RECiP: Wireless Channel Reciprocity Restoration Method for Varying Transmission Power

Gerhard Wunder, Rick Fritschek, Khan Reaz
Heisenberg Communications and Information Theory Group
Freie Universität Berlin,
Takustr. 9, D–14195 Berlin, Germany
Email: wunder@zedat.fu-berlin.de, rick.fritschek@fu-berlin.de, kahn.reaz@ieee.org

***Abstract*—The use of wireless channel reciprocity properties for secret key generation has been an attractive method for many wireless applications. Most of the available methods rely on some sort of pilot signalling from the transceiver. This reduces the intrinsic security of the generated key. In this paper we have introduced a novel key generation and exchange method that requires no pilot signalling and have also included an algorithm to restore wireless channel reciprocity properties for varying transmission power. Finally we presented information theoretic analysis and the result of our implementation on off-the-shelf TelosB motes.**

***Index Terms*—Physical-Layer Security, Transmission Power Control, Channel Reciprocity**

I. Introduction

Secret key generation (SKG) from channel randomness is a promising path in IoT-related scenarios, e.g. with massive connectivity or the tactile Internet [1], [2], and was introduced in [3]. In these scenarios, Alice communicates with Bob (or several Bobs) representing resource-constrained devices. Moreover, if Alice represents a low-cost access point with "off-the-shelf" hardware, even Alice might be resource-constrained. Both Alice and Bob use lightweight symmetric ciphers, which require at least 128 bits entropy to be considered as secure against attacks by an eavesdropper Eve, notably assuming that Eve has no further side information. A crucial step in such SKG schemes is the channel estimation step where both Alice and Bob transmit known pilot signals to measure the channel. Assuming channel reciprocity, the measurements are highly correlated and hence represent a source of common randomness. Using the standard procedure over a public link reconciles such measurements between Alice and Bob to finally extract a common key [2].

Today, most available channel-based SKG schemes use received signal strength indicators (RSSIs), as this is typically the only type of channel information that is readily available with off-the-shelf hardware. However, while the use of RSSI can greatly simplify the implementation of the scheme (longer channel coherence times, reciprocity easier to establish etc.) it aggravates some of the main pitfalls of channel-based SKG schems: 1) RSSI traces typically have low entropy in stationary environments, 2) they are easily eavesdrop on and easier to manipulate, and 3) reciprocity is often broken when devices are close due to highly nonlinear power amplifiers. In this situation, powers have to be adapted to actual scenario dependent on the location of acces point and devices. Altogther, as of today, these pitfalls prevent channel-based SKG from widespread deployment in the IoT.

Contributions: In this paper we introduce a new RSSI SKG scheme called the RECiP protocol which alleviates some of the mentioned problems. A most surprising feature of RECiP is that it allows to adapt the transmission powers "on the fly" (e.g. while transmitting payload) without neither informing Alice nor Bob on any change of the parameter settings. RECiP uses a loophole in the classical pilot-assisted signaling scheme and extracts common randomness just from the reciprocity property (but *not* from highly correlated measurements!). We also indeed show that information-theoretically higher key rates compared to the classical scheme are possible due to the additional "local" (i.e. *not* common!) randomness at both Alice and Bob. Hence, RECiP can potentionally overcome some of the intrinsic problems of channel-based SKG schemes which will be shortly discussed as well. We present the full RECiP protocol in all its details. Eventually, we discuss some practical implementation of RECiP and show simulation results.

II. System model

The system model is illustrated in Fig. 1. Alice communicates with Bob and afterwards Bob communicates with Alice, both in a non-duplex way. Each of them also have a local source of randomness ω_A and ω_B which can be used for the inputs. For sake of simplicity we assume that a wire-tapper Eve has only access to a public channel over which Alice and Bob can communicate without noise impairment. Therefore Eve has no side-information from the wireless channel, see Fig. 1.[1] The two direct channels experience fading K and K', and have a source of additive white Gaussian noise $Z_i \sim \mathcal{N}(0, N)$. Moreover, we assume reciprocity of the channel gain parameters. The result is that the fading parameters K and K' are the same for the duration of one

This work was carried out within DFG grant WU 598/8-1 (DFG Priority Program on Compressed Sensing) and Providing Physical Layer Security for the Internet of Things (PROPHYLAXE), which was funded by the Federal Ministry of Education and Research (BMBF) under grant number 16KIS0005K.

