TECHNISCHE UNIVERSITÄT DRESDEN

ComNets
Deutsche Telekom Chair of Communication Networks

Faculty of Electrical and Computer Engineering Institute of Communication Technology
Deutsche Telekom Chair of Communication Networks

Evaluation on the use of commodity WiFi hardware for testing out physical layer security key generation schemes

Markus Schmidl

Course: Diploma Information Systems Engineering
Matriculation number: 4759854
Matriculation year: 2018

Student Thesis

Supervisor

M.Sc. Hosein Kangavar Nazari

Supervising professor

Prof. Dr.-Ing. Dr. h.c. Fitzek

Submitted on: 12th June 2023

TECHNISCHE UNIVERSITÄT DRESDEN

Fakultät Elektrotechnik und Informationstechnik
Institut für Nachrichtentechnik
Deutsche Telekom Professur Kommunikationsnetze
Prof. Dr.-Ing. Dr. h. c. Frank Fitzek

Task for Student Thesis of
Mr. Markus Schmidl (Academic year: IST/2018, matriculation number: 4759854)

Topic: Evaluation on the use of commodity WiFi hardware for testing out physical layer security key generation schemes

Task description:

Physical layer security key generation provides a novel framework for securing the communication between two nodes without the need for prior secret sharing on higher layers or computationally expensive public key cryptography. Use cases include dynamic mesh networks, Internet of Things (IoT) devices, and any situation where prior secret sharing is impractical or where computational resources are constrained.

Different key generation procedures and random sources are being discussed in literature, but to the best of our knowledge there are no open source prototypes.

The research field lacks techniques for building reproducible environments and evaluating different algorithms with actual hardware. An open source implementation will make it easier for other researchers to validate findings and improve upon them.

This student thesis should focus on the following points:

- Profound literature research about physical layer security key generation and how it can be employed in wireless networks
- Evaluation if and how key generation can be employed into existing commodity WiFi hardware
- Examining the possibilities for implementing a future prototype enabling key generation

The student thesis will be written in English.

Supervisor: M.Sc. Hosein Kangavar Nazari

Started at: 12.12.2022

To be submitted by: 29.05.2023

Prof. Dr.-Ing. Dr. h.c. Fitzek
Responsible Tutor

Statement of authorship

I hereby certify that I have authored this document entitled *Evaluation on the use of commodity WiFi hardware for testing out physical layer security key generation schemes* independently and without undue assistance from third parties. No other than the resources and references indicated in this document have been used. I have marked both literal and accordingly adopted quotations as such. There were no additional persons involved in the intellectual preparation of the present document. I am aware that violations of this declaration may lead to subsequent withdrawal of the academic degree.

Dresden, 12th June 2023

Markus Schmidl

TECHNISCHE UNIVERSITÄT DRESDEN

ComNets
Deutsche Telekom Chair of Communication Networks

Faculty of Electrical and Computer Engineering Institute of Communication Technology

Deutsche Telekom Chair of Communication Networks

Abstract

Wireless networks have become an integral part of our society and increasingly being used in critical applications. However, the inherent possibilities for eavesdropping on wireless channels call for improved security measures. Following the requirement of secrecy in communication, potential vulnernbilities should be ruled out where possible, to secure against powerful adversaries. This thesis will focus on the requirements needed to implement physical layer key generation. Perfect secrecy is ensured applying the knowledge of information theory. Utilizing the physical properties of the wireless communication channel, an encryption key is created without prior secret sharing. We investigate how these physical measurements fulfill the theoretical requirements, taking a bottom up view on WiFi hardware for key generation and its consequences for these methods. We present relevant measurements indicating that the prerequisites of physical layer key generation could be fulfilled, however more consideration has to be given to important system parameters, e.g. the antenna design or the operating frequency.

List of Figures

List of Tables

Symbols and Acronyms

CSI Channel State Information

RSS Received Signal Strength

RSSI Received Signal Strength Indicator (Used as a synonym for RSS. They only have a differentiate in absolute value and scaling.)

Contents

1 Introduction

Wireless networks have become an essential backbone of our society. According to [24] it is estimated that by 2023 the number of wireless devices have surpassed the three times global population. This trend also includes that critical applications, i.e. smart home, smart traffic, healthcare or e-commerce (Internet of Things) use wireless channels [17, Figure 1]. Due to inherent possibility to eavesdrop on an air interface more easily than on a cable, we need to look at the security of these systems and how security can be improved in the future.

Existing algorithms for enforcing privacy (encryption) are computationally expensive [17, Table 11] or need prior secret sharing. Additionally, they also make assumptions about the potential enemy's computing power [4]. However, IoT and wireless mesh networks usually have a tight computational and power budget, rendering increasing the computational complexity of encryption algorithms impractical. Therefore, other alternative approaches to generating a shared secret key for encryption have to be considered.

Ahlswede and Csiszar [3] and Maurer [4] proposed a framework for enforcing perfect secrecy using the tools of information theory. This approach, called physical layer key generation, allows generating a secret key between two devices if they have access to a common random variable. Recent literature has combined this theory with physical properties of the wireless radio link as side channel information [8]. This is possible without additional hardware as wireless channels are being estimated in modern wireless systems already, though only some provide documented interfaces in firmware.

The goal of this thesis is to explain the background and theory of physical layer key generation in wireless networks in a structured way that is easy to follow.

This is assisted by showing if and how existing commodity WiFi chipsets can be employed for key generation. Then, relevant measurements, which are direct logical conclusions of the theory, are explained to verify the practicality of this approach.

In Section 2, we will explain the relevant background and introduce related papers and ideas. Section 3 will focus on introducing the testbed using commodity WiFi chips that was build during this thesis. Finally, in Section 5 the thesis is concluded by a discussion on how a future prototype can be implemented based on the knowledge gained in this thesis. Measurements and results using the testbed will be explained in Section 4.

2 Background and Related Work

There are several approaches for securing wireless networks. Figure 2.1 provides an overview of the different research streams in wireless network security. Common methods like symmetric or asymmetric encryption have been researched extensively, but they suffer from two problems. First, symmetric encryption needs a prior shared secret and there are attack scenarios if the same data is encrypted twice. For more detailed background we advise reading a textbook about cryptanalysis. Second, the significant computational complexity of asymmetric encryption given though the use of complex mathematical problems, limits its applicability in very low power systems.

Because of these downsides, physical layer security tries to provide approaches that are based on the physical properties of the wireless channel. Methods for keyless security include beam forming [14] or artificial noise [21] to degrade the wireless channel of an unwanted recipient. Key generation on the other hand uses

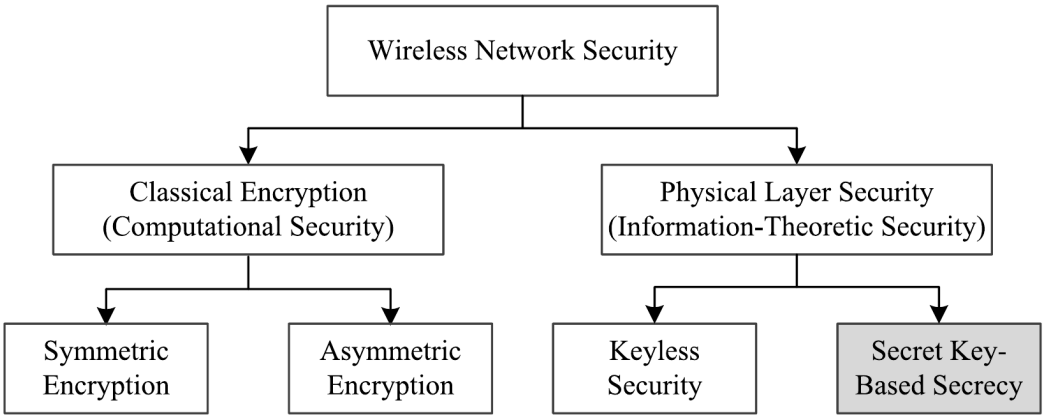

Figure 2.1: Research streams in wireless network security. Extracted from [13]

the physical properties of the wireless channel between two nodes to establish a key which can be used for encryption. This work will focus on physical layer key generation.

