Practical Known-Plaintext Attacks against Physical Layer Security in Wireless MIMO Systems

Matthias Schulz, Adrian Loch and Matthias Hollick
Technische Universität Darmstadt
Secure Mobile Networking Lab
Email: {matthias.schulz, adrian.loch, matthias.hollick}@seemoo.tu-darmstadt.de

***Abstract*—Physical layer security schemes for wireless communication systems have been broadly studied from an information theory point of view. In contrast, there is a dearth of attack methodologies to analyze the achievable security on the physical layer. To address this issue, we develop a novel attack model for physical layer security schemes, which is the equivalent to known-plaintext attacks in cryptoanalysis. In particular, we concentrate on analyzing the security of orthogonal blinding schemes that disturb an eavesdropper's signal reception using artificial noise transmission. We discuss the theory underlying our attack methodology and develop an adaptive filter trained by known-plaintext symbols to degrade the secrecy of orthogonal blinding. By means of simulation and measurements on real wireless channels using software-defined radios with OFDM transceivers, we obtain the operating area of our attack and evaluate the achievable secrecy degradation. We are able to reduce the secrecy of orthogonal blinding schemes to Symbol Error Rates (SERs) below 10 % at an eavesdropper, with a knowledge of only a 3 % of the symbols transmitted in typical WLAN frames.**

I. Introduction

Security solutions for wireless systems based on cryptography are inevitably bound to an expiration date, since today's state-of-the-art cryptographic protocols may be broken in the future. Well known examples include the DECT Standard Cipher (DSC)—broken using cryptanalysis [16], the A5/1 encryption standard used in GSM—vulnerable to a ciphertext-only attack made possible due to error-correction codes being used before the encryption [4], the Wired Equivalent Privacy (WEP) as used in the early 802.11 standard—broken due to inappropriate use of the RC4 stream cipher [8], and the Advanced Encryption Standard (AES)—vulnerable to side channel attacks in some implementations [18]. While the attack model underlying each of these vulnerabilities is different, they all share a common threat, i.e., a message eavesdropped and stored today may be decrypted as soon as the security scheme in use is broken. Moreover, such broken schemes cannot always be avoided, as support for legacy systems might be needed [13].

NDSS '14, 23-26 February 2014, San Diego, CA, USA

http://dx.doi.org/10.14722/ndss.2014.23162

A highly promising approach to overcome this limitation is to implement physical layer security, which prevents eavesdropping in the first place. Intuitively, senders manipulate data either before transmission or "on-air" such that only the intended receiver can decode it successfully, while eavesdroppers only receive a degraded signal [25]. To achieve this, a number of techniques exist, including friendly jamming [11]. In this case, well-behaved nodes selectively jam signals to prevent their reception by malicious nodes. Conversely, they also may jam signals sent by the malicious nodes themselves to prevent message injection [24]. However, such approaches exhibit weaknesses, since a conveniently placed attacker with multiple antennas may cancel out the jamming signal and recover the transmitted data [23]. While physical layer security is powerful, this attack showcases that such techniques necessitate a thorough analysis to fully understand their limitations in practical settings.

To provide confidentiality, more sophisticated techniques than jamming based on orthogonal blinding exist. A transmitter can protect data by sending artificial noise into a channel orthogonal to the receiver's channel. In other words, all nearby nodes receive a superposition of noise and signal, while the intended receiver gets only the signal. The achievable secrecy rate [25] using such a scheme has been thoroughly investigated in theory [10], [15], [26]. Additionally, there exists also practical work showing the feasibility of orthogonal blinding in a software-defined radio testbed [3]. This technique requires a MIMO configuration, since the transmitter needs at least two antennas in order to send noise into an orthogonal channel. However, the intended receiver only needs one antenna. In particular, the transmitter uses the Channel State Information (CSI) to the intended receiver to prefilter data and artificial noise. As a result, there is only data in the dimension visible to the intended receiver, while potential eavesdroppers always receive a mixture of noise and signal. Even if eavesdroppers have more than one antenna, artificial noise is not fully orthogonal to any of them with high probability. Additional antennas at the transmitter can be used to send noise on multiple orthogonal channels. This reduces even further the probability that an eavesdropper has a link to the transmitter which suppresses the artificial noise.

In this paper, we investigate the limitations of orthogonal blinding. Existing work is partly based on assumptions which might not hold in a realistic setting. Specifically, (a) data is typically assumed to be fully unknown to the eavesdropper and (b) the attacker is assumed to have less antennas than the transmitter [3]. Regarding (a), the use of well-known protocols

and addresses may allow the attacker to *guess* parts of the transmitted data. This information can be used to mount an attack analog to a *known-plaintext attack* in the cryptography domain, where the attacker has samples of both the plaintext and the ciphertext. Concretely, the attacker can compare the known plaintext with the ciphertext and derive the key, which can then be used to decrypt the rest of the data. Similarly, an eavesdropper who can guess parts of the data transmitted by a sender using orthogonal blinding can use this information to suppress the artificial noise reception by training an adaptive filter based on the known plaintext. In other words, the eavesdropper iteratively filters the data, starting with a given default filter, and compares the output to the known plaintext. Based on the observed difference, the filter is adapted in each iteration until converging, i.e., until the difference between the filter output and the expected plaintext is minimized. Since the attacker needs at least as many antennas as the transmitter to be able to discern all signal dimensions used by the transmitter, assumption (b) would prevent this weakness, but this relegates the security of the scheme to the capabilities of the attacker, who might be very powerful.

More precisely, the analogy of our attack model to the case of a known-plaintext attack is as follows. The data sent by the transmitter corresponds to the plaintext, the mixture of transmitted data and artificial noise is the ciphertext, and the CSI of the link from the transmitter to the intended receiver is the key.

Note that our attack model is not limited to the case of orthogonal blinding. Known-plaintext attacks based on adaptive filtering could also be used to compromise other physical layer security schemes which prefilter data at the transmitter. In summary, our contributions are as follows:

- We propose an attack model which applies the concept of known-plaintext attacks from the cryptography domain to physical layer security.
- We design a practical attack scheme which instantiates our model for the case of physical layer security based on orthogonal blinding.
- We discuss the theory underlying our attack and obtain the operating area of our scheme by means of extensive simulation.
- We implement and evaluate our scheme on software-defined radios to show its practicability.

The remainder of this work is structured as follows. In Sections II and III we introduce our system and communication model. After that, we first briefly explain how orthogonal blinding works in Section IV and then delve into the details of our known-plaintext attack in Section V. In Section VI we present our simulation outcome and our practical evaluation on the Wireless Open-Access Research Platform (WARP) [2] software-defined radio. We discuss our results in Section VII. Finally, we give an overview of related work in Section VIII and conclude our work in Section IX.

II. System and Attack Model

Our system model is illustrated in Figure 1. It contains a transmitter Alice, who confidentially sends data to the intended receiver Bob over a wireless channel $H_{\mathcal{A}\to\mathcal{B}}$. We additionally consider a passive eavesdropper Eve, who intends to extract the confidential data $D_{\mathcal{A}\to\mathcal{B}}$. To prevent the latter, Alice applies a physical layer secrecy scheme. In our example, this is orthogonal blinding [3]. In this scheme, Alice transmits artificial noise in addition to the data signal so that