[1] Depending on the channel, side-information at Eve will most probably decrease the secure key rate, since Eve can obtain information about both local randomness contributions.

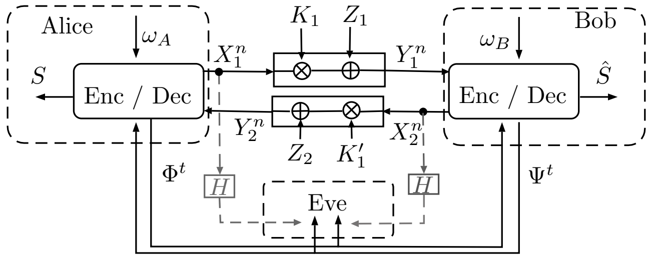

Figure 1. Illustration of the system model with (dashed communication) and without side information at Eve

communication round. To simplify the matter, we assume the fading to change randomly after every communication round with a Gaussian distribution $K \sim \mathcal{N}(\mu_K, \sigma_K^2)$. Therefore the channel equations for the direct wireless communication are

$$Y_B = KX_1 + Z_1$$
$$Y_A = KX_2 + Z_2,$$

where we assumed that Alice transmits the signal X_1 which Bob receives as Y_B and Bob sends X_2, which Alice receives as Y_A. As we have mentioned before, there is a public noiseless channel with Φ and Ψ available. For the operation of the channel, we look at t time instances. There are n rounds of wireless communication, where in each round i Alice (Bob) sends a codeword $X_1(\omega_A, i)$ ($X_2(\omega_B, i)$) over the channel. We denote $X_1^n = (X_1(1), \cdots, X_1(n))$ and $X_2^n = (X_2(1), \cdots, X_2(n))$. After n communication rounds, the public channel may be used $k = t - n$ times, where $n \leq t$. Moreover, let f_A and f_B denote the key generation functions at Alice and Bob, respectively. The generated keys for Alice and Bob are therefore $S_A = f_A(X_1^n, Y_A^n, \Phi^k)$ and $S_B = f_B(X_2^n, Y_B^n, \Psi^k)$, respectively.

As in [4] we define an achievable key rate R_{key} if for every $\epsilon > 0$ and sufficiently large n there exists a strategy such that S_A and S_B satisfy

$$\Pr\{S_A \neq S_B\} < \epsilon, \tag{1}$$

$$\tfrac{1}{n} I(\Phi^k, \Psi^k; S_A) < \epsilon, \tag{2}$$

$$\tfrac{1}{n} H(S_A) > R_{key} - \epsilon, \tag{3}$$

$$\tfrac{1}{n} \log |S_A| < \tfrac{1}{n} H(S_A) + \epsilon, \tag{4}$$

where $|S_A|$ denotes the alphabet size of the discrete key random variable S_A.

Under these conditions it was shown in [4] that if both terminals observe correlated source outputs X^n and Y^n from a discrete memoryless multiple source with generic sources (X, Y), a secrecy key rate of $I(X;Y)$ can be achieved. The proof uses only a single forward or backward transmission of the public channel along with an extended Slepian-Wolf coding scheme. Originally proved for discrete sources, this result can be extended to continuous sources as well [5], [6].

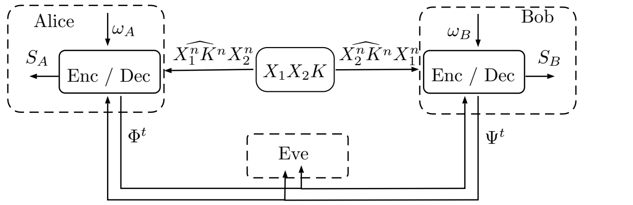

Figure 2. Illustration system model after the transformation with the proposed scheme.

III. A Novel Key Exchange Scheme

For simplicity suppose the channel noise is zero, then the channel equations would be

$$Y_B = KX_1$$
$$Y_A = KX_2$$

assuming reciprocity and therefore a fading gain K in both directions. Assuming we have a one-shot strategy with $n = 1$, the objective is now to choose good key generation functions f_A and f_B. The natural choice is to multiply the observation with the own signal and we would therefore have $S_A = X_1 Y_A$ and $S_B = X_2 Y_B$. Since we do not have any noise and perfect reciprocity, we see that $S_A = S_B = K_1 X_1 X_2$. Thus, we got a key for both Alice and Bob even without using the public channel.