First, we give an overview of the information theoretic background based on Shannon's theory in Section 2.1. Then we give background information about channel estimation, which will be essential for the implementation of key generation in Section 2.2. Section 2.3 will tie all of this together with the theory of key generation.

2.1 Information Theoretic Background on Encryption

Shannon introduced a model for cryptography in [1]. He defined that a cipher is perfect if the encrypted message C does not leak any information about the plaintext M. This property is described by the mutual information[1] in the following equation:

$$I(M;C) = 0 \tag{2.1}$$

$$I(X;Y) := \sum_{y\in\mathcal{Y}} \sum_{x\in\mathcal{X}} p_{(X,Y)}(x,y) \log \left(\frac{p_{(X,Y)}(x,y)}{p_X(x)\, p_Y(y)} \right) \tag{2.2}$$

One perfect cipher is the one-time pad, where the key K is a random binary sequence with the same length as the message. A key will not be reused for encryption. He also proved that perfect secrecy is only possible if the entropy[2] of the key is at least as high as that of the message, given that an attacker can receive the same information as the intended receiver.

$$H(K) \geq H(M) \tag{2.3}$$

$$H(X) := -\sum_{x\in\mathcal{X}} p(x) \log p(x) \tag{2.4}$$

Since all practical ciphers using classical encryption approaches violate this constrains, they can theoretically be broken even if it is currently not viable [4]. This has been show in the past and is actively being exploited by intelligence agencies [7].

In [2], [4] Maurer showed that Equation (2.3) does not hold true if the eavesdropper does not have the same information as the legitimate sender/receiver. This opens the possibility for perfect secrecy to be achieved using more practical

[1] The mutual information is denoted with the symobl I.
[2] The entropy is denoted with the symobl H.

methods. These ideas were also discovered by Ahlswede and Csiszàr [3]. Quantum cryptography is another approach to this problem, by distributing keys securely based on the knowledge of quantum superpositions and entanglement. It will not be considered in this thesis.

2.2 Channel Estimation

The performance of modern wireless networks depends on the possibility to accurately determine the wireless channel at a given time between transmit and receive antennas. However, modelling wireless channels accurately is difficult [6]. Different physical phenomena occur in wireless communication, such as scattering, refraction, reflection, or diffraction, resulting in inter symbol interference which depends on the environment in which the waves interact. This effect is summarized in the term multi-path propagation. The environment however is not constant, it changes constantly, this effect is called fading and is important for practical key generation.

Due to the nature of these complex physical effects, one can not simply model the wireless channel for all possible scenarios, they have to be estimated. This procedure is called channel estimation and is the process of determining the coefficients of channel matrix described below by using pilot symbols, sequences of known symbols.

The wireless channel between t transmitting and r receiving antennas is modelled through the equation

$$y = Hx + z \tag{2.5}$$

$$H = \begin{bmatrix} h_{11} & h_{12} & \dots & h_{1t} \\ h_{21} & h_{22} & \dots & h_{2t} \\ \vdots & \vdots & \ddots & \vdots \\ h_{r1} & h_{r2} & \dots & h_{rt} \end{bmatrix} \tag{2.6}$$

where $x \in \mathbb{C}^t$ is the transmitted signal and $y \in \mathbb{C}^r$ the received signal. $H \in \mathbb{C}^{r \times t}$ is the channel matrix, where each element h_{ij} describes the channel gains from the jth transmitting to the ith receiving antennas. z is zero-mean complex gaussian noise. The range of i and j is: $1 \leq j \leq t$ and $1 \leq i \leq r$ [5].

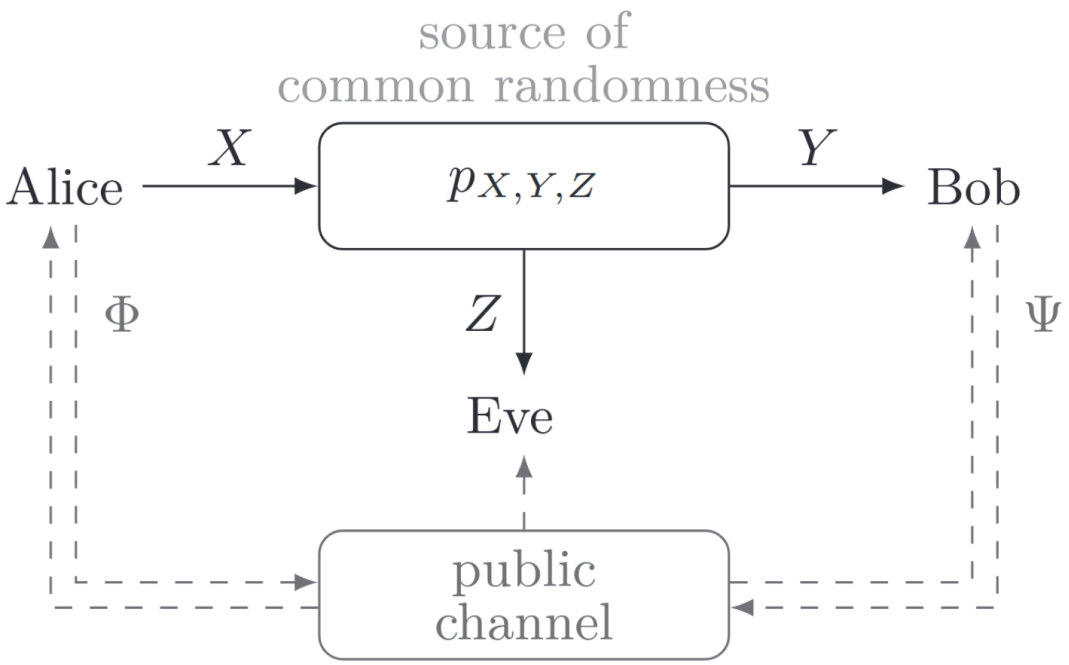

Figure 2.2: Source type model for key generation. Extracted from [10]

2.3 Key Generation

Ahlswede and Csiszàr [3] and Maurer [4] laid the foundation for generating a shared secret (key generation) between two parties without prior secret sharing. The two system models, source-type and channel-type, were introduced in [3]. Though this theory is purely information theoretical, here we will explain it with direct application in wireless networks.

For the system model generally two wireless devices called Alice and Bob want to generate a key unknown to a passive eavesdropper Eve. In the source-type model illustrated in Figure 2.2, Alice, Bob and Eve observe the random statistical variables X, Y and Z respectively, which are jointly distributed through the probability function $p_{X,Y,Z}$. Additionally, there is a public channel where Alice and Bob can exchange messages. Φ and Ψ denote messages sent over this channel by Alice and Bob, respectively [3]. We will focus on this model, as we will be using the wireless channel reciprocity as the source of common randomness. In the next chapter we explain this in more detail.

The channel-type model has a wiretap channel from Alice to Bob, where Eve is the wiretapper. The public channel is present as explained above. For a detailed description of this model, we refer to [10, §1.11].

Ahlswede and Csiszàr [3] proposed a theoretical model how common randomness can be used and what information-theoretic inequalities communication after

retrieving common randomness has to fulfil to generate a secret key without leaking any information about it both at Alice and Bob.

