© 2009 IEEE. doi: 10.1109/NPSEC.2009.5342245

On Key Agreement in Wireless Sensor Networks based on Radio Transmission Properties

Matthias Wilhelm, Ivan Martinovic, and Jens B. Schmitt

disco | Distributed Computer Systems Lab
TU Kaiserslautern, Germany
{wilhelm, martinovic, jschmitt}@cs.uni-kl.de

***Abstract*—Recently, several research contributions have justified that wireless communication is not only a security burden. Its unpredictable and erratic nature can also be turned against an adversary and used to augment conventional security protocols, especially key agreement. In this paper, we are inspired by promising studies on such key agreement schemes, yet aim for releasing some of their limiting assumptions. We demonstrate the feasibility of our scheme within performance-limited wireless sensor networks. The central idea is to use the reciprocity of the wireless channel response between two transceivers as a correlated random variable. Doing so over several frequencies results in a random vector from which a shared secret is extracted. By employing error correction techniques, we are able to control the trade-off between the amount of secrecy and the robustness of our key agreement protocol. To evaluate its applicability, the protocol is implemented on MicaZ sensor nodes and analyzed in indoor environments. Further, these experiments provide insights into realistic channel behavior, available information entropy, and show a high rate of successful key agreements, up to 95 %.**

I. Introduction

From a security perspective, wireless communication is usually considered a disadvantage. Its broadcast nature does not allow for network traffic to be physically separated, and the typically performance-limited wireless clients are constrained in utilizing conventional key agreement protocols. But while an adversary eagerly takes advantage of such wireless peculiarities to construct sophisticated attack vectors against different security objectives, the existing security designs abstract from them. Recently, a number of research contributions turned the table by using the nature of wireless communications as a source of novel security features to extend conventional security protocols (e.g., [2], [11], [8], [1], [4], [7]). Specifically, [8] and [1] follow an information-theoretic approach to derive secret keys from the wireless channel by taking advantage of the strong decorrelation of channel behavior in both time and frequency domain. Such rapid decorrelation is especially experienced in the measured received signal strength (RSS) and consequently, as long as not being on the same physical position as legitimate nodes, an attacker remains ignorant of their RSS estimations. While existing contributions offer valuable insights for deriving secrets from such physical phenomena, their major assumption lies in device mobility and the strong impact of the resulting Doppler effect. Consequently, the following questions remain unanswered: (a) can static networks profit from the unpredictability of the wireless channel, i.e., if neither Doppler effect nor Rayleigh fading can be assumed, and (b) what are the trade-offs between the secrecy of the derived shared secret and the robustness of the key agreement, i.e., the protocol's sensitivity to errors in estimating the channel behavior by low-cost and resource-limited hardware? Specifically, the contribution of this paper is:

- design of a key-agreement protocol applicable in both static and dynamic networks,
- implementation of the protocol on "off-the-shelf" MicaZ sensor motes, and
- experimental analysis of the protocol using a real-world wireless sensor network.

A. Shared Secrets from the Wireless Channel

Transmitted signals are attenuated due to path loss, shadowing and multipath fading [10]. While path loss is a function of the distance between sender and receiver, both other components are depending on the signal frequency and on the surrounding environment. Arriving at the receiver through multiple paths, the received signal is modified by different phase offsets which may either result in constructive or destructive interference, i.e., fading. A small change in position can lead to drastically changed signal paths, resulting in a different attenuation. The related key agreement approaches base their security on randomness generated by changing paths due to continuous movement. Without such movement, the measured values are stationary and further probing does not increase the secrecy. But an unpredictable change of attenuation is also observed under a variation of frequency, as the phase shifts of each multipath component depend on both the path and the frequency of the signal. We aim to exploit this property to generate strong secret keys in a reliable way. By using the frequency-selectivity of channel fading as the source of randomness, we can avoid the necessity for movement during key agreement, even if an eavesdropper can monitor the probing messages and

Notice: This work presents preliminary results, for the definite version please cite
M. Wilhelm, I. Martinovic, J. B. Schmitt. *"Secure Key Generation in Sensor Networks Based on Frequency-selective Channels."* IEEE J. Sel. Areas Commun. 31(8), pp. 1779–1790, Sept. 2013. doi: 10.1109/JSAC.2013.130911

has knowledge of the positions of sender and receiver and of the environment.