Lets look at the general model with n rounds of communication, where we have the following equations

$$Y_B = KX_1 + Z_1$$
$$Y_A = KX_2 + Z_2,$$

which are without the side-information at Eve. The idea is now, that we do not need to estimate the fading gain via channel probing, but we can estimate the *product* of the channel input and the fading gain. Therefore, Bob estimates $\widehat{KX_1}$ and Alice estimates $\widehat{KX_2}$. After n steps, Alice and Bob have obtained the vector

$$\widehat{K^n X_2^n} = (K(1)X_2(1), \ldots, K(n)X_2(n)) \qquad \text{and}$$
$$\widehat{K^n X_1^n} = (K(1)X_1(1), \ldots, K(n)X_1(n)),$$

respectively. We now apply element-wise multiplication with the local randomness symbol-vectors X_1^n and X_2^n to obtain $\widehat{K^n X_2^n X_1^n}$ and $\widehat{K^n X_1^n X_2^n}$ at Alice and Bob, respectively. Now, Alice and Bob have correlated observations of the product of all three randomness source (see Fig. 2). We therefore have transformed our model into the related form of [4] and one can show that a key rate of

$$R_{Key} = I(Y_A X_1; Y_B X_2) \tag{5}$$

can be reached. In the next section we will provide a deeper analysis of this key rate.

IV. Information Theoretic Analysis

The proposed key exchange scheme in this paper works with a multiplication operation between the received signals and the own local randomness sources. In fact, any commutative operation could yield a working scheme as long as it corresponds to the respective channel operation between channel gain and send signal. From an information theoretical point of view it would be interesting to get insights into a general case, prior to any commutative combination of signals. Assuming that Alice receives $Y_A = X_2 K + Z_2$ she estimates the product $X_2 K$ and Bob receives $Y_B = X_1 K + Z_1$ and estimates $X_1 K$. This means that the problem can be cast in a source model, where Alice observes the pair (X_1, Y_A) and Bob observes the pair (X_2, Y_B). It is therefore insightful to look at the mutual information between Alice and Bob's observations $\frac{1}{2} I(Y_A, X_1; Y_B, X_2)$, where the factor of one-half is accounting for the two communication rounds Alice-to-Bob and Bob-to-Alice. One can show that this mutual information can be separated into the three terms

$$\begin{aligned}&\tfrac{1}{2} I(Y_A, X_1; Y_B, X_2)\\ &= \tfrac{1}{2} I(X_1; Y_B) + \tfrac{1}{2} I(Y_A; X_2) + \tfrac{1}{2} I(Y_A; Y_B | X_2, X_1),\end{aligned}$$

where the first two terms can be lower bounded by techniques from non-coherent fading channels and the last term by small variations to the standard key generation rate for pilot channel gain measurements. This leads to the following theorem.

Theorem 1. *For general observations $Y_A = K X_2 + Z_2$ and $Y_B = K X_1 + Z_1$, of the products $K X_2$ and $K X_1$, with input $X_1, X_2 \sim \mathcal{N}(\mu_X, P)$, channel gain $K \sim \mathcal{N}(\mu_K, \sigma_K^2)$ and noise $Z_1, Z_2 \sim \mathcal{N}(0, \sigma_Z^2)$ it holds that*

$$\begin{aligned}&\tfrac{1}{2} I(Y_A, X_1; Y_B, X_2)\\ &\geq \tfrac{1}{2} E_K[\log(1 + \tfrac{|k|^2 P}{\sigma_Z^2})]\\ &\quad - \tfrac{1}{4} E_{X_1}[\log(1 + \tfrac{|x_1|^2 \sigma_K^2}{\sigma_Z^2})] - \tfrac{1}{4} E_{X_2}[\log(1 + \tfrac{|x_2|^2 \sigma_K^2}{\sigma_Z^2})]\\ &\quad + \tfrac{1}{4} E_{X_1, X_2}\left[\log\left(1 + \frac{x_1^2 x_2^2 \sigma_K^4}{(x_1^2 + x_2^2)\sigma_K^2 \sigma_Z^2 + \sigma_Z^4}\right)\right].\end{aligned}$$