K and L will be the key generated by Alice and Bob, respectively. R_s is the secret key rate. Following [3, Definition 2.1]: R_s is the achievable key rate if for every $\epsilon > 0$ and sufficient large n (number of outputs of the random variables X and Y), there is a communication strategy such that

$$Pr\{K \neq L\} < \epsilon \tag{2.7}$$

$$\frac{1}{n} I(\Phi^k, \Psi^k; K) < \epsilon \tag{2.8}$$

$$\frac{1}{n} H(K) > R_s - \epsilon \tag{2.9}$$

and

$$\frac{1}{n} log|K| < \frac{1}{n} H(K) + \epsilon. \tag{2.10}$$

Equation (2.7) states that the generated key by both Alice and Bob have to be the same with regard to a small error probability ϵ. Equation (2.8) states that practically no information about the key may be leaked through the public discussion. The indices k denote that each iteration of the public discussion should fulfil this inequality. Equation (2.10) states that the generated key K should be uniformly distributed.

This very theoretical groundwork has been expanded to an algorithm depicted in Figure 2.3. Generally, multiple names for the same process are used in literature. A more detailed explanation of each step can be found in [12]. Following steps must be carried out.

1. Initialization or Channel Coding: Alice and Bob exchange signals, e.g. pilot signals, from which a physical layer property will be estimated.
2. Estimation of physical layer characteristics or Preprocessing: Alice and Bob estimate a physical layer property, e.g. calculate the channel state information.
3. Quantization and Encoding: Convert the signals into a bitstream suitable for secret key generation.
4. Information reconciliation: Alice and Bob communicate over a public channel to make sure they generate the same key.
5. Privacy Amplification: Generate a shorter bit stream with a higher entropy to minimise the chance of guessing the secret key based on potential leaked information, e.g. a hash function.

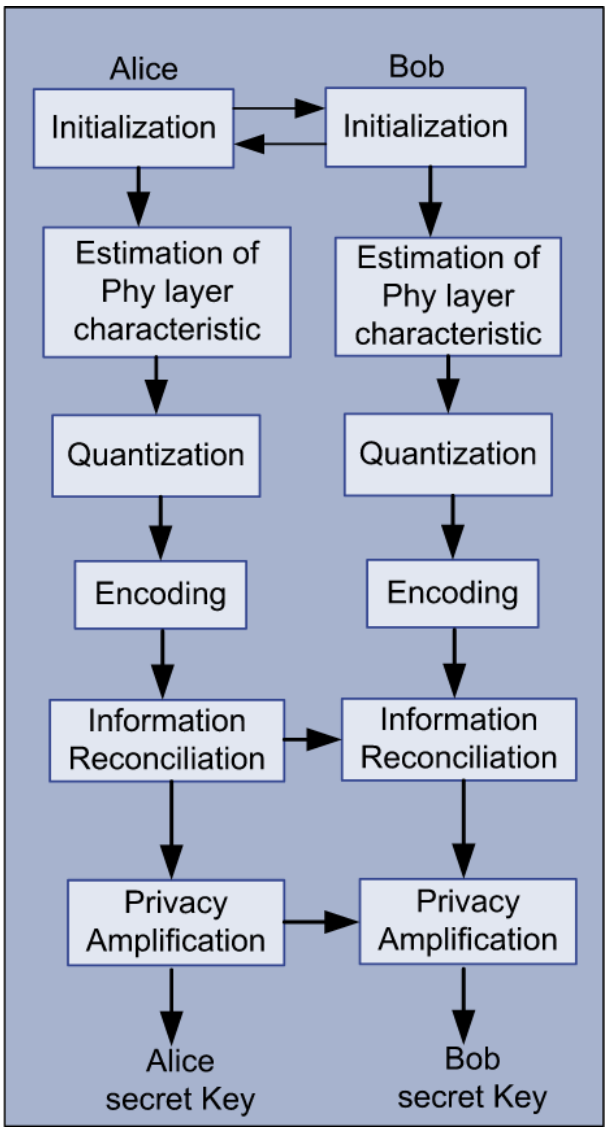

Figure 2.3: Secret key generation steps. Extracted from [12]

Different authors suggested different approaches to get common randomness from the wireless channel. Early work suggested *"to receive the signal of a satellite broadcasting random bits [...], or of a deep space radio source."* [4]. Most popular is RSS and CSI [13], though CSI is preferable as it provides more data which can be used to achieve a higher key rate. To get the channel state information, pilot signals have to be sent over the wireless channel from Alice to Bob and back. Due to the wireless channel fading, these pilots have to be timed as close to each other as possible, to reduce the difficulty of information reconcilement. Some authors [19] looked at the problem of extracting common randomness when the uplink and downlink channel of a wireless communication system are not using the same frequency. Generally, a lot of different approaches exist which have to be evaluated for a given system.

We know of some implementations [9], [16], though to the best of our knowledge no code has been published. Zhang et. al. [17] provide a recent overview of current

work and different methods used in all of these steps. The key generation steps 3 to 5 are outside the scope of this thesis. We will focus on the measurements from our hardware which can be used as the basis for key generation.

2.4 P-Value

To check if realizations of statistical variables are correlated, we need to calculate p-values. The p-values describes the probability of how likely we would see a specific outcome if the realization of the statistical variables are not correlated (null hypothesis). If the null hypothesis is not true, the p-values should be below a certain threshold α, which is commonly set to 0.05 [23, Sec. 9.5, 12.5].

3 Experimental Setup

To implement physical layer key generation in practice, we have to extract correlated measurements of the wireless channels between Alice and Bob. This resulted in a testbed to extract the wireless channel parameters from commercial wireless hardware. We build upon the AtherosCSI framework [11], a framework for extraction of channel estimation measurements from an 802.11n WiFi routers running the open-source OpenWRT linux distribution. This allows us to easily modify the firmware and running code. Though we used WiFi 802.11n hardware, our methods and physical layer key generation in general is not dependent on this specific choice.

The resulting concept for the measurements will be presented in Section 3.1. Section 3.2 shows what data can be extracted from the access points and what consequences this has for the testbed. Finally, Section 3.3 will conclude how the requirements from the previous sections shape the testbed in practice.

3.1 Concept

To evaluate our hardware, we needed to create a suitable concept. It should allow for data to be saved for future use, be easy to implement and evaluate different potential problems.

Figure 3.1 depict the result of those requirements. It shows the flow of wireless data between our testbed of three access points (Alice, Bob and Eve) and a server for collection of the CSI measurements. Alice and Bob want to create a secret key,