In order to use the wireless channel's properties, both communicating parties must be able to use it as a correlated source of secret information. The principle of reciprocity states that the same attenuation is experienced at two communicating nodes, as the electromagnetic waves travel on the same paths. In our experiments, we observed sufficient reciprocity in order to justify the goal of building a reliable protocol upon this principle, as well as a high degree of uncertainty in the amount of attenuation. An illustrative measurement with two MicaZ motes over 16 different channels in the 2.4 GHz range is given in Figure 1. The difference in wavelengths between two adjacent channels with 5 MHz spacing is $\Delta\lambda \approx 0.259mm$, yet even with this small deviation, a strong frequency-selectivity can be observed. The relatively small deviations in the RSS values between the two probing nodes are caused by imperfect reciprocity, interference from concurrent wireless traffic and other sources such as noise in the measurement circuits. Yet, if we can overcome these deviations, two parties (which we refer to as Alice and Bob) can use these measurements to generate a shared secret. Regarding the secrecy, we can observe that there is also partial information available to an eavesdropper (Eve), since assumptions about the path loss component, and to a smaller degree about other environmental effects, are possible. An example of this is shown in Figure 1d. A shift of 3 cm of one of the sensor motes results in a different signal strength profile on both sides, yet the measurements remain in a similar range. These defects in the random string must be dealt with in order to generate a truly strong secret.

In Section II, we introduce the necessary building blocks and in Section III we show the work-flow of our key agreement protocol. Section IV presents our experimental analysis of the secrecy capacity and the robustness of our approach. Finally, we present interesting future directions and provide a conclusion.

II. Protocol Building Blocks

This section describes the way from measurements to the derivation of a strong secret bit string. Our proposed protocol conceptually operates as follows: (*i*) make estimations of the signal strengths on different frequencies, (*ii*) reconcile these estimations such that Alice and Bob have a common seed for a secret and finally (*iii*) amplify the secrecy of the seed to a strong secret.

Each of the necessary steps is presented in the following subsections. We employ results of information and coding theory as a basis for the protocol. In this context, the work of Maurer and Wolf [9] introduced a framework for secure key exchange from correlated random variables.

A. Estimation of the Signal Strength

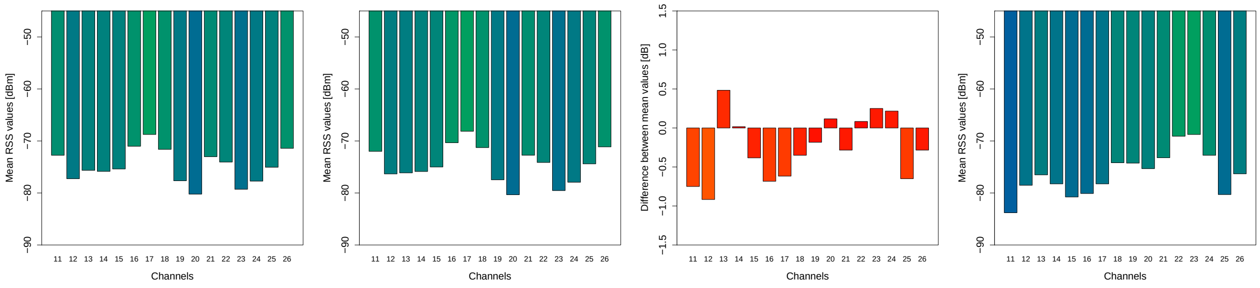
First, we formulate our domain specific terms into the terminology of coding theory. This section provides the basic notation that we use in the remainder of the paper.

In the following, we assume that we can conduct measurements by sampling RSS values on a set of n different frequencies $\mathcal{F} = \{f_1, \ldots, f_n\}$ (also referred to as *channels*). We view the mean of these samples taken from an individual channel f_i as a random variable M_i, and the means of all n channels as the random vector $\mathbf{M} = (M_1, \ldots, M_n)$. A realization, the outcome of our measurements is $\mathbf{m} = (m_1, \ldots, m_n)$, with $m_i \in \mathcal{M} = [m_{min}, m_{max}]$, the range of mean values that can be measured. We assume that $\mathcal{M}$ is a finite subset of $\mathbb{R}$, i.e., only a finite precision in the measurements is achieved, and use properties of $\mathbb{R}$ such as ordering and relations when discussing dependencies of elements in $\mathcal{M}$. As an example for this set, in our wireless sensor network (WSN) measurements we used $\mathcal{M} = [-104, -40]$ dBm, with a precision depending on the number of samples taken, since each RSS sample is integer valued. We associate $\mathcal{M}$ with a distance function $dis : \mathcal{M} \times \mathcal{M} \to \mathbb{R}^+$ defined as $dis(m, m') := |m - m'|$, which is the difference in dB in our case. Thus, $\mathcal{M}$ together with this distance function constitutes a *metric space*.