Moreover, we can directly give a result for the case of a constant channel gain with $\sigma_K^2 = 0$, which yields

$$\tfrac{1}{2} I(Y_A, X_1; Y_B, X_2)|_{K=\text{const}} \geq \tfrac{1}{2} \log(1 + \tfrac{|k|^2 P}{\sigma_Z^2}). \qquad (6)$$

This is in a way the key rate for the theoretical optimal model and thereby an upper bound on the product scheme. We can see that the key rate is split into contributions from the two local randomness sources X_1, X_2 and the channel gain K. The first three terms represent the rate of the local randomness sources and stem from a lower bound on the respective non-coherent fading channel rates of $I(Y_A; X_2)$ and $I(Y_B; X_1)$. These rates therefore represent the influence of the local randomness sources on the total key rate. The last term represents the rate from the channel variation itself. It is closely connected to the expression for the key rate of the state-of-the-art channel measurement method, without the factor $\frac{1}{T}$ which stems from channel probing over an coherence interval. Instead, we have a factor of one-half, since we need two communications, one from Alice and one from Bob, to generate a key. This shows that as long as a local randomness source can be exploited our scheme would achieve a rate bounded away from zero, scaling with the input power even in case that the channel gain is constant. This can be seen in (6), where for constant K the penalty terms (for not knowing the non-coherent channel gain) of the rate terms of X_1, X_2 as well as the channel gain contribution gets zero.

Remark. *In the product scheme we combine the pairs on both sides with a product operation. This results in a practical key generation scheme. However, it is important to note that the resulting key rate of this scheme is possibly lower than in the general set-up due to the data processing inequality.*

$$I(\widehat{KX_2}, X_1; \widehat{KX_1}, X_2) \geq I(\widehat{KX_2} X_1; \widehat{KX_1} X_2) \qquad (7)$$

The key generation function can reduce the entropy of the key.

V. Protocol Description

In this section we describe the *RECiP* protocol which enhances the *6doku* protocol [7]. Throughout this paper we attribute transmitted signals as PINGs and PONGs for Alice (A) and Bob (B) respectively. The protocol starts by creating three hash chains at each side to guarantee that Alice and Bob are indeed communicating with each other instead of their evil twins. For Alice, these are: $ping_l, \ldots, ping_0$, $auth_{A,m}, \ldots, auth_{A,0}$, $ack_{A,n}, \ldots, ack_{A,0}$ and for Bob's side: $pong_0, \ldots, pong_l$, $auth_{B,m}, \ldots, auth_{B,0}$, $ack_{B,n}, \ldots, ack_{B,0}$

Below, we elaborate the different phases of Physical-Layer (PHY) key generation.

1) Transmit Signal Generation: Let us denote the transmit signals (either PINGs or PONGs) by $X_A(\omega_i, i)$ and $X_B(\omega_i, i)$ where ω_i is the frequency slot index used for the i-th transmission. For any i the actual slot index ω_i is from the interval $\omega_i \in [11, 26]$. The following channel hopping pattern has been adopted:

$$\omega_i = ((\omega_{i-1} - 11 + 7) \mod 16) + 11 \qquad (8)$$

The above equation generate the sequence $\ldots, 26, 17, 24, 15, 22, 13, \ldots$. These numbers correspond to the 16 channels of 802.15.4 in the 2.4GHz band, which are indexed 11 through 26. Having selected the frequency slot, the transmit powers of $X_A(\omega_i, i), X_B(\omega_i, i)$ are independently fixed by Alice and Bob and are taken randomly from the set:

$$X_A(\omega_i, i), X_B(\omega_i, i) \in [0, -1, -3, -5, -7, -10, -15, -25] \qquad (9)$$

TelosB$[0, -1, -3, -5, -7, -10, -15, -25]$ refers to the available 8 power levels of our chosen prototyping platform: y. These power levels are set by internal registers $R_g = \{31, 27, 23, 19, 15, 11, 7, 3\}$ accordingly. For the implementation, we formulated the below method so that the transmitter can randomly select a power level for each PING-PONG.

