Exploiting multiple side channels for secret key agreement in Wireless Networks

Hailun Tan, Sanjay Jha
School of CSE, UNSW, Sydney, Australia
Email: (thailun,sanjay)@cse.unsw.edu.au

Vijay Sivaraman
School of EE UNSW, Sydney, Australia
Email: vijay@unsw.edu.au

Diet Ostry
Data 61, CSIRO, Sydney,Australia
Email: Diet.Ostry@csiro.au

Technical Report
UNSW-CSE-TR-201615
September 2016

School of Computer Science and Engineering
The University of New South Wales
Sydney 2052, Australia

Abstract

Generating a secret key between two wireless devices without any priori information is a challenging problem. Extracting the shared secret from a wireless fading channel is proven as an effective solution to this problem. However, the unreliable wireless channel results in a significant communication overhead. Most of the related works focus on minimizing the impact of channel unreliability in the key agreement process. In this paper, we explore another direction, multiple side channels, to establish the shared key. One of the side channels is packet transmission power. By switching among multiple transmission power levels, the receiver is able to decode the bits by comparing the Received Signal Strength of the current packet with that of the previous one. However, a side channel of transmission power changes alone is not secure enough as adversary could intercept the packets and infer the transmission power change pattern. Therefore, we employed another side channel by swapping the source and Destination address of the packets. We showed that adversary is able to extract shared bit with only one of the these side channels deployed but it cannot when both side channels are utilized. We showed that our approach could establish the N-bit shared key with $O(N)$ packets.

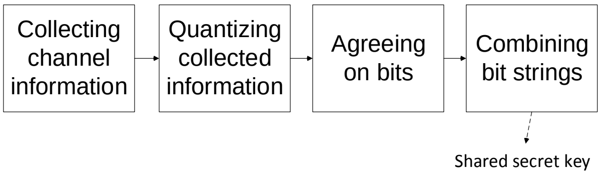

Collecting channel information
Quantizing collected information
Agreeing on bits
Combining bit strings
Shared secret key

Figure 1.1: four phases in key agreement protocol based on wireless reciprocal channel.

1 Introduction

Traditional security schemes rely on cryptographic keys to support various security services, including authentication, confidentiality, and integrity. With the increasing popularity of wireless security, key agreement in wireless networks becomes more important. For example, in a dynamic environment, mobile devices need to form their shared secret in an ad-hoc way. A Certificate Authority (CA) or a centralized key management server might not be available. Therefore, it is necessary to have alternative approach for key establishment between wireless entities without relying on a fixed infrastructure.

There has been an increasing interest in a key agreement among the wireless devices by exploiting the reciprocal property of wireless fading channel [9, 17]. The bidirectional channel state would be identical between two transceivers at a given instant due to channel reciprocity. Therefore, those two wireless transceivers are able to establish the shared secret from their identical reciprocal channel state. On the other hand, the channel state observed by the eavesdropper would not be the same if the eavesdropper is more than half a wavelength away from legitimate wireless parties [14]. All these related key agreement protocols would have the following four phases, shown in Figure 1.1.

Both wireless communication parties would need to probe the channel to collect enough channel information so that they can further quantize the collected information for channel characteristics model. After this model is established, both parties would setup some threshold range to define the value of each bit according to the comparison between the respective channel values of the probing packets and the threshold range. After the enough number of bits (e.g., 128 bits for an AES key) are collected, the bit string can be combined and created.

Although the key agreement based on reciprocal channel is well-suited for the wireless devices, it has the following limitations:

- *Low bit generation rate:* The related experimental work showed two wireless devices can generate a shared key at approximately 1 bit/sec by using off-the-shelf 802.11a hardware [9]. Advanced Encryption Standard (AES) needs a key whose length is at least 128 bits. It means this approach takes more than two minutes to establish one AES key. The low bit generation rate is not realistic for wireless networks, where the communication is intermittent. Some subsequent works tried to improve the bit generation rate by deploying multiple antennas [16,24], multiple fading channels [22] or Multiple Input Multiple Output (MIMO) [20] to accelerate key generation process. However, the bit generation rate is fundamentally constrained by time-variation of the channel(s). Even both the wireless transceivers attempt to probe the channel(s) with multiple antennas, it could not extract more randomness in a shorter period of time for key generations. In addition, not all the wireless devices would have multiple antenna installed or be able to probe multiple channels.

- *Significant communication overhead:* In order to retrieve the random bits, both wireless devices need to send and receive a number of probing packets. Both of them would setup a related threshold range in terms of channel characteristics. This range can filter out those probing packets whose value does not yield the required randomness for bits construction. For example, in RSSI-based approaches [8, 11–13, 17], the key agreement scheme would have an RSSI range. If the RSSI of the received probing packet is within this range, they would be ignored. Only those whose RSSI values are beyond this range would be used for bit extractions. Therefore, many probing packets would be wasted if the RSSI values do not fall beyond this threshold range. Some subsequent works adopted other channel characteristics such as Channel State Information (CSI) [21], rather than RSSI, to extract the shared secret bits and claimed that communication overhead can be reduced with more flutuations than RSSIs. However, to be the best of our knowledge, no existing works can provide the upper limit for the number of probing packets required to establish an N-bit key given the unreliability of wireless fading channel state.

- *Inconsistencies of wireless fading channels on both communication parties:* The channel characteristics are correlated but not exactly the same even for both legitimate communication parties. Therefore, the threshold range setup for both communication parties could have some minor differences (i.e., phase 2 in Figure 1.1). The probing packets whose corresponding channel characteristics are in the gap within these threshold range differences, would be filtered. Otherwise, it would cause the mismatch on the shared bits. How to optimize the different threshold range due to wireless channel unreliability poses another challenge for these protocols.