B. Secret Reconciliation using Codes

Given the values $m, m' \in \mathcal{M}$ measured by Alice and Bob, our goal is to obtain a shared value without revealing information to Eve. To reliably reconcile information, efficient error correction is crucial because brute force approaches using all possible combinations are infeasible in the context of resource-limited devices. Coding theory provides a useful framework to describe error-correcting codes [6]. In general, a code $\mathcal{C}$ is a subset of a metric space $\mathcal{M}$, $\mathcal{C} = \{c_1, \ldots, c_K\} \subseteq \mathcal{M}$, with a total of K elements. The map from $\mathcal{M}$ to $\mathcal{C}$ is called *encoding*, denoted as enc. The most important property of a code for our application is the *error-correcting distance* t of $\mathcal{C}$. This is the smallest distance for which an $m \in \mathcal{M}$ is encoded uniquely, i.e., all values m, m' are encoded to c given their distance to c is small enough. We refer to this value of t as the *tolerance* of the code.

Common codes such as Hamming and Reed-Solomon codes operate on the Hamming distance metric and therefore lead to undesirable tolerance characteristics. Thus, we need to construct a code that considers our special distance function. The construction is as follows: we choose $K = 2^p$ elements of $\mathcal{M}$ such that we have the same distance d between all codewords, where p is the number of bits that are needed to identify a codeword. We denote this code as $\mathcal{C}_p = \{c_1, \ldots, c_{2^p}\}$, the mapping to the binary representation as $bin : \mathcal{C}_p \to \{0, 1\}^p$, which maps codewords to binary strings. Since m_{min} and m_{max}



(a) Alice's View (b) Bob's View (c) Difference (d) Shifted Bob

Figure 1: Perceived signal strength of two sensor nodes, and the deviations between the two measurements. Figure 1d shows the effects on the received signal strength when Bob's position is shifted by 3 cm.

are both fixed values, the distance d between neighboring codewords is reduced as the number of codewords increases. The relation is given by $d = \frac{|m_{max} - m_{min}|}{2^p - 1}$. The tolerance of such a code is given by $t = \frac{d}{2}$, since all codewords are evenly spaced. The process of encoding maps the value m to the codeword c with the minimal distance in $\mathbb{R}$, which can be viewed as a *quantization* of the measured value.

The amount of uncertainty is reduced in this process as some values become impossible, but at the same time the tolerance for deviations is increased. Thus, we can trade robustness vs. secrecy by choosing a code $\mathcal{C}_p$ with suitable parameter $p \in \mathbb{N}$ which is able to correct errors in measurements given $dis(m, m') < t$. Similarly to the distance function, the tolerance can be described as acceptable measurement deviations, such as $\pm 1\,\text{dB}$ in received signal strength.

With this construction, we are usually able to reconcile many deviations between m and m' given $dis(m, m') < t$. Still, some constellations are possible such that m and m' are encoded to two different codewords (e.g., given $\mathcal{C}_5$, $m = -70.9\,\text{dBm}$ and $m' = -71.1\,\text{dBm}$ are encoded as -70 and -72, respectively). To correct these error patterns, we need to send a public piece of information P that helps Bob to reconcile his measurement and recover the same codeword as Alice. Our construction is straightforward: Alice calculates $P = enc(m) - m$, the shift that is necessary from m to the corresponding codeword $c = enc(m)$, and uses c as her secret information. This shift is always smaller than or equal to $\frac{t}{2}$, revealing only the information that is discarded by Alice and Bob anyway due to the rounding property of the code. She then sends P via a public channel to Bob, who uses P to generate the same codeword c using his measurement m' (given $dis(m, m') < t$) by calculating $c = enc(m' + P)$. To prove the correctness, consider that when $dis(m, m') < t$, then $dis(m+P, m'+P) < t$, and thus $dis(c, m'+P) < t$. Finally, since the error-correcting distance of the code is t, $m' + P$ is encoded to c by Bob as well.