Let us consider that the probability, $\mathcal{P}$ of choosing a value from the set R_g is uniformly distributed, $\mathcal{U}\lceil 0, 1\rceil$. Transmitter of Alice and Bob adopts the following power hopping scheme:

$$\forall_i \epsilon \mathbb{N} : R_g \to 4 \times \lceil 8 \times R_i \rceil - 1 \tag{10}$$

2) Sampling: A and B exchange l PING-PONGs. The i-th PING and PONG has $ping_i$ and $pong_i$ appended, respectively. As B receives the i-th PING, B checks whether $h^{i-\gamma}(ping_i) = ping_\gamma$, where γ is the index of the last accepted PING or 0 if no PING was accepted, yet. If true, B immediately replies with a PONG. Likewise, A verifies that $h^{i-\gamma}(pong_i) = pong_\gamma$. If so, A immediately sends the next PING and so on. A timer has been set to avoid dead lock due to the missing PING-PONGs. The RSSI values of the accepted PINGs and PONGs are handed over to the quantization phase.

3) Quantization: Now, let's denote the set of RSSI value, r at Alice end by $RSSI_{A,1}, \ldots, RSSI_{A,r}$ given by

$$RSSI_{A,i} = X_B(\omega_i, i) H_{BA}(\omega_i, i) + N_A(\omega_i, i),$$

where $H_{BA}(\omega_i, i)$ is the (reciprocal) channel from $B \to A$ and $N(\omega_i, i)$ is the interference. A multiplies $RSSI_{A,i}$ with its own $X_A(\omega_i, i)$ and centers its RSSI around zero by subtracting their mean $\overline{RSSI_A}$:

$$RSSI'_{A,i} = RSSI_{A,i} X_A(\omega_i, i) - \overline{RSSI_A X_A(\omega_i, i)} \tag{11}$$

Subsequently, A quantizes its centered RSSI as follows. Let $t \in \mathbb{R}_{>0}$ and:

$$Q_t = \{\ldots, -3t, -2t, -t, 0, t, 2t, 3t, \ldots\} \tag{12}$$

For $i = 1, \ldots, r$, A maps $RSSI'_{A,i}$ to the nearest value $q \in Q_t$:

$$RSSI''_{A,i} = \underset{q \in Q_t}{\arg\min} |q - RSSI'_{A,i}| \tag{13}$$

If $\arg\min$ returns two quantization levels, smallest value is chosen.

4) Reconciliation: After having quantized r RSSI, A sends reconciliation messages to B, containing the indices of missed PONGs, as well as $\delta_1, \ldots, \delta_r$. Thereby,

$$\delta_i = RSSI''_{A,i} - RSSI'_{A,i} \tag{14}$$

is the offset between $RSSI''_{A,i}$ and $RSSI'_{A,i}$.

Upon receipt, B retains the r Received-Signal-Strength-Identifications (RSSIs) of PINGs where A has received a corresponding PONG. We denote B's retained RSSIs by $RSSI_{B,1}, \ldots, RSSI_{B,r}$ given by

$$RSSI_{B,i} = X_A(\omega_i, i) H_{AB}(\omega_i, i) + N_B(\omega_i, i),$$

where $H_{AB}(\omega_i, i)$ is the channel from $A \to B$; for the sake of exposition let us assume channel reciprocity such that $H_{BA}(\omega_i, i) = H_{AB}(\omega_i, i)$. Now, B multiplies $RSSI_{B,i}$ again with its own $X_B(\omega_i, i)$ and centers its retained RSSIs around zero by subtracting their mean:

$$RSSI'_{B,i} = RSSI_{B,i} X_B(\omega_i, i) - \overline{RSSI_B X_B(\omega_i, i)} \tag{15}$$

Finally B quantizes them, taking into account the reconciliation information $\delta_1, \ldots, \delta_r$:

$$RSSI''_{B,i} = \underset{q \in Q_t}{\arg\min} |q - (RSSI'_{B,i} + \delta_i)| \tag{16}$$

As long as the maximum discrepancy between A's and B's centered RSSI trajectories is smaller than $\frac{t}{2}$, reconciliation outputs equal trajectories $RSSI''_{A,1}, \ldots, RSSI''_{A,r}$ and $RSSI''_{B,1}, \ldots, RSSI''_{B,r}$. The quantized and reconciled RSSI trajectories $RSSI''_{A,1}, \ldots, RSSI''_{A,r}$ and $RSSI''_{B,1}, \ldots, RSSI''_{B,r}$ are input to the privacy amplification phase.