In this paper, we explore a completely different direction to overcome the said limitations of the key agreement. We adopt multiple *side channels* to exchange the secret bits between two legitimate wireless devices. In computer security, a side channel is a way to gain from the physical implementation of a cryptosystem, rather than brute-force or theoretical weaknesses in the algorithms [1]. A side channel is virtually undetectable by administrators or users and is used to steal the information from a highly secure system. We exploit the undetectable characteristics of side channel to hide the exchanged bits from external parties so that an adversary is not able to detect these exchanged bits. However, we still adopt the reciprocal channel to authenticate the received packets to defend against the Man-In-Middle (MIM) attack. Our contributions are as follows:

- *A key agreement protocol based on multiple side channels and reciprocal channel:* Different from the existing works, our key agreement protocol extracts the bit of shared key from side channels and relies on the reciprocal channel to authenticate the source of the packets. Though a side channel has been used in "Shake them up!" [5], it had some limitations, which would be discussed in Section 2 which our protocol can address. To the best of our knowledge, we are the first to propose a key agreement associated with multiple side channels.

- *Optimization on communication overhead for key agreement exploiting the wireless channel characteristics:* both the legitimate wireless devices can reach a N-bit keys in $O(N)$ using the side channel in our protocol. The communication overhead can be significantly reduced.

- *Independence to the channel characteristics when extracting the agreed bits:* our protocol employs the channel characteristics to authenticate the sender of packets but we do not rely on the channel characteristics to retrieve the agreed bits (phase 3 in Figure 1.1). Therefore, we do need to setup the related threshold range, which is subject to unreliability of the wireless channel conditions.

The rest of paper is organized as follows: The related works is surveyed in Section 2. The system and threat model is defined in Section 3. Our key agreement protocol is detailed in Section 4 with the potential attack analyzed in Section 5. The related communication overhead is analyzed in Section 6.2. Our protocol is concluded in Section 7.

2 Related Works

There has been active research in secret key agreement through exploiting the wireless channel randomness and principle of reciprocity [3, 9, 17, 19, 23] . The existing works focused on exploiting the temporal and spatial variations of the radio channel [8, 11–13, 17]. However, these RSSI-based approaches are subject to the flutuations of wireless fading channel condition. The probing packets in these schemes would be filtered if their RSSI values are not beyond the pre-set threshold range. Therefore, all these filtered probing packets are wasted, resulting in battery drains on the wireless devices, which could be power-limited (e.g., wireless sensors). The subsequent works employed the different channel characteristics to extract the shared bits such as CSI [21] or phase reciprocity of frequency selective fading channels [7]. other researchers have suggested deploying the multiple antennas [16, 24], multiple fading channels [22], multi-path [3] or for probing packets to improve the performance in phase 1- 3 in Figure 1.1. However, the limitations mentioned in Section 1 have not been fully resolved.

Castelluccia et al. proposed a first key agreement protocol, "Shake them Up!" [5], based on the swap of source address in packets for wireless devices . It was the first attempt to rely on a secret channel (i.e., the true/false source address to indicate 1/0 for the bit, respectively) to establish the shared key. Though it was easy to implement and efficient, it required the communication parties to be as close as 1-2 cm to initialize the key agreement process, which is not practical for many applications. In addition, both wireless devices need to be held by the user and one of them is "shaken" to generate a constant moving pattern and signal power for other device to recognize it. This assumption does not hold when the wireless devices are not readily accessible to user (e.g., sensor devices monitoring in the hostile area). Last by not least, "Shake them up!" protocol is subject to received signal Strength attack. An adversary can collect a sufficient number of exchange packets and their RSSIs, associated with its physical distances to both communication parties to infer which packet contains the true source address (i.e., shared bit as "1") and the false address (i.e., shared bit as "0"), respectively. Therefore, Castelluccia et al. suggested that those two devices need to be held very close to each other to obfuscate the signal Strength analysis [5], which is not practical for the devices after deployment. Another work improved the secrecy of this process by introducing a designated mobility pattern for spatial obfuscation [10]. However, it failed to fully address the received Signal Strength analysis.

In this paper, we further enhance the security of "Shake them up!" by inviting the transmission power side channel and overcome these said limitations of the protocol. In our protocol, we do not require both wireless devices to be as close as 1-2 cm to establish the shared key [5] or follow the pre-set moving pattern to defend against received signal Strength analysis [10].

3 System & Threat Model

In our system, we assume that both wireless devices do not share any pre-distributed information to establish the shared key beforehand. However, they would have agreed how to extract the shared bits from the exchanged packets and adversary would know the rules of extracting the bits as well. It is the basic assumption for a "protocol" to operate. They are static and within a one-hop transmission range when they are in the key agreement process. Both legitimate wireless devices are able to change their transmission power per packet basis. We also assume that the changes of transmission power is so significant that Receiving Signal Strength Indicator (RSSI) variations would completely reflect the same transmission power changes in the presence of wireless noise (i.e., if the transmission power of the sender increases, so is the RSSI of the received packet and vice versa). In addition, we assume that the antennas of the legitimate communication parties are omni-directional, which means the directions of the wireless antenna of the recipient would not affect RSSI. We would further evaluate how this assumption can hold in Section 6.1.

The goal of adversary in our key agreement process is to learn about the shared key by intercepting the communications without being noticed. Therefore, we assume that adversary is able to intercept all data packets between both communicating parties passively. In addition, we also assume that adversary could be in the ambient environment, injecting the packets to disrupt the key agreement process. However, we assume that adversary is not within the range of half of the wavelength to the legitimate wireless devices, Therefore, adversary is not close enough to mimic the same channel condition as that between the legitimate wireless devices. We also do not consider the Denial of Service (DoS) as it can readily expose the existence of the stealthy adversary. Last but not least, we assume that the parties who wish to establish the shared key are not subject to node compromise during the entire key agreement process.

In our protocol we assume that the communication channel is authenticated through RF finger printing [4, 15, 16]. Therefore, adversary (e.g., Eva) should not be able to impersonate as Alice or Bob to communicate with the other legitimate party. We will also discuss this type of Man-In-Middle Attack in Section 5 when this assumption is relaxed.

4 Design & Implementations

We assume that Alice wish to communicate with Bob. In our design, Alice would initiate the key agreement protocol. On the other hand, Bob would acknowledge the packets from Alice and establish the shared secret according to Alice's specified configurations on source/destination address and RSSI of the packet.