C. *Amplification using Randomness Extractors*

After this reconciliation step, Alice and Bob share a secret seed generated from the wireless channel. This seed based on the n codewords corresponding to the different channels is not yet suitable to be used as a key due to the non-uniformity of its random distribution.

The amount of uncertainty of a random variable can be quantified by the notion of *entropy*. We are interested in the minimum amount of secret information in a variable, or put differently, the *predictability* of a random variable. A metric for this purpose is the *min-entropy* [9] of a discrete random variable A with $supp(A) = \mathcal{A}$, defined as

$$\mathbf{H}_\infty(A) = -\log_2(\max_{a \in \mathcal{A}} Pr[A = a]).$$

The available min-entropy is maximal in case A is uniformly distributed, i.e., in our context this would mean no preference for some RSS measurements over others is present. As our random variables of interest M_i are not uniformly distributed due to a baseline value given by the path loss, the amount of min-entropy is given by the distribution of signal powers which are affected by non-uniform factors like multipath fading.

Some constructions are known (e.g., [5]) to be able to extract secure bit strings from $\mathbf{M}$ with a length in the order of the min-entropy. We use the notion of randomness extractors [3] as a method to produce strong secrets:

Definition 1. Let $ext : \{0,1\}^{n_0} \to \{0,1\}^{l_0}$ be a polynomial time probabilistic function which uses r bits of randomness. We say that ext is an efficient $(n_0, h_{min}, l_0, \epsilon)$-*strong extractor* if for all distributions W over $\{0,1\}^{n_0}$ with min-entropy h_{min} holds

$$SD((ext(W; X), X), (U_{l_0}, X)) \leq \epsilon,$$

where X is uniform on $\{0,1\}^r$, U_{l_0} is the uniform distribution on l_0 bit binary strings and SD is the statisti-

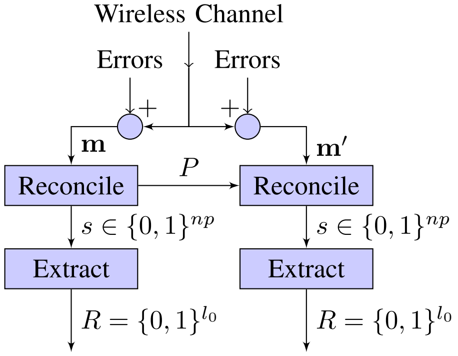


Figure 2: Key agreement protocol: (*i*) make estimations of the state of the wireless channel, (*ii*) reconcile these estimations to a common seed s and (*iii*) amplify the secrecy of the seed for a strong secret R.

cal distance between two probability distributions.[1]

As an implementation of this strong extractor, we use *universal hash functions* (UHF) [5] to extract the maximum possible amount of entropy from our input. Due to space limitations, we refer the reader to [12], which describes UHF for resource-limited devices.

III. Protocol Design

A bird's eye view on the key generation is given in Figure 2. Using the building blocks from the previous section, we are able to compose a protocol which can be used for key agreement in a way that is both robust and secure.

A. Key Agreement Protocol

The complete protocol is shown as pseudo code in Protocol 1. In the *probing phase*, k samples of the received signal strength are gathered for each of the n available channels. Then the means m_i of those samples are computed for every channel, resulting in $\mathbf{m} = (m_1, \ldots, m_n)$. A set of samples must be gathered to reduce the impact of temporal effects on the measurements. In the *key generation phase*, a suitable code $\mathcal{C}_{p_i}$ with tolerance t_i is chosen to extract the maximum amount of entropy. It is possible to use a different parameter p_i for each channel f_i depending on the expected error in the measurements. Alice encodes each of the m_i using $\mathcal{C}_{p_i}$ to create a tuple of codewords $\mathbf{c} = (c_1, \ldots, c_n)$ and creates the string $\mathbf{P} = (P_1, \ldots, P_n)$ in order to send it via a public channel to Bob. Given $dis(m_i, m'_i) < t_i$ for all $i = 1, \ldots, n$, he can now recreate the same vector of codewords $\mathbf{c}$ by applying error correction and encoding his measurements $\mathbf{m}' = (m'_1, \ldots, m'_n)$. Both Alice and Bob calculate their secret seed s by converting their codewords into a single bit string with length $|s| = \sum_{i=1,\ldots,n} p_i$ and amplify this seed by employing a $(n_0 = |s|, h_{min}, l_0, \epsilon)$-randomness extractor ext to compute the strong secret string R. R is a bit string of length l_0, which is given by the available entropy h_{min} and a chosen ϵ which measures the remaining non-uniformity of R. Finally, in the *acceptance phase*, a challenge-response scheme ensures that the secret key was created successfully. In case of failure, Alice can attempt to alter the tolerances by modifying some of the p_i in order to increase the odds that the next run will be successful. Note that this key generation protocol can be viewed as an $(\mathcal{M}, h_{min}, l_0, t, \epsilon)$-fuzzy extractor as described by Dodis et al. [3].