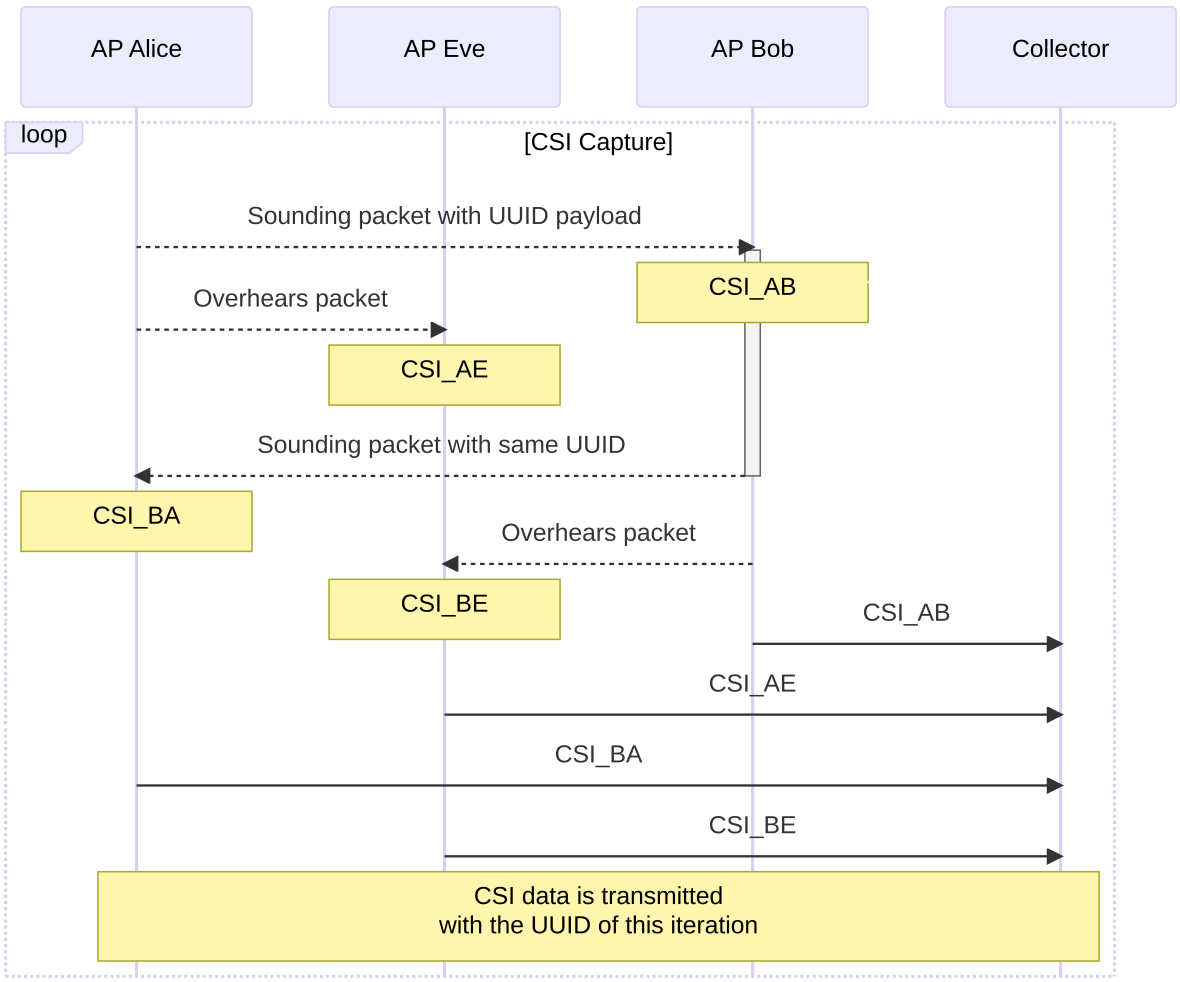

Figure 3.1: Dataflow inside the testbed. Dotted arrows indicate wireless packets, which can be used to extract the channel state information. Solid ones indicate the channel state being saved on the collection server.

which Eve the passive Eavesdropper should have no knowledge about. First, the channel from Alice to Bob is estimated with a pilot signal. Shortly after, the channel from Bob to Alice. Eve, the passive eavesdropper, can also estimate their channel from Alice to Eve and Bob to Eve using this pilot signal. All access points send the collected channel measurements to a central collection server afterwards. This way we can analyse the measurements in the future.

We chose WiFi access points following the 802.11n standard as a cheap and readily available prototyping platform for the physical layer measurements. This standard allows Channel State Information extraction through the Transmit Beamforming (TxBF) feature. It is not widely supported, but some WiFi chipsets allow extracting this data with a modified linux kernel module. The AtherosCSI framework [11] demonstrated the possibility of extracting the channel state estimation measurements for specific Atheros chipsets, which are shown in Table 3.1. This allows for getting rapid results in evaluating how key generation can be implemented on existing

WiFi hardware. Extracting hardware measurements is also possible for newer WiFi standard such as 802.11ax [20][1]. It has also been demonstrated for RSSI measurements with a now outdated WiFi standard in the PROPHYLAXE project [15]. However, all of these solutions need some sort of modification in the linux kernel or driver for the chipset, since there is no standardized interface for extracting the required data.

Supported	Supported Devices (excerpt)	Chipset	Reference
Yes	TL-WR2543ND	AR9380	GitHub, wands.sg
Yes		AR9382	wands.sg
Yes		AR9462	wands.sg
Yes		AR9565	wands.sg
Yes		AR9580	wands.sg
Yes		AR9590	wands.sg
Yes		QCA9344	wands.sg
Yes	TL-WDR4300	QCA9544	wands.sg
Yes	TL-WR1043NDv2	QCA9558	wands.sg
Unknown		AR9280	GitHub
Unknown		QCA9342	wands.sg
Unknown		QCA9563	wands.sg
No		QCA9531	wands.sg
No	GL-iNet AR300M16	QCA9533	We tested it.
No	Archer C7 v5	QCA9563	GitHub, GitHub

Table 3.1: Supported chipsets by the AtherosCSI framework.

To use the AtherosCSI framework, we first had to port it to a recent version of OpenWRT. This involved understanding the code for extracting the CSI from the wireless chipset and writing a userspace application to send the data to a server. For the server, an application had to be written to save the data for future analysis. Furthermore, the ath9k kernel module was re-written to remove multiple bugs of the original version. Prior to this rewrite, documentation was distributed in forks of OpenWRT without seperate commits for their changes, papers [18], leaked datasheets and the linux kernel mailing list.

We use the beamforming feature (called TxBF in datasheets) of 802.11n to extract the CSI. Figure 3.2 displays how the CSI is propagated to the userspace. One access point sends out a so called “Sounding Packet” which is received by another access point.[2] This packet contains a pilot signal which is used to estimate the wireless channel between the sending and receiving access point. The modified kernel driver sets some registers to enable extracting the CSI parameters. Once the packet is

[1] PicoScenes Supports CSI Extraction for 802.11ax-Format Frames via Intel AX200 Wi-Fi NIC

[2] This is described in the following GitHub issue: https://github.com/xieyaxiongfly/Atheros-CSI-Tool/issues/4/#issuecomment-348391011

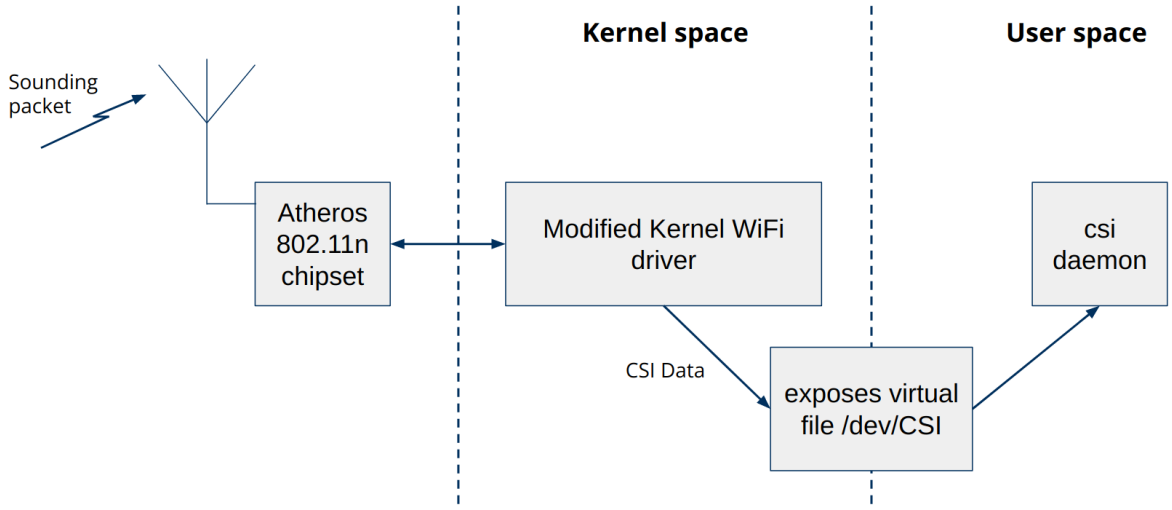

Figure 3.2: Wireless chipset estimates the wireless channel. The modified kernel driver forwards the estimated channel measurements to a virtual file descriptor, where they can be read by a userspace application.

received and the parameters extracted from the kernel driver, these get forwarded to a virtual file descriptor where they can be read from an userspace application. This application called “csi daemon” sends this data to the collection server and controls the dataflow as show in Figure 3.1.