5) Privacy Amplification: For randomness extractor, we adopted CBC-MAC. It works as follows. Let $x_1, \ldots, x_t$ be 16-byte blocks of a quantized and reconciled RSSI trajectory, potentially padded with zeroes. Given this input and a publicly known 128-bit key k_{pub}, A and B generate a chain of blocks $\bar{x}_0, \ldots, \bar{x}_t$ and use $\bar{x}_t$ as PHY key:

$$\bar{x}_0 = 0 \tag{17}$$

$$\bar{x}_i = \text{AES}(k_{pub}, x_i \oplus \bar{x}_{i-1}) \text{ for } i = 1, \ldots, t \tag{18}$$

We use $k_{A,phy}$ and $k_{B,phy}$ to denote the generated PHY key of Alice and Bob respectively.

VI. Implementation

This section is devoted to the implementation details of our proposed method. We extended the work of [7]. In similar fashion, we have an Android device connected with a TelosB mote via USB host cable. Currently, smart-phones or tablets available in the market are not equipped with built in IEEE 802.15.4 supported wireless chip. Hence, the smart phone with the connected TelosB mote gives it a functionality of *Alice* for a 6LoWPAN network. Likewise, other motes(s) depict as *Bob*.

A. Software Platform

The chosen hardware board, TelosB needs to be programmed using some operating systems that supports 6LoWPAN protocol stack. There are multiple options available such as TinyOS, Contiki, MINIX, different version of RTOS. We opted in for Contiki OS since the existing work was primarily done with it. At the same time it has well supported security features, one of which was done by Felix [8]. For our project we have used Contiki 2.7. Additionally, the whole development has been done on Linux Ubuntu desktop OS 14.04 LTS. For developing the *Android* app, we used Android Studio. Simultaneously, we also needed *Android* running device, we have used Google Nexus 7 2012 edition with Lollipop 5.1.1.

B. Hardware Platform

We chose TelosB platform because it is highly integrated and widely used platform for IoT scenarios. The advantages of using this board are that (1) it has hardware accelerated AES-128 encryption module (2) it has built in CC2420 Chipcon made 802.15.4 wireless chip with programmable transmission power. (3) It is relatively cheaper and smaller in size and comes with standard USB port for debugging, and programming. (4) The hardware platform has well documented programming API and developer support.

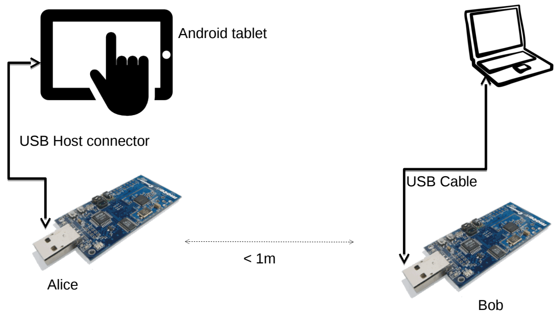

Figure 3. Evaluation setup in an indoor environment

C. Transmission Power Control

In our proposed method, controlling transmission power of the communicating devices serves two major purposes. In the first case, it limits the range of the authenticating devices, hence, together with the time-out constraints it can mitigate *Impersonation Attacks* from remote devices. Even though there is no defined precise boundary of such range control but the platform allows developers to control transmission power to limit sensitivity within few centimetres. The second aspect of controlling transmission power is that it enhances the entropy of the generated key by employing additional intrinsic randomness in the channel. Essentially in this case, we make use of different power levels available on the chip.

Ideally, any latest wireless transceiver is capable of transmitting at multiple level of power levels. Our used motes is equipped with Chipcon CC2420 wireless chipset. The onboard Texas Instruments MSP430 micro-controller allows programmable output power. As it is specified in its datasheet [9], there are 8 programmable power levels to transmit at $0, -1, -3, -5, -7, -10, -15, -25$ dBm. These power levels are set by internal registers $R_g = 31, 27, 23, 19, 15, 11, 7, 3$ accordingly. In the previous section we have already described the mathematical formula for the randomisation of the transmission power.