Our key agreement protocol consists the following phases

- Initialization

- Secret Bit extractions
- Shared key establishment and re-keying process

Table 4.1 shows the notations used in this document.

Table 4.1: The notations in Section 4.

notation	meaning
K_{A/B_i}	the i^{th} shared key bit
$P_{\mathrm{A/B}_i}$	the i^{th} bit to determine the power Level change "1" means the packet is transmitted in increasing power "0" denotes the packet is transmitted in decreasing power.
$SD_{\mathrm{A/B}_i}$	the i^{th} bit to determine whether the addresses are swapped "1" means the source/destination pair is not swapped "0" denotes it is swapped

4.1 Initialization

At this stage, Alice and Bob are able to communicate with each other within one-hop distances and remain static. They exchange a number of probing packets with each other to sample the channel characteristics with various transmission power levels. According to the Recieved Signal Strength Indicator (RSSI), Alice and Bob both specify multiple Levels of transmission powers and perform the channel sampling given different transmission powers. Please note that the levels of transmission powers for Alice are not necessarily the same as Bob. However, these transmission power levels should be set distinctively so that the receiver is able to interpret the transmission power changes with the presence of noise. In Section 6.1, we would further investigate how long it takes to reach a stable channel state with hardware experiments.

4.2 Secret Bit extraction

The bit extraction process is illustrated in Figure 4.1.

After both the communication parties are able to authenticate each other's packets based on the wireless channel reciprocity with various levels of transmission power, Alice would send out a "start" message to Bob with a default transmission power level (e.g., Power level 2 in Figure 4.1). Upon receipt the "start" message from Alice, Bob would reset his power level to default, record the receiving power level of Alice's "start" message and send the "start" message back to Alice. Alice would record the receiving power level of Bob's response.

Please note that in order to obfuscate this "start" process, Alice or Bob could choose to swap the actual source and destination address (i.e., Alice sends the "Start" message with the source address set to Bob or Bob sends the "Start" message back to Alice with source address as Alice). Only Alice and Bob would know who sends the respective packets.

After the exchange of "Start" message, Alice would generate a random bit for transmission power (P_{A_i}) and another random bit for source/destination behaviour (SD_{A_i}) for the i^{th} shared key bit. The i^{th} key bit from Alice and Bob (K_{A_i} and K_{B_i}) is the exclusive OR result of the above two bits:

$$K_{A_i} = P_{A_i} \oplus SD_{A_i}$$

$$K_{B_i} = P_{B_i} \oplus SD_{B_i}$$

Alice would adjust the transmission power level according to P_{A_i}. If P_{A_i} is 1, Alice would increase the transmission power for the i^{th} packet, compared to that of $i-1^{th}$ packet. Otherwise, Alice would decrease the transmission power. For the first bit extraction (i.e., $i = 1$), Alice would refer to transmission power level of the "Start" message for power adjustment. Alice set the Source/Destination (S/D) address of the packet according to SD_{A_i}. If SD_{A_i} is 1, Alice would set the source address as "Alice" and the destination address as "Bob". If SD_{A_i} is 0, Alice would set the source address as "Bob" and destination address as "Alice". Alice would send this key request with the sequence number with the set S/D in the packet with the adjusted transmission power.

Upon the receipt of Alice's request, Bob could retrieve P_{A_i} by comparing the RSSI of this packet and previous packet from Alice. If the RSSI becomes higher, P_{A_i} is 1. Otherwise, it is 0. In addition, Bob could tell whether the source/destination addresses in Alice's packet is true or false (i.e., 1 or 0 for SD_{A_i},respectively) as Bob knows whether he himself has sent this packet or not. Therefore, Bob is able to retrieve the intended i^{th} key bit (K_{A_i}). According to K_{A_i}, Bob could configure the transmission power and S/D addresses of its response. For example, in Figure 1.1, Bob has two combinations of P_{B_1} with SD_{B_1} given K_{B_1} as 1:

$$P_{B_1} = 1, SD_{B_1} = 0$$

$$P_{B_1} = 0, SD_{B_1} = 1$$

Please note that the transmission power levels could not be increased or decreased indefinitely. Therefore, if the transmission power has reached the upper limit in the previous packet, $P_{B_1} = 0, SD_{B_1} = 1$ is chosen to decrease the transmission power and vice versa. If the transmission power of the previous packet has not reach the limit, either option is fine.

Bob would send the response, containing the same sequence number, back to Alice with the specified configurations. Alice would decode the information from the comparison of RSSIs and S/D bit from the source/destination setting. If the extracted bit is the same as the one she set earlier, it means Alice and Bob has agreed on this bit. Alice would iterate this request-response process until N agreed bits are extracted. On the other hand, if the retrieval information does not match with Alice's key bit in this round, Alice would know that the secret bit extraction process is being poisoned. This scenario would be further discussed in Section 5. In order to defend against the temporal traffic analysis, Alice and Bob can swap the roles of the initiation of key extraction process. For example, Bob can send the request packet to Alice while Alice replies with the matching configurations on the transmission powers and the Source/Destination information so that both parties can agree on the subsequent bits.

After N shared key bits are extracted, Alice would send an "End" message to Bob and reset the transmission power level. Bob would respond with the "End" message to reset the transmission power to default level.

In our protocol, each secret bit is agreed on one request-response packet pair. Therefore, the communication overhead for secret bit extraction is $O(N)$.

4.3 Shared key establishment and re-keying process

Both Alice and Bob concatenate N secret bits from the phase of "secret bit extraction" to form a N bit shared key. Both Alice and Bob can exchange the hashed checksum of the N bit shared key and check if both checksums match. If they do, both parties have come to a shared secret key. Otherwise,

They could perform various bit-wise operations to further obfuscate the final shared key (e.g., reversal of the secret bits sequence). However, we assume that Alice and Bob do not share any priori information before they establish the shared key. Therefore, we do not expect Alice and Bob would have a protocol about how to perform the bit-wise operations for the final N-bit key beforehand.