As shown in Table 3.1 we tried to extract the CSI from a GL-inet AR300M16, which has 2 antennas (both for transmit and receive). This would have resulted in very cheap prototyping hardware, but due to the limited time of the thesis we decided to skip debugging this hardware and move to a known working device/chipset, the TP-Link WR1043NDv2 featuring 3 antennas (both for transmit and receive). This results in $3 \times 3 = 9$ different wireless channels between two WiFi devices.

3.2 Extracted Data

Following the steps from Section 2.3 we need to check if and how the measurements of the wireless channel are correlated to each other. From the Atheros chipset, we can extract received signal strength (RSS) for each antenna and the complex channel coefficients for each TX-RX antenna pair. We opted to use CSI for evaluating key generation as higher key rates compared to RSS can be achieved, due to inherently having more bits available [13]. Furthermore, only the magnitude of those complex values is used due to potential phase ambiguity for the used modulation scheme.

If the wireless channel between Alice and Bob would be correlated 100% and the channel to Eve 0%, we would have a key rate as high as the number of bits of the CSI parameters. This ideal assumption gives a figure of the maximum possible key rate. For our setup the maximum is defined by the chosen bandwidth of the WiFi channel, the number of receive/transmit antennas, and the resolution of complex number. This results in the values displayed equation Equation (3.1).

$$|K|_{max} = \begin{cases} 3 \text{ tx antennas} \times 3 \text{ rx antennas} \times \\ 20 \text{ bits per complex number} \times 56 = 1260\,\text{B}, & \text{if } BW = 20\,\text{MHz} \\ 3 \text{ tx antennas} \times 3 \text{ rx antennas} \times \\ 20 \text{ bits per complex number} \times 114 = 2565\,\text{B}, & \text{if } BW = 40\,\text{MHz} \end{cases} \tag{3.1}$$

3.3 Testbed in Practice

The stated requirements from Section 3.1 and data we are able to extract from the hardware resulted in a setup with three TP-Link WR1043NDv2 access points and one server. The access points run a modified OpenWRT 21.02. The patches for OpenWRT and all other software is attached on a CD-ROM.[3]

Figure 3.3 shows the placement of the three access points relative to each other. They were at least half a wavelength spaced apart from each other so that the wireless channels are spatially decorrelated from each other [22].

Figure 3.4 displays all access points next to each other. Please note that the measurements were not done in this configuration, but it is rather for illustrational purposes, as they would not fit inside the picture otherwise. The small access points were not used for the measurements, these are the GL-iNET AR300M16 which we evaluated, but turned out not to work. Presumably the TxBF is not implemented in hardware.

The next section will show what kind of measurements were performed and the conclusion we draw regarding the feasibility of creating a prototype using this hardware.

[3] Additionally it can also be found in the following GitHub repository: https://github.com/marenz2569/student-thesis-physical-layer-key-generation

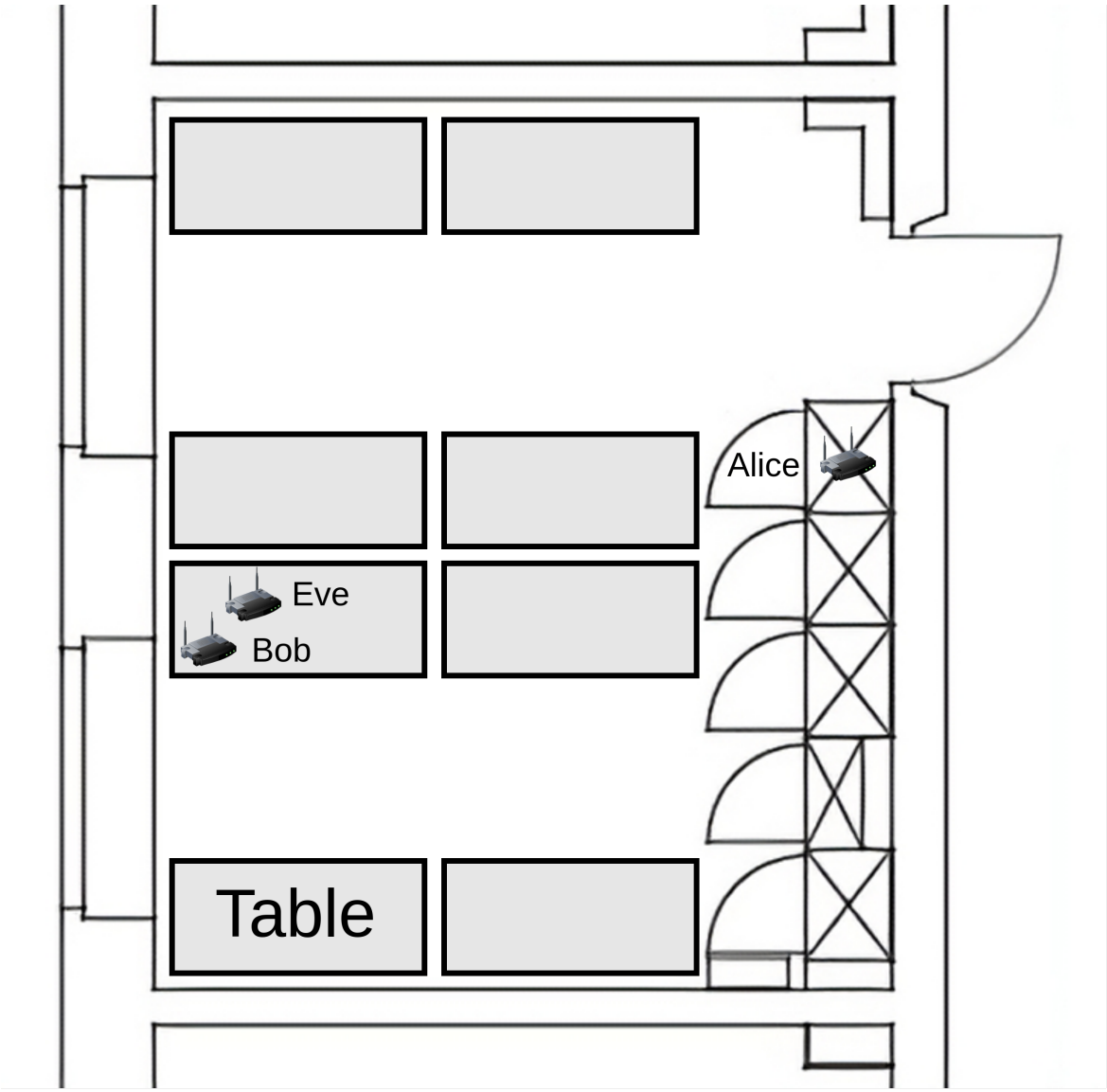

Figure 3.3: Physical placement of the access points. Alice is placed inside a closet. Bob and Eve are separated by about $30\,\mathrm{cm}$ and located on a table at the other end of the room.