IV. Experimental Analysis

We now evaluate the applicability of the protocol, and describe insights on the amount of secrecy that our concept can offer as well as its robustness in real-world environments.

A. Testbed

Our testbed consists of MicaZ nodes equipped with CC2420 radio transceiver chips using omnidirectional antennas for a maximum of reciprocity. The experiments were conducted over several days on a university floor, and during the measurements a WLAN access point was operating in the 2.4 GHz band, i.e., the experiments are performed in a real-world environment with unpredictable factors.

The sampling process is initiated and managed by Alice. Initially, she sends a sampling message to Bob, who will record the RSS value for this message and sends a reply, which in turn is measured and replied by Alice. Both parties were programmed to respond to sampling messages as fast as possible, which ensures that the answer is sent back during the channel coherence time in which both are able to observe the same channel characteristics. When enough samples for the current channel are collected, Alice initiates a channel switch and continues sampling on the next channels until the measurement phase is complete. This process takes the largest share of the overall time of key agreement, with a duration of approximately 7.5 seconds.

B. Robustness

The protocol is guaranteed to find a shared secret if the deviations between Alice and Bob are bounded. In this experiment, we measure from different distances, both with and without line of sight connections in order to quantify the performance and robustness of our proposed protocol. A total of 175 different positions without repetitions was tested.

The experiments show that our protocol is usable in real-world applications. The success rates are given in Figure 3 (a,b). With a tolerance of ±1 dB, we can achieve

[1] The statistical difference is defined as

$$SD(A,B) = \frac{1}{2}\sum_{\nu} |Pr(A=\nu) - Pr(B=\nu)|\,.$$

```
Protocol 1 Key Agreement
Measurement Phase: $k$ probes are exchanged on each channel in $\mathcal{F}$ to estimate the received signal strength means.
Key Generation Phase:
  1) For each channel $f_i \in \mathcal{F}$:
      a) Alice chooses an appropriate error-correcting code $\mathcal{C}_{p_i}$.
      b) Alice uses error correction on the mean $m_i$ and produces the codeword $c_i$ and a public string $P_i$ for this channel.
  2) Alice sends the collection of reconciliation strings $\mathbf{P} = (P_1, \dots, P_n)$ to Bob.
  3) Bob repairs his measurements $\mathbf{m}'$ to encode to the same codewords $\mathbf{c}$ as Alice.
  4) Both parties use the seed $s = bin(c_1)||\dots||bin(c_n)$, the concatenation of the binary representations of the codewords.
  5) Alice and Bob generate $R \in \{0,1\}^{l_0}$ by calculating $ext(s)$.
Acceptance Phase: A challenge-response scheme is used to validate the secret.
```

agreement on the first run with an accuracy of 69 %. An adapted choice of the tolerance interval gives an increase in robustness, while at the same time sacrificing a minimal amount of entropy. With a fixed tolerance of 2 dB, nearly all positions reach agreement on the first try, with a relative frequency of 96 %. The histogram of deviations between the two sensor nodes (c.f. Figure 3c) suggests that it can be approximated well by a Normal distribution with a standard deviation of roughly $\sigma = 1\,\text{dB}$.

C. Entropy and Secrecy

In this experiment, we want to quantify the amount of secrecy of the shared strings. A fixed distance from the master node of 3 meters was used so that the path loss is not a factor in the estimation of the available entropy, in two scenarios: a single room with a diameter of more than 7 meters so that there is always a line of sight (LOS) connection between the nodes, and an experiment across several rooms such that walls contribute to the overall attenuation.