D. Evaluation

To demonstrate our proposed method, we set up testing environment with two TelosB motes in a indoor environment where they are placed together in less than 1m distance, as depicted in Fig. 3. In the work of [7], the minimum working environment was 1m, with our implementation this has been significantly reduced to one third of a meter.

In Fig. 5, one can notice that the RSSI values for Alice (in Blue) and Bob(in Red) significantly correlate compared to the case in Fig.4 where reciprocity principle is not holding due to power randomization. Hence, with the restored set of RSSI values at both Alice and Bob, we can generate symmetric PHY key.

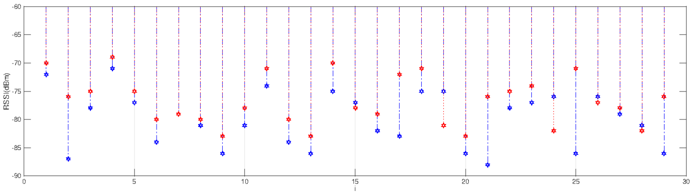

Figure 4. Uncorrelated RSSIs (in Red and Blue dots) for two closely located (<1m) nodes: Alice and Bob

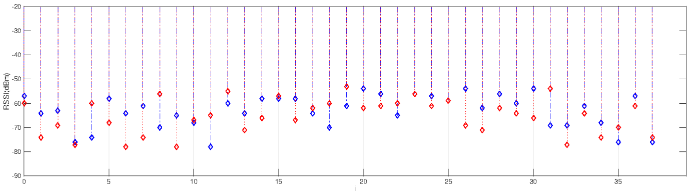

Figure 5. RSSIs are highly correlated after transmission power randomization for two closely located (<1m) nodes: Alice and Bob

VII. Conclusions

In this paper we introduced the RECiP protocol which allows to modify the transmission powers on the fly. Hence, local randomness can be exploited in this scheme which can information-theoretically increase the possible key rates of channel-based secret key generation schemes. We verified the practicability of the scheme with measurements which indeed exhibits improved robustness particularly in scenarios where Alice and Bob are in close proximity. We mention that further analysis and verifications of the scheme is needed and left for future work when an eavesdroper is also present.

References

[1] A. Mukherjee, "Physical-layer security in the internet of things: Sensing and communication confidentiality under resource constraints," *Proceedings of the IEEE*, vol. 103, no. 10, pp. 1747–1761, Oct 2015.

[2] C. T. Zenger, M.-J. Chur, J.-F. Posielek, C. Paar, and G. Wunder, "A novel key generating architecture for wireless low-resource devices," in *Secure Internet of Things (SIoT), 2014 International Workshop on Secure Internet of Things (SIoT 2014), September 2013, Wroclaw (Poland), published by Springer Lecture Notes in Computer Science (LNCS) series and by IEEE Explore.* IEEE, 2014, pp. 26–34.

[3] J. E. Hershey, A. A. Hassan, and R. Yarlagadda, "Unconventional cryptographic keying variable management," *Communications, IEEE Transactions on*, vol. 43, no. 1, pp. 3–6, 1995.

[4] R. Ahlswede and I. Csiszar, "Common randomness in information theory and cryptography. i. secret sharing," *IEEE Transactions on Information Theory*, vol. 39, no. 4, pp. 1121–1132, Jul 1993.

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

[6] S. Nitinawarat, "Secret key generation for correlated gaussian sources," in *IEEE International Symposium on Information Theory*, July 2008, pp. 702–706.

[7] K. F. Krentz and G. Wunder, "6doku: Towards secure over-the-air preloading of 6lowpan nodes using phy key generation," in *Smart SysTech 2015; European Conference on Smart Objects, Systems and Technologies*, July 2015, pp. 1–11.

[8] K. Krentz and G.Wunder, "6LoWPAN Security: Avoiding Hidden Wormholes using Channel Reciprocity," in *ACM Int. Workshop on Trustworthy Embedded Devices*. Scottsdale, USA: ACM, November 2014.

[9] M. Corporaton, "Tmote sky: Datasheet," 2006.