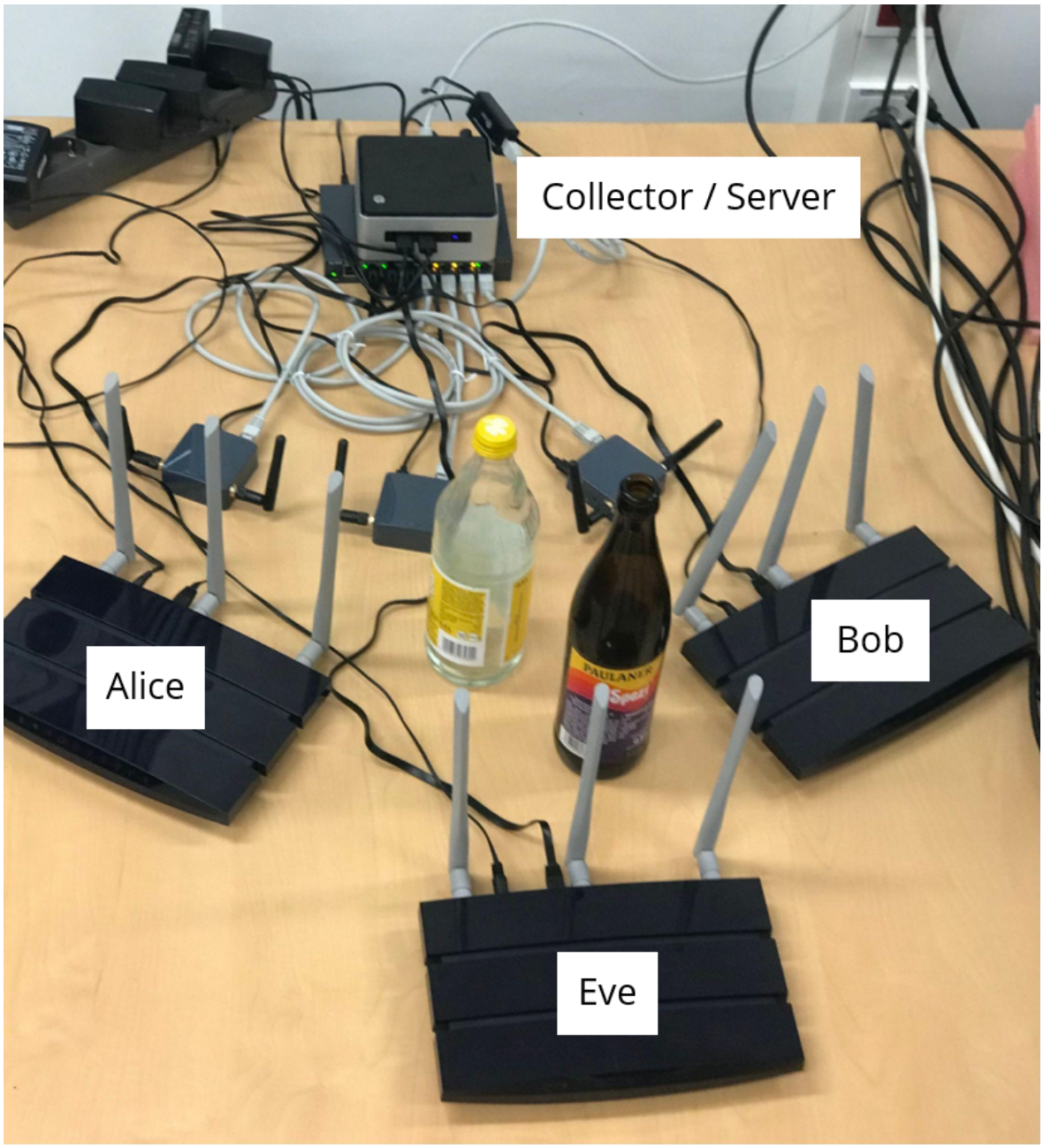

Figure 3.4: Components of the testbed placed together.

4 Results

We did two different kind of measurements for this thesis. With the first measurement, we evaluate the round trip time between the channel estimations. The second is used to checked how correlated the wireless channels are. If not stated otherwise, the channel estimation have been repeated 500 times every $1\,\mathrm{s}$. We discarded measurements where not all channel estimations (Alice $\rightarrow$ Bob, Alice $\rightarrow$ Eve, Bob $\rightarrow$ Alice and Bob $\rightarrow$ Eve) were available, resulting in a total of 402 repetitions. This can happen, for example, due to interference.

Figure 4.1 displays one measurement of the wireless channel between Alice and Bob. The grid displays each combination of each 3 receive and transmit antennas. One can see visual similarities in some of the 9 subplots, but this has to be investigated on a more thorough basis. The phase is not displayed as it might have ambiguity due to used modulation, this is another problem which should be investigated, further in the future.

4.1 Measurement I: Round Trip Time

To achieve good performance, in this context a low key disagreement rate and high key rate, depend on the channel reciprocity between Alice and Bob. Following this fact, the time difference between the measurement of the CSI between Alice and Bob should idealy be zero, though this is not possible in a time-sharing system. Therefore, we have to measure this time difference. This can be done by Eve, as it overhears both the packets from Alice and Bob and assigns them a timestamp with

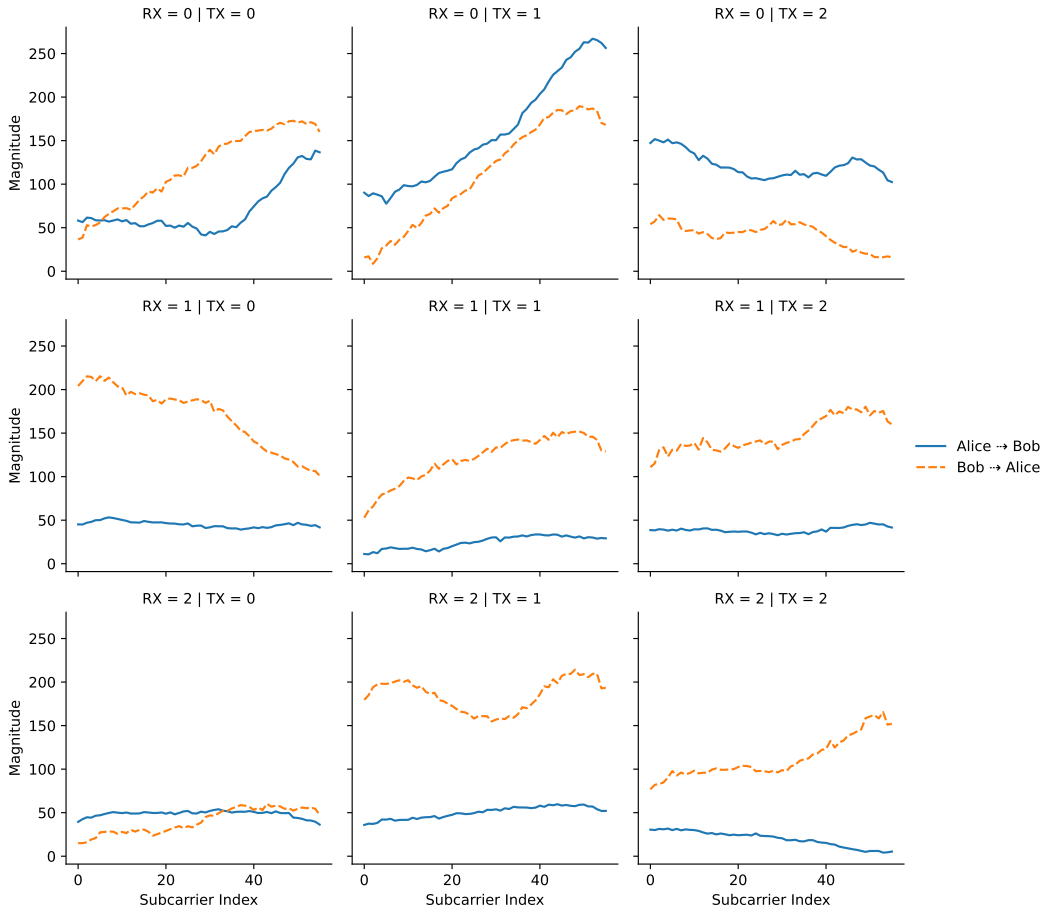

Figure 4.1: Exemplary measurement of the wireless channel between Alice and Bob. The vertical axis displays the magnitude of the complex channel matrix for each subcarrier.

a hardware counter on reception. We use these timestamps to calculate the time difference. This data is plotted in Figure 4.2.