We estimate the amount of available secret bits with an analysis of the distribution of observed codewords. The histogram of codewords is given in Figure 4. The min-entropy of an individual channel can be estimated from the relative frequency of the most frequent codeword. Both the LOS and non-LOS case have an entropy of at least 2 bits per channel, given a tolerance of ± 1 dB, which is sufficient according to our experiments. The average amount of uncertainty for an eavesdropper is $\mathbf{H}_\infty = 2.246$ bits for the first scenario with a line-of-sight connection and a fixed distance of 3 m, $\mathbf{H}_\infty = 2.149$ bits for the second scenario with additional attenuation from walls. As a comparison, the average min-entropy in our long-term measurements was $\mathbf{H}_\infty = 2.749$ bits per channel for mixed distances.

So far, we have focused on the min-entropy of individual channels. An estimation for the joint entropy is given by the sum of min-entropies of each channel. For example, in our WSN setting with 16 channels the joint entropy is approximately 44 secure bits. However, as the RSS values on different channels are not independent from each other, this is only an estimation. An in-depth analysis of this channel interdependency is left for future work. Yet, if a wireless technology supports a wider frequency spectrum (such as cognitive radios), these dependencies can be reduced and a higher number of channels can be employed to provide additional entropy for longer secrets.

D. Impact of Tolerances

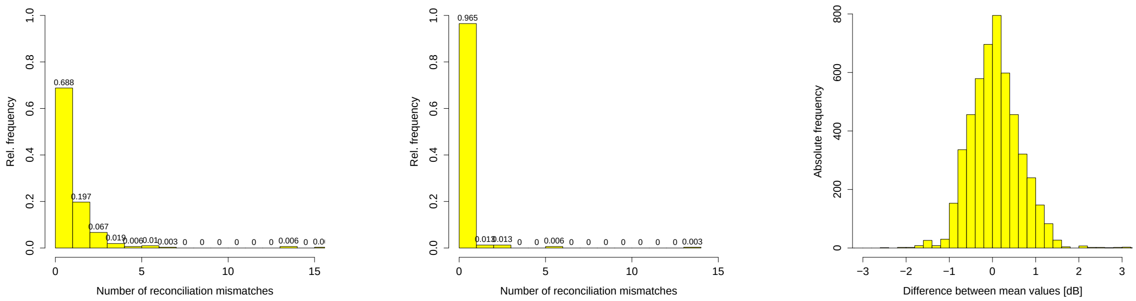
We experimented with the impact of tolerance intervals on secrecy. We evaluated the impact of larger tolerances with t taking the values of $\{1, 2, 4\}$, estimating the entropy in each case. As an example for the entropy loss, with tolerance changing from $t = 1$ to 2, the entropy of the long-term experiment drops from $\mathbf{H}_\infty = 2.749$ bits to $\mathbf{H}_\infty = 2.038$ bits. A tolerance of 4 finally decreases the entropy to 1.22 bits, but such a large amount of tolerance is not necessary, as the measurements are stable. The measured histograms for $t \in \{1, 2\}$ are given in Figure 4c and 4d.

V. Conclusion and Future Work

In this work, we presented a novel key agreement protocol that is based on the physical properties of frequency-selectivity of the wireless channel response as a source of shared randomness. By relying on slow fading, we can remove the limitation of a strict need for swiftly changing environments. We showed in extensive experiments that the protocol produces strong secrets in a reliable way and is applicable even on resource-constrained devices such as sensor nodes. By a number of experiments, we showed that the use of received signal strength is both a stable as well as unpredictable source for shared secret information. Our protocol can reach agreement in excess of 95 % on the first run.

Acknowledgment

The authors gratefully acknowledge the financial support from the Carl-Zeiss Foundation.

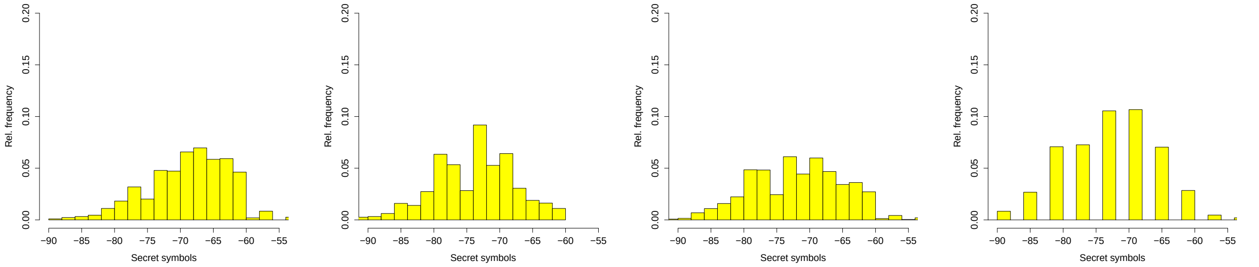


(a) Success rate with a tolerance value of $\pm 1\,\mathrm{dB}$

(b) Success rate with a tolerance value of $\pm 2\,\mathrm{dB}$

(c) Deviations of RSS values between Alice and Bob

Figure 3: Success rates of the key agreement protocol from 175 positions for different error tolerance values. By increasing tolerances, the number of mismatches decreases, but this is paid with a reduction in secrecy capacity.



