taken 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 took the first set inside a large lobby of an engineering building on the University of Utah campus (*Indoor Trail* set). We acquired the second set on a flat square outside of the building (*Outdoor Trail* set). We collected 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).

B. 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. We histogram the values that contribute to the correlation coefficient to study their distribution.

Fig. 2 shows clear indications of the model impact. The correlation among the CIRs of the random walk model lie in [0.1, 1], while that of the Levy walk model and the Brownian motion model lie in [0.3, 1] and [0.6,1], respectively. It appears that the more diffusive a mobility model is the less correlated its measurements become. We find the similar trend in the Outdoor Trail measurements and Fine Grid measurements as well. Surprisingly, the Grid measurements in Fig. 3 does not show this trend. In comparison of the two figures, we notice the striking difference between their distributions. The correlation values of the Indoor Trail measurements predominantly concentrate to the right of 0.6, whereas the correlation values of the Grid measurements are mostly centered at 0.4. Because we use the Grid measurements to approximate measurements generated by the mobility models and because the grid line distance in the Grid set is 1 foot, we suspect that the one-foot distance is far enough to make most signature measurements uncorrelated and thus dilute the effect of 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.

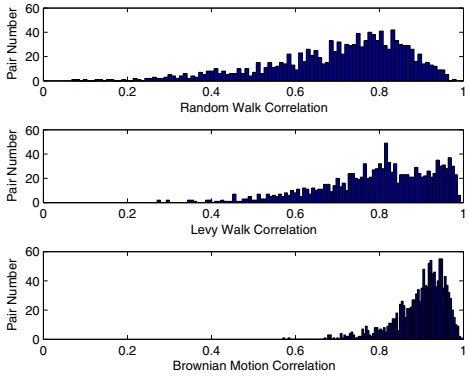


Fig. 2. Signature correlations in the Indoor Trail measurement set

C. 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

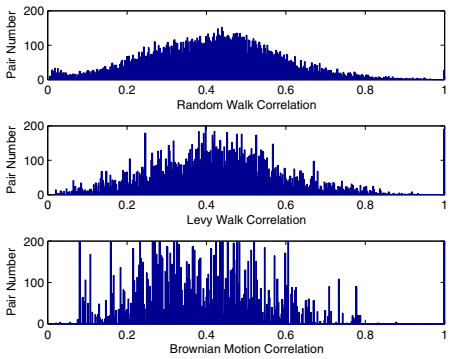


Fig. 3. Signature correlations in the Grid measurement set

tests from the NIST test suite [6] on the secret bits generated using the indoor Levy walk data. 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. 4 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 [2] 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.

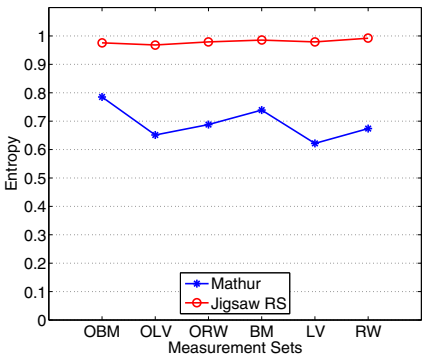


Fig. 4. Entropy comparison between Mathur's method and ours. OBM, OLV, and ORW represent the outdoor Brownian motion, Levy walk, and random walk measurement sets. BM, LV, and RW are indoor measurement sets from the respective mobility models.

To assess the the efficiency of key extraction, we use a metric called *Secret Bit Rate* that is 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. We plot the entropy values of the bit stream generated with different quantization bit numbers (per channel response). We find in all six trail 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. 5. 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 channel response, our method generates $2.59 - 5$ bits per channel response, which is more than one order of significance (note the logarithm y axis) higher than the existing method.

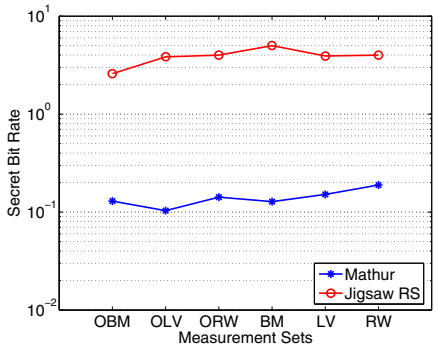


Fig. 5. Secret bit rate comparison between Mathur's method and ours.

REFERENCES

[1] A. Goldsmith, *Wireless communications*. Cambridge Univ Pr, 2005.
[2] 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.
[3] S. Wiesner, "Conjugate coding," *ACM Sigact News*, vol. 15, no. 1, pp. 78–88, 1983.
[4] 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.
[5] 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.
[6] NIST, "A statistical test suite for the validation of random number generators and pseudo random number generators for cryptographic applications," December 2008.
[7] R. Impagliazzo, L. A. Levin, and M. Luby, "Pseudo-random generation from one-way functions," in *STOC'89*, 1989.
[8] J. Proakis and M. Salehi, *Digital communications*. McGraw-Hill New York, 1995.
[9] S. Wicker and V. Bhargava, *Reed-Solomon codes and their applications*. Wiley-IEEE Press, 1999.
[10] 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.
[11] 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.
[12] J. Zhang, M. H. Firooz, N. Patwari, and S. K. Kasera, "Advancing wireless link signatures for location distinction," in *ACM Intl. Conf. on Mobile Computing Networking (Mobicom'08)*, Sept. 2008.