The average time between the channel estimations is $2.5\,\mathrm{ms}$. This is half compared to the result Xi et. al. achieved with a similar setup [9, Sec. II-B]. The minimum is at $1.0\,\mathrm{ms}$ and maximum at $11.5\,\mathrm{ms}$.

When this measurement was done for the first time, the results were much worse, because a user space program is controlling the time of sending out a “Sounding Packet”. After waiting for the measurement to be finished and only then sending the data to the server, the results improved significantly. Furthermore, results could be improved by moving the application responsible for timing of this inside the kernel space or even the firmware, though this can only be done by the manufacturer of the hardware.

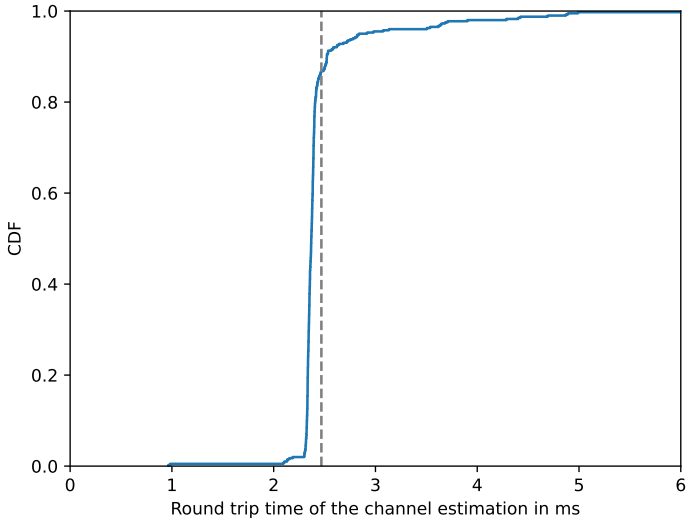

Figure 4.2: Empirical cumulative distribution function of the time difference between the pilot signals of Alice and Bob.

4.2 Measurement II: Channel Reciprocity

To get insight how correlated the CSI measurements between Alice and Bob are, we calculated the p-values for each RX-TX pair for each measurement using the *pearsonr* method from scipy[1]. For each of the 402 measurements, an empirical cumulative distribution is drawn.

It is clear that only the RX=0, TX=1 and RX=1, TX=1 antenna pairs show significant correlation in all measurements. Though it is unclear what causes this, it is important to take this into account for designing an algorithm for key generation. If it would not be taken into account, it could result in a high key disagreement rate. However we assume that if the time between channel estimations is reduced this effect will turn out to be less drastical.

[1] https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.pearsonr.html

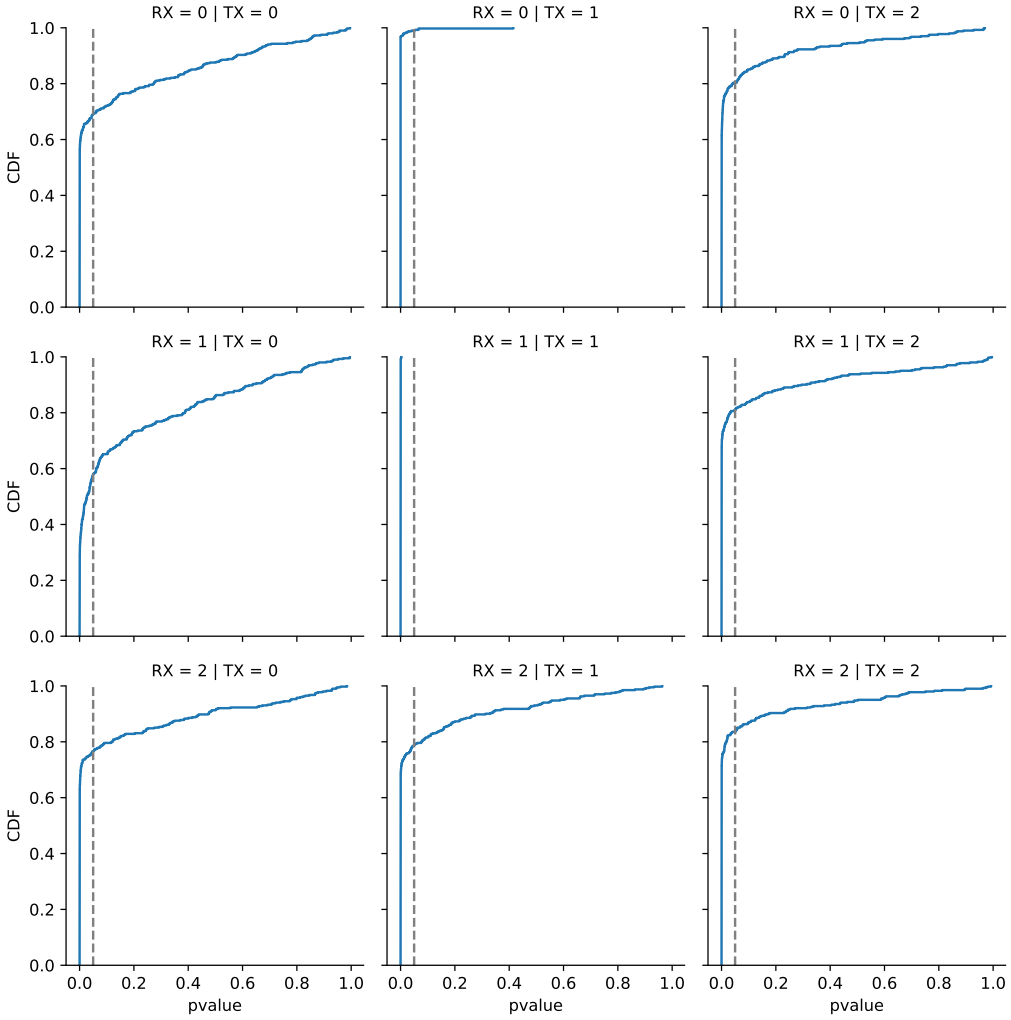

Figure 4.3: Empirical cumulative distribution function of the calculated p-values between the channel estimations of Alice and Bob. The gray broken line denotes $\alpha = 0.05$.

5 Conclusion and Further Work

In this thesis, we evaluated if and how commodity WiFi hardware can be employed for key generation. A testbed with an 802.11n WiFi chipset was built using the open source OpenWRT platform. Measurements of the wireless channel are presented and conclusions are drawn based on the requirements of key generation. Our platform allows researchers to extract the channel state parameters and experiment with real-world data. We were able to extract correlated measurements of the wireless channel, the first step necessary towards a key generation prototype. However, different physical phenomena and complex cross connections from multiple engineering fields have to be taken into account to successfully create a secure implementation. These include system parameters, like the antenna design or how the channel estimation inside the chipset works. They can have a significant impact on the design of algorithms and the performance of the system. Furthermore, systems with a higher operating frequency should be studied as they would observe faster fading, an important metric for how fast or often a key can be generated. Our results and software are published[1], which allows other researchers to build upon these and give them a head start.

[1] It can be found in the following GitHub repository: https://github.com/marenz2569/student-thesis-physical-layer-key-generation

Bibliography

[1] C. E. Shannon, "Communication theory of secrecy systems", *The Bell System Technical Journal*, vol. 28, no. 4, pp. 656–715, Oct. 1949, Conference Name: The Bell System Technical Journal, ISSN: 0005-8580. DOI: 10.1002/j.1538-7305.1949.tb00928.x.