(a) Line of sight

(b) Non-LOS

(c) Mixed distances, $t = 1$

(d) Mixed distances, $t = 2$

Figure 4: Estimation of available min-entropy $\mathbf{H}_\infty$ of a single channel. The histograms (a)-(c) show the relative frequency of all observed codewords subject to tolerance $t = 1$. The measured RSS mean values (in dBm) are rounded to even values, in this case to the values in $\mathcal{C}_5 = \{-104, -102, \ldots, -40\}$. The amount of uncertainty for an eavesdropper, the average min-entropy per channel, is $\mathbf{H}_\infty = 2.246\,$bits for LOS, $\mathbf{H}_\infty = 2.149\,$bits for non-LOS and $\mathbf{H}_\infty = 2.749\,$bits for mixed distances, respectively. As an example for the entropy loss with tolerance $t = 2$ (Figure 4d), the entropy of the mixed case drops to $\mathbf{H}_\infty = 2.038\,$bits, as the number of codewords is reduced.

References

[1] Babak Azimi-Sadjadi, Aggelos Kiayias, Alejandra Mercado, and Bulent Yener. Robust Key Generation from Signal Envelopes in Wireless Networks. In *CCS '07: Proceedings of the 14th ACM Conference on Computer and Communications Security*, pages 401–410, New York, NY, USA, 2007. ACM.

[2] Y. Chen, W. Trappe, and R. Martin. Detecting and Localizing Wireless Spoofing Attacks. In *Proceedings of the Fourth Annual IEEE Communications Society Conference on Sensor, Mesh, and Ad Hoc Communications and Networks*, pages 193–202, May 2007.

[3] Yevgeniy Dodis, Rafail Ostrovsky, Leonid Reyzin, and Adam Smith. Fuzzy Extractors: How to Generate Strong Keys from Biometrics and Other Noisy Data. *SIAM Journal on Computing*, 38:97, 2008.

[4] D. B. Faria and D. R. Cheriton. Detecting Identity-based Attacks in Wireless Networks using Signalprints. In *WiSe '06: Proceedings of the 5th ACM workshop on Wireless Security*, pages 43–52, September 2006.

[5] Johan Håstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. A Pseudorandom Generator from any One-way Function. *SIAM Journal on Computing*, 28:12–24, 1999.

[6] David J. C. MacKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, 2003.

[7] I. Martinovic, F. A. Zdarsky, M. Wilhelm, C. Wegmann, and Jens B. Schmitt. Wireless Client Puzzles in IEEE 802.11 Networks: Security by Wireless. In *Proceedings of the ACM Conference on Wireless Network Security (WiSec 2008)*, Alexandria, VA, USA, March 2008.

[8] Suhas Mathur, Wade Trappe, Narayan Mandayam, Chunxuan Ye, and Alex Reznik. Radio-telepathy: Extracting a Secret Key from an Unauthenticated Wireless Channel. In *MobiCom '08: Proceedings of the 14th ACM International Conference on Mobile Computing and Networking*, pages 128–139, New York, NY, USA, 2008. ACM.

[9] U. Maurer and S. Wolf. Secret-Key Agreement Over Unauthenticated Public Channels - Parts I-III. *IEEE Transactions on Information Theory*, 49(4):822–851, April 2003.

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

[11] S. Čapkun, R. Rengaswamy, I. Tsigkogiannis, and M. Srivastava. Implications of Radio Fingerprinting on the Security of Sensor Networks. In *Proceedings of the 3rd International Conference on Security and Privacy in Communication Networks*, September 2007.

[12] Kaan Yüksel, Jens-Peter Kaps, and Berk Sunar. Universal Hash Functions for Emerging Ultra-Low-Power Networks. In *Proceeding of The Communications Networks and Distributed Systems Modeling and Simulation Conference (CNDS)*, San Diego, CA, January 2004. Society for Modeling and Simulation International (SCS).