Alice and Bob could resume the mobility after the key is established. If they need to perform the rekey process, they would need to be within one-hop distance and remain static. Then they perform the three phases (i.e., "initialization–>secret bit extraction –> shared key establishment") to establish the shared key as they could be in a different environment where the wireless channel conditions have been changed.

5 Security Analysis

In key agreement protocol, Adversary could either retrieve the secret bits passively or disrupt the key agreement process actively (Man-In-Middle Attack). Packet Interception is analyzed in Section 5.1 while the MIM attack is discussed in Section 5.2.

5.1 Packet Interception

We assume that adversary (e.g., Eva) captures all the packets exchanged between Alice and Bob. At the initialization stage, Alice and Bob exchange the probing packets to setup various distinct transmission power levels so that the respective RSSIs would follow the same variation of transmission power. Eva could not mimic the channel conditions between Alice and Bob despite capturing all the probing packets due to the reciprocity of wireless channel if Eva is located further than half of the wavelength from Alice/Bob [16]. At secret bit extraction stage, Eva could learn that the transmission power change

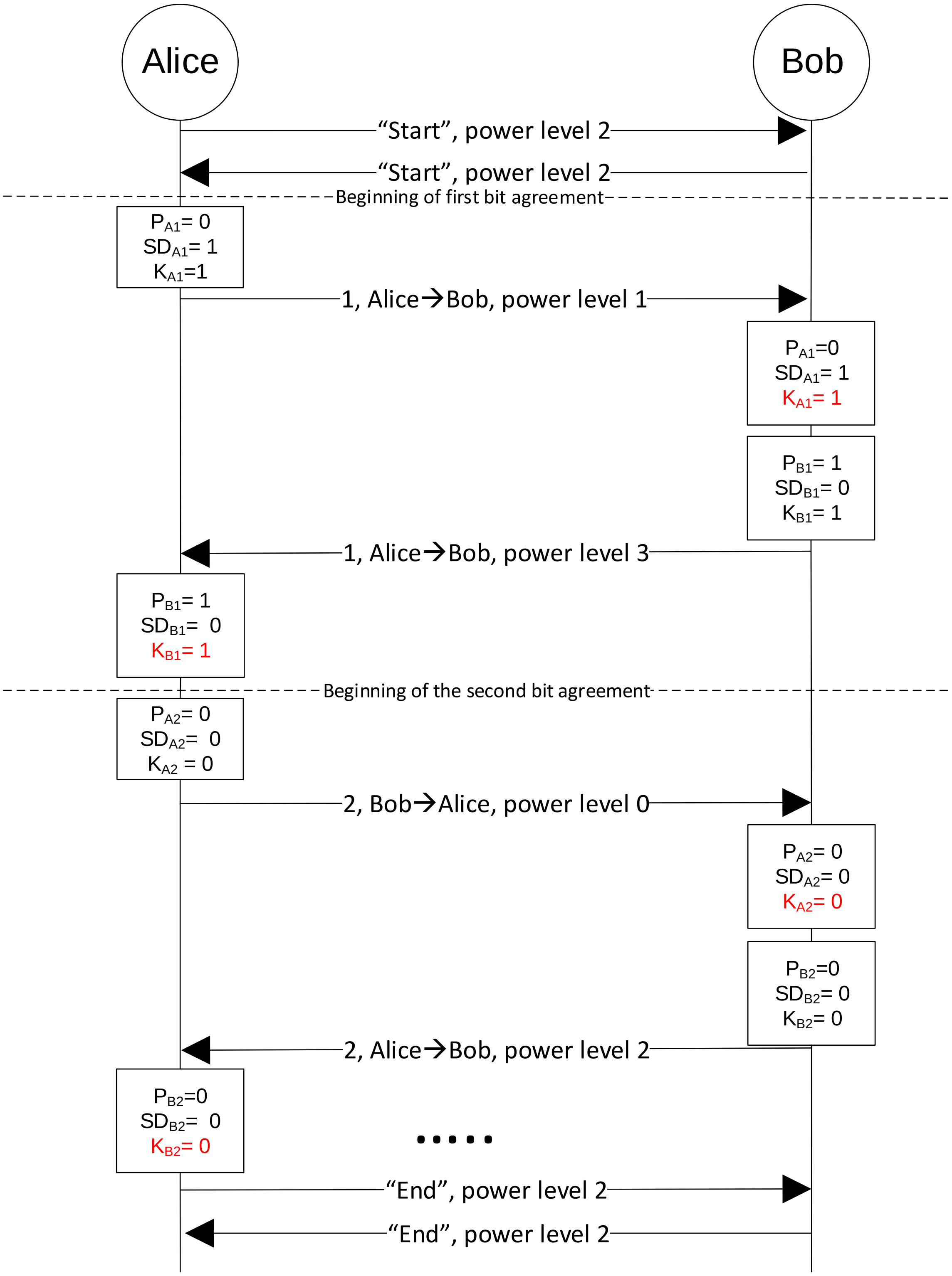


Figure 4.1: Multiple side channels key agreement process between Alice and Bob: $K_{A_i} = P_{A_i} \oplus SD_{A_i}$. Both parties come to the shared key bit in the step highlighted in red.

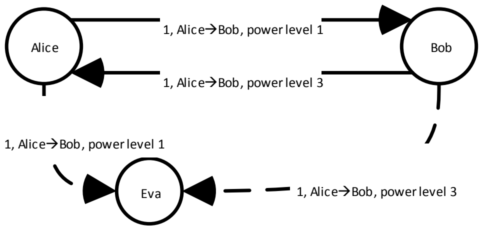

Alice
1, Alice→Bob, power level 1
Bob
1, Alice→Bob, power level 3
1, Alice→Bob, power level 1
Eva
1, Alice→Bob, power level 3

Figure 5.1: Packet Interception: Eva overhears all the packets between Alice and Bob. Eva is closer to Alice so RSSI of packets from Alice would be higher than those from Bob if the transmission power remains constant. A variable transmission power solves this security issue.

pattern by comparing the RSSIs for every two packets. In a request-response protocol, adversary can group all the requests and responses seperately for traffic analysis. In order to defend against this temporal traffic analysis, Alice and Bob can swap the roles of requester and responder for each extracted bit. For example, in Figure 4.1, Bob could send the request packet to Alice while Alice responds with the matched packet in the second bit extraction process. Therefore, the packet exchange pattern, "Every two packets are sent from the same wireless device", would not hold for Eva in this case.

As to the spatial traffic analysis, Eva is able to infer the actual source and destination of a packet based on the RSSI of the packets and its spatial distances to Alice and Bob (see Figure 5.1) if the transmission power is constant [5]. For example, in Figure 5.1, Eva is closer to Alice. The intercepted packets from Alice would have a higher RSSI values than those from Bob to Eva. After intercepting enough number of packets, Eva could tell if the S/D information is true or false by analyzing the respective RSSIs. In our protocol, the transmission power is changed per packet basis. In Figure 5.1, the transmission power from Alice is decreased while the transmission power from Bob is increased, which offsets the spatial distances effects on RSSIs of packets intercepted by Eva. Eva could not tell whether the S/D information is true or false any more.

The received signal Strength is determined by the following three independent factors if the antennas of Alice and Bob are omni-directional:

- Transmission Power: The higher transmission power the source sends the packets in, the stronger the received signal Strength the recipient would get if other factors remain unchanged.
- Distance between the packet source and the signal analyzer: The longer distance it is between source and signal analyzer, the lower the received signal Strength is given the same wireless channel and transmission power.
- Relative position of the wireless devices: When two wireless devices are very close, they could be obstacles for each other. For example, If Alice is in front of Bob within 1-2 cm, Packets from Bob to Alice would be received at much higher signal level while the received signal Strength of Bob's packets to other devices would be significantly reduced with Alice's presence.

In order to defend against the spatial traffic analysis, the third factor in the above list is required in "Shake-them-up" protocol [5]. As a result, the signal analyzer is not able to retrieve the true source of the packets as Alice and Bob are placed close enough to interfere with Eva's spatial traffic analysis. However, such requirement is not practical for wireless networks. Our protocol explored the first factor, transmission power, to obfuscate the spatial traffic analysis. The transmission power level changes serves as a second side channel to increase the entropy of the extracted key bits. Moreover, the key bit in our protocol is not directly transmitted in a side channel as "Shake them up!" [5], which is constraint to the distance between Alice and Bob.

As a result, although Eva is able to capture all the exchanged packets between Alice and Bob, she lacks the the information from S/D channel for key retrieval due to the change of transmission power.

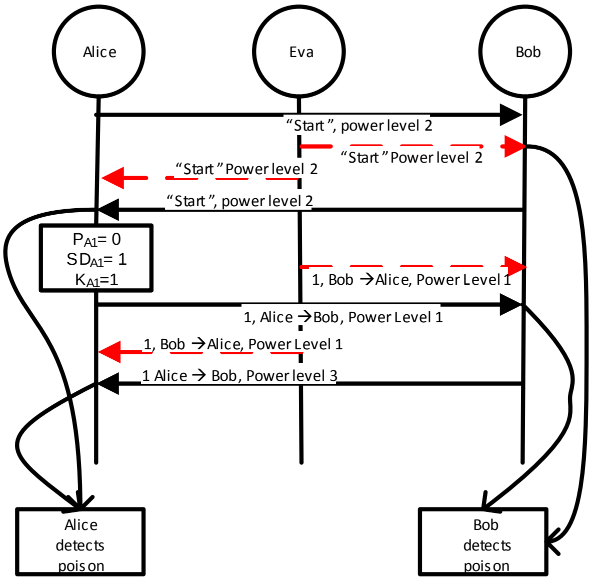


Figure 5.2: Man-In-Middle Attack: Eva could inject the arbitrary packets to Alice or Bob. But the repeated request and response would reveal the existence of adversary.

5.2 Man-In-Middle (MIM) Attack

In MIM attack, Eva attempts to impersonate as one of the legitimate communication parties and poison the key agreement process.

At initialization stage, Eva could inject a number of probing packets to disrupt the channel sampling process between Alice and Bob. However, due to channel reciprocity, the probing packets from Eva to Alice/Bob could only help extract the channel characteristics between Eva and Alice/Bob. Though Eva could cause the DoS by sending the exccessive probing packets, this DoS attack can be defeated by other anti-jamming techniques [6].

At Secret bit extraction stage, Eva could attempt to poison the process by injecting the arbitrary packets (see Figure 5.2) between Alice and Bob. In our protocol, each request would be paired with one response only (see Figure 5.2). Eva's packet injections would either make Bob receive multiple requests or Alice receive multiple responses. Whenever a repeated request or response is received, the recipient would know the key agreement process is being poisoned (see Figure 5.2). Unless Eva is able to block all the traffic from Alice or Bob, the Man-In-Middle attack would be immediately detected. In addition, if Eva's injected packet could not yield the same key bit because Eva is not able to retrieve the key bits from packet interception in Section 5.1. The mismatch of the key bits also indicates the key agreement is being poisoned. Both communication parties would restart the key agreement process to construct the rest of the shared bits if the packet injection from a third party is detected.

6 Performance Evaluation

6.1 Key Error Rate

In our scheme, each shared key bit is retrieved from two side channels: the Source/Destination channel and the transmission power channel. The source/desination channel is based on the true/false source/destination pair. Therefore, Alice and Bob would not interpret the bits wrongly as long as the packet was from the legitimate sender. On the other hand, for the transmission power channel, we assume that the Received Signal Strength (RSS) of a packet should follow the same variation trend as Transmission Power (TP) of it if both communication parties remain static with omni-directional antennas. As we have a TP side channel to exchange the bits between Alice and Bob, if the RSS did not change in the same way as the respective TP did, the packet sender and receiver would interpret the bit in different values from the transmission power side channel, which can result in a mismatch bits in the shared key. Though it can be detected in the Section 5.2 and the key agreement process would reset for the rest of the bits, the bits mismatch from the variations of TP and RSS mainly depends on three factors:

- *Variation of the distance between both communication parties:* If the distance of sender and receiver is different each time a packet is received, the variation of RSS would not only reflect the change of TP. In order to minimize such impact, we assume that Alice and Bob are static when they exchange the packets for key agreement. However, we would relax this assumption in this section to evaluate how the variation of distance could affect the key error rate in our scheme.
- *The transmission power values:* If TP is low, the recipient would receive the packets reluctantly and the measured RSS would not be accurate. On the other hand, if TP is unnecessarily strong, the measured RSS would be affected due to signal interferences. Therefore, we wish to retrieve the variation range of TP so that the key error rate is minimized.
- *The hopping gap of TP levels:* As is specified in Section 4.1, the sender should choose the distinct TP levels so that the respective RSS could be statistically different. Therefore, the bigger hopping gap there is between TP levels, the more distinct the RSS could be. However, the bigger TP level gap would lead to a smaller number of TP levels available for use. Therefore, we wish to find out the optimal TP hopping gap.

We employed the MicaZ motes from Crossbow Technologies [18] in our experiments. The MicaZ mote operates in the 2.4 GHz frequency band, and can support a 250 Kbps data rate. It supports 32 RF output power levels at run-time via a register. The experimental settings in terms of the specified three factors (i.e., distances, TP values and TP level hopping gaps) are:

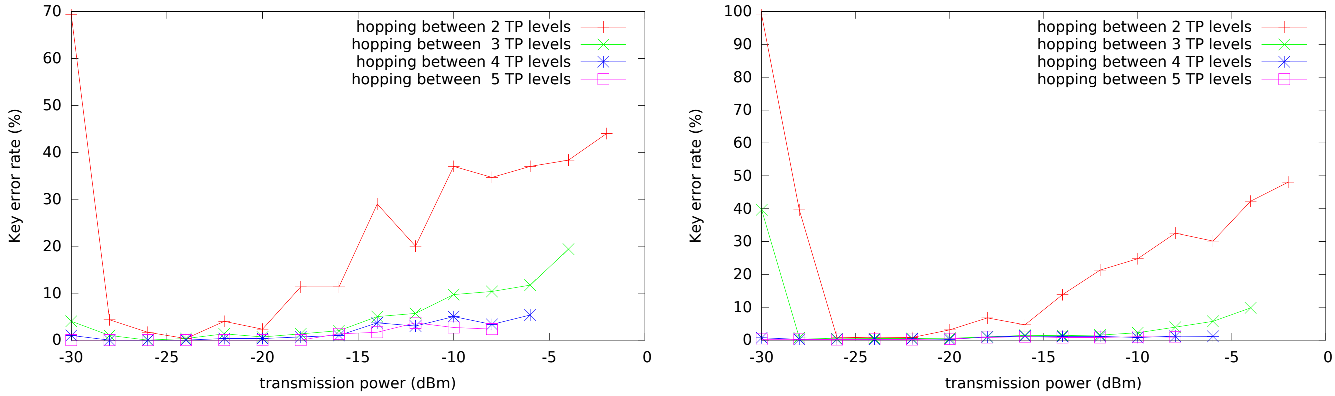
- *Distances*: we have four different scenarisos. In the first two scenarios, Both communication parties are stationary and their distances are are 2 metres and 5 metres, respectively. In the last two scenarios, the packet sender remained static when the receiver moved between 1 metres and 5 metres slowly (1 metre per minute) or moved in normal walking paces between 1 metre and 7 metres.
- *TP values* We adopted TP level from 1-31 in MicaZ. Some of the TP values for the main TP levels are listed in Table 6.1.
- *The TP level hopping gaps:* We would compare RSS of those respective packets where the TF level hopping gap is from 2 to 5 as there are 30 TF levels in Micaz. There would be 6 levels for TP side channel if the gap is 5, which would be sufficient in our scheme.

Table 6.1: Part of the Transmission Power settings in key error rate experiments (selective)

Transmission level	**Output(dBm)**	**Power (mW)**
31	0	31.3
27	-1	29.7
23	-3	27.4
19	-5	25.0
15	-7	22.5
11	-10	20.2
7	-15	17.9
3	-25	15.3

Given the same conditions and hardware, we measured RSS for 1,500 -2,000 packets for each TP levels (1 packet/second) and each packet had a sequence number. We compared those packets with the same sequence number in two different TP levels. The cases where the packets with lower RSS values which were sent in higher transmission power and vice versa would be counted. The key error rate was the percentage of these cases among all the transmssion cases (i.e., 1,500 -2,000 cases).

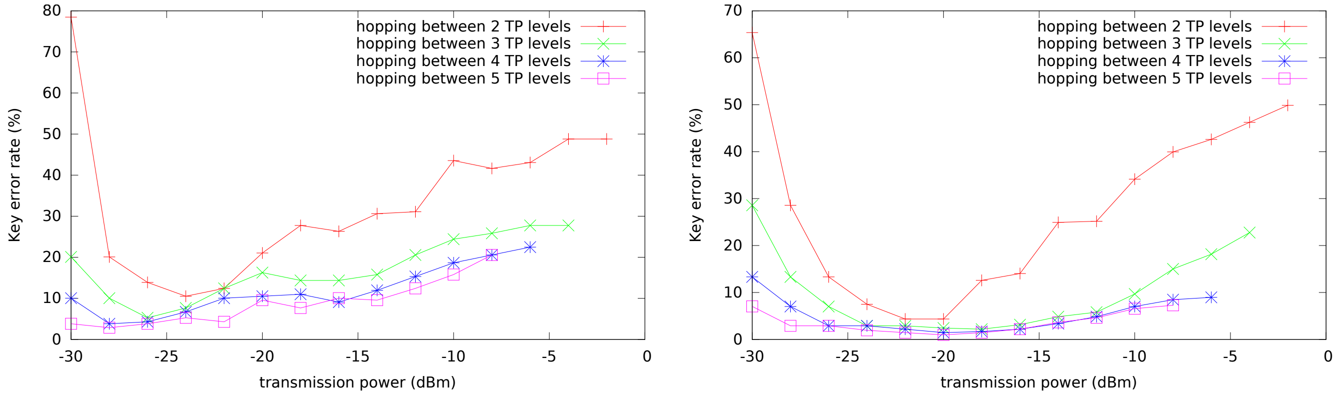
Figure 6.1 and Figure 6.2 depict the key error rates in the static and moving scenarios, respectively. The static scenarios perform much better than moving ones as there is significant impact on RSS due to distance variations in moving scenarios. This experimental results also confirm that our scheme could perform much better in a static scenario where the distance should not be too close (i.e., further than 2



(a) 2 meters (static).

(b) 5 meters (static).

Figure 6.1: The key error rate for static scenarios.



(a) fast walking within 7 metres.

(b) slow walking within 5 metres.

Figure 6.2: The key error rate for moving scenarios.

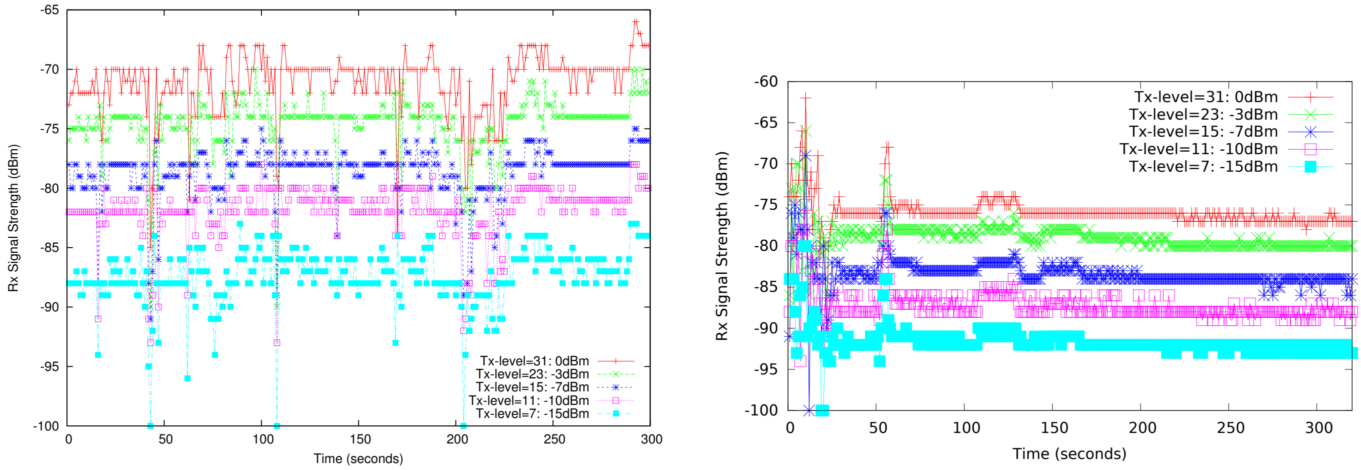
metres). If both Alice and Bob remain static during the key exchange process, the key error rate could be reduced by more than 10%. The optimal range of transmission powers is between -25 and -15 dBm. The most significant impact on key error rate is from the TP level hopping. In our experiments, the TP level hopping gap should be 4 or above so that the key error rate could be controlled under 1%.

6.2 Communication Overhead

Given an N-bit key to be established, the number of exchanged packets from the key extraction process would be $2N + 6$ as one request-response packet pair confirms one bit agreement in Figure 4.1. Alice and Bob would need to exchange a pair of "Start", "End" messages, as well as the packets of hashed checksums for the verification of final shared key. Therefore, the communication overhead is linear to the key length in our protocol. As to the payload of each packet, only the sequence number is included for the $2N$ packets. Therefore, one byte is needed for each of these exchanged packets except the hash checksum, which is 20 bytes. Given an $N-$bit key, the number of bytes to be transmitted are $2(N + 2) + 40$ as additionally, there is one pair of "Start" and "End" messages with one-byte payload, respectively and two hash checksums (20 bytes each) would be exchanged between Alice and Bob.

6.3 Bit Generation Rate

Figure 6.3 depicts the relationship between Received Signal Strength (RSS) and the Transmission Power for around 5 minutes in the static scenarios (1 packet/second). For the first 50 samples, the RSS values are overlapped with each other for different transmission powers regardless the distance as the wireless



(a) 2 meters (static).

(b) 5 meters (static).

Figure 6.3: The relationship between transmission power and RSSI: The longer the distance between two communication parties, the lower probabilities a mismatch would occur between the transmission power variations and received signal strength.

channel is being stablized in the first 50 seconds. After the wireless channel becomes stable, the RSS values fluctuate according to the transmission power variations. The closer two wireless devices are, the more likely the RSS would be overlapped with each other given a different transmission power (see Figure 6.3(a)). The further two wireless devices are, the more distinct the RSS would be (see Figure 6.3(b)).

Therefore, the distance between Alice and Bob could not be too close (e.g., within 2 meters) in our scheme. Otherwise the bits from the transmission power channel would be more likely to be interpreted differently by the packet recipient.

As is depicted in Figure 6.3, it takes around 50 seconds to reach the channel stablities (i.e., within the first 50 seconds, the RSS values are more subject to be overlapped given different transmission powers). If Alice and Bob needs to establish a 128-bit AES key, the communication overhead is 300 bytes according to Section 6.2. Given 250 kbps data rate, the transmission time is $300 * 8/(250 * 10^3) = 9.6$ms. The number of exchanged packets for 128-bit key is 256 (one request/response for one bit). We assume that the distance between Alice and Bob is 5 meters, the total propagation latency is $256*5/(3*10^8) = 42.6$us. Therefore, the packet delivery latency is negligible compared to the latency due to the channel stablities. The power switching latency is in the order of microseconds, which is also negligible [2]. The average bit rate for a 128-bit key generation is 2.6 bits/second. It is a significant improvement over the the existing key extraction through the wireless channels [9]. In our scheme, the bit generation rate is subject to the duration of the wireless channel can be stablized with a set of distinctive transmission power levels, rather than the key length. Therefore, our scheme is more efficient for long key generation in wireless networks.

7 Conclusions and future works

A key agreement protocol based on multiple side channels has been proposed. In this protocol, two side channels were employed to extract the secret bits between two legitimate wireless devices. One of the side channels is the change in transmission power while the other side channel is to swap the source destination address of the packets. These two side channels are subject to packet interception if only one of them is deployed for key agreement but they are robust against packet interception if they are both employed for key agreement at the same time. In addition, we showed that the communication overhead of key extraction process is linear to the number of bits in the shared key. In the future works, we will further evaluate the performance on our protocols in terms of key error rate due to the mismatch of transmission power change and variation of RSSI. In addition, we would extend our protocol to a group key agreement.

Bibliography

[1] Side-channel attack. https://en.wikipedia.org/wiki/Side-channel_attack, 2001, last accessed on 02/27/2016.

[2] Transmission power control in wireless sensor networks. http://web.cs.wpi.edu/r̃ek/Adv_Nets/Fall2009/WSN_TxPo 2009, last accessed on 04/08/2016.

[3] T. Aono, K. Higuchi, M. Taromaru, T. Ohira, and H. Sasaoka. Wireless secret key generation exploiting the reactance-domain scalar response of multipath fading channels : RSSI interleaving scheme. In *Wireless Technology, 2005. The European Conference on*, pages 173–176. IEEE, Oct. 2005.

[4] 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*, MobiCom '08, pages 116–127, New York, NY, USA, 2008. ACM.

[5] C. Castelluccia and P. Mutaf. Shake them up!: A movement-based pairing protocol for CPU-constrained devices. In *Proceedings of Mobisys*, MobiSys '05, pages 51–64, New York, NY, USA, 2005. ACM.

[6] K. Grover, A. Lim, and Q. Yang. Jamming and anti-jamming techniques in wireless networks: A survey. *Int. J. Ad Hoc Ubiquitous Comput.*, 17(4):197–215, Dec. 2014.

[7] H. Koorapaty, A. A. Hassan, and S. Chennakeshu. Secure information transmission for mobile radio. *IEEE Communications Letters*, 4(2):52–55, Feb. 2000.

[8] H. Liu, J. Yang, Y. Wang, and Y. Chen. Collaborative secret key extraction leveraging received signal strength in mobile wireless networks. In *2012 Proceedings IEEE INFOCOM*, pages 927–935. IEEE, 2012.

[9] S. Mathur, W. Trappe, N. Mandayam, C. Ye, and A. Reznik. Radio-telepathy: Extracting a secret key from an unauthenticated wireless channel. In *Proceedings of the 14th ACM International Conference on Mobile Computing and Networking*, MobiCom '08, pages 128–139, New York, NY, USA, 2008. ACM.

[10] R. Mayrhofer and H. Gellersen. Shake well before use: Intuitive and secure pairing of mobile devices. *IEEE Transactions on Mobile Computing*, 8(6):792–806, June 2009.

[11] N. Patwari, J. Croft, S. Jana, and S. K. Kasera. High-Rate uncorrelated bit extraction for shared secret key generation from channel measurements. *Mobile Computing, IEEE Transactions on*, 9(1):17–30, Jan. 2010.

[12] S. N. Premnath, J. Croft, N. Patwari, and S. K. Kasera. Efficient High-Rate secret key extraction in wireless sensor networks using collaboration. *ACM Trans. Sen. Netw.*, 11(1), July 2014.

[13] S. N. Premnath, S. Jana, J. Croft, P. L. Gowda, M. Clark, S. K. Kasera, N. Patwari, and S. V. Krishnamurthy. Secret key extraction from wireless signal strength in real environments. *Mobile Computing, IEEE Transactions on*, 12(5):917–930, May 2013.

[14] T. S. Rappaport. *Wireless Communications: Principles and Practice (2nd Edition)*. Prentice Hall, 2 edition, Jan. 2002.

[15] K. B. Rasmussen and S. Capkun. Implications of radio fingerprinting on the security of sensor networks. In *Security and Privacy in Communications Networks and the Workshops, 2007. SecureComm 2007. Third International Conference on*, pages 331–340. IEEE, 2007.

[16] G. Revadigar, C. Javali, W. Hu, and S. Jha. DLINK: Dual Link Based Radio Frequency Fingerprinting for Wearable Devices. In *Proc. 40th IEEE Conference on Local Computer Networks (LCN)*, 2015.

[17] B. A. Sadjadi, A. Kiayias, A. Mercado, and B. Yener. Robust key generation from signal envelopes in wireless networks. In *Proceedings of CCS'07*, pages 401–410, New York, NY, USA, 2007. ACM.

[18] C. Technologies. Mica2 and micaz motes. http://www.xbow.com, 2001, last accessed on 03/19/2016.

[19] J. W. Wallace, C. Chen, and M. A. Jensen. Key generation exploiting MIMO channel evolution: Algorithms and theoretical limits. In *Antennas and Propagation, 2009. EuCAP 2009. 3rd European Conference on*, pages 1499–1503. IEEE, Mar. 2009.

[20] J. W. Wallace and R. K. Sharma. Automatic secret keys from reciprocal MIMO wireless channels: Measurement and analysis. *IEEE Transactions on Information Forensics and Security*, 5(3):381–392, 2010.

[21] Z. Wang, J. Han, W. Xi, and J. Zhao. Efficient and secure key extraction using channel state information. *J. Supercomput.*, 70(3):1537–1554, Dec. 2014.

[22] M. Wilhelm, I. Martinovic, and J. B. Schmitt. Secure key generation in sensor networks based on Frequency-Selective channels. *IEEE Journal on Selected Areas in Communications*, 31(9):1779–1790, Sept. 2013.

[23] R. Wilson, D. Tse, and R. A. Scholtz. Channel identification: Secret sharing using reciprocity in ultrawideband channels. *Trans. Info. For. Sec.*, 2(3):364–375, Sept. 2007.

[24] K. Zeng, D. Wu, A. Chan, and P. Mohapatra. Exploiting Multiple-Antenna diversity for shared secret key generation in wireless networks. In *INFOCOM, 2010 Proceedings IEEE*, pages 1–9. IEEE, Mar. 2010.