[2] U. M. Maurer, "Perfect cryptographic security from partially independent channels", en, in *Proceedings of the twenty-third annual ACM symposium on Theory of computing - STOC '91*, New Orleans, Louisiana, United States: ACM Press, 1991, pp. 561–571, ISBN: 978-0-89791-397-3. DOI: 10.1145/103418.103476. [Online]. Available: http://portal.acm.org/citation.cfm?doid=103418.103476 (visited on 04/05/2023).

[3] 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, ISSN: 1557-9654. DOI: 10.1109/18.243431.

[4] U. Maurer, "Secret key agreement by public discussion from common information", *IEEE Transactions on Information Theory*, vol. 39, no. 3, pp. 733–742, May 1993, Conference Name: IEEE Transactions on Information Theory, ISSN: 1557-9654. DOI: 10.1109/18.256484.

[5] E. Biglieri and G. Taricco, *Transmission and Reception with Multiple Antennas: Theoretical Foundations*. now, 2004. [Online]. Available: https://ieeexplore.ieee.org/document/8186857.

[6] K. Anusuya, S. Bharadhwaj, and S. Rani, "Wireless Channel Models for Indoor Environments", en, *Defence Science Journal*, vol. 58, no. 6, pp. 771–777, Nov. 2008, ISSN: 0011748X, 0976464X. DOI: 10 . 14429 / dsj . 58 . 1706. [Online]. Available: http://publications.drdo.gov.in/ojs/index.php/dsj/article/view/1706 (visited on 06/04/2023).

[7] "Inside the NSA's War on Internet Security", en, *Der Spiegel*, Dec. 2014, ISSN: 2195-1349. [Online]. Available: https : / / www . spiegel . de / international / germany/inside-the-nsa-s-war-on-internet-security-a-1010361.html (visited on 04/04/2023).

[8] S. Primak, K. Liu, and X. Wang, "Secret Key Generation Using Physical Channels with Imperfect CSI", in *2014 IEEE 80th Vehicular Technology Conference (VTC2014-Fall)*, ISSN: 1090-3038, Sep. 2014, pp. 1–5. DOI: 10 . 1109 / VTCFall . 2014 . 6966172.

[9] W. Xi, X.-Y. Li, C. Qian, *et al.*, "KEEP: Fast secret key extraction protocol for D2D communication", in *2014 IEEE 22nd International Symposium of Quality of Service (IWQoS)*, ISSN: 1548-615X, May 2014, pp. 350–359. DOI: 10.1109/IWQoS.2014.6914340.

[10] A. Wolf, "Robust optimization of private communication in multi-antenna systems", Ph.D. dissertation, Technische Universität Dresden, 2015. [Online]. Available: https://nbn-resolving.org/urn:nbn:de:bsz:14-qucosa-203827.

[11] Y. Xie, Z. Li, and M. Li, "Precise power delay profiling with commodity wifi", in *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '15, Paris, France: ACM, 2015, pp. 53–64, ISBN: 978-1-4503-3619-2. DOI: 10.1145/2789168.2790124. [Online]. Available: http://doi.acm.org/10.1145/2789168.2790124.

[12] A. Badawy, T. Elfouly, T. Khattab, A. Mohamed, and M. Guizani, "Unleashing the secure potential of the wireless physical layer: Secret key generation methods", en, *Physical Communication*, vol. 19, pp. 1–10, Jun. 2016, ISSN: 18744907. DOI: 10.1016/j.phycom.2015.11.005. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S1874490715000713 (visited on 07/06/2022).

[13] J. Zhang, T. Q. Duong, A. Marshall, and R. Woods, "Key Generation From Wireless Channels: A Review", *IEEE Access*, vol. 4, pp. 614–626, 2016, Conference Name: IEEE Access, ISSN: 2169-3536. DOI: 10.1109/ACCESS.2016.2521718.

[14] Y. Zhang, "Wireless security with beamforming technique", Ph.D. dissertation, Queen's University Belfast, 2016. arXiv: 1609.03629 [cs.IT].

[15] C. Zenger, "Physical-layer security for the internet of things", en, Ph.D. dissertation, Ruhr-Universitat Bochum, 2017. [Online]. Available: https://hss-opus.ub.ruhr-uni-bochum.de/opus4/frontdoor/deliver/index/docId/5187/file/diss.pdf.

[16] N. Felkaroski and M. Petri, "Secret Key Generation Based on Channel State Information in a mmWave Communication System", in *SCC 2019; 12th International ITG Conference on Systems, Communications and Coding*, Feb. 2019, pp. 1–6. DOI: 10.30420/454862049.

[17] J. Zhang, G. Li, A. Marshall, A. Hu, and L. Hanzo, "A New Frontier for IoT Security Emerging From Three Decades of Key Generation Relying on Wireless Channels", *IEEE Access*, vol. 8, pp. 138 406–138 446, 2020, Conference Name: IEEE Access, ISSN: 2169-3536. DOI: 10.1109/ACCESS.2020.3012006.

[18] Z. Jiang, T. H. Luan, X. Ren, *et al.*, *Eliminating the Barriers: Demystifying Wi-Fi Baseband Design and Introducing the PicoScenes Wi-Fi Sensing Platform*, arXiv:2010.10233 [cs], Aug. 2021. [Online]. Available: http://arxiv.org/abs/2010.10233 (visited on 01/17/2023).

[19] X. Zhang, G. Li, J. Zhang, A. Hu, Z. Hou, and B. Xiao, *Deep Learning-based Physical-Layer Secret Key Generation for FDD Systems*, arXiv:2105.08364 [cs, eess], Aug. 2021. [Online]. Available: http://arxiv.org/abs/2105.08364 (visited on 01/17/2023).

[20] F. Gringoli, M. Cominelli, A. Blanco, and J. Widmer, "AX-CSI: Enabling CSI Extraction on Commercial 802.11ax Wi-Fi Platforms", en, in *Proceedings of the 15th ACM Workshop on Wireless Network Testbeds, Experimental evaluation & CHaracterization*, New Orleans LA USA: ACM, Jan. 2022, pp. 46–53, ISBN: 978-1-4503-8703-3. DOI: 10.1145/3477086.3480833. [Online]. Available: https://dl.acm.org/doi/10.1145/3477086.3480833 (visited on 01/18/2023).

[21] M. S. Kumar, R. Ramanathan, and M. Jayakumar, "Key less physical layer security for wireless networks: A survey", en, *Engineering Science and Technology, an International Journal*, vol. 35, p. 101 260, Nov. 2022, ISSN: 22150986. DOI: 10.1016/j.jestch.2022.101260. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S2215098622001690 (visited on 06/10/2023).

[22] L. Yang, Y. Gao, J. Zhang, S. Camtepe, and D. Jayalath, "A Channel Perceiving Attack on Long-Range Key Generation and Its Countermeasure", *Computer Communications*, vol. 191, pp. 108–118, Jul. 2022, arXiv:1910.08770 [eess], ISSN: 01403664. DOI: 10.1016/j.comcom.2022.04.027. [Online]. Available: http://arxiv.org/abs/1910.08770 (visited on 06/04/2023).

[23] Barbara Illowsky and Susan Dean, *Introductory Statistics*. De Anza College. [Online]. Available: https://stats.libretexts.org/Bookshelves/Introductory_Statistics/Book:_Introductory_Statistics_(OpenStax) (visited on 05/06/2023).

[24] "Cisco Annual Internet Report (2018–2023)", Tech. Rep. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.pdf?gh_jid=4423171 (visited on 03/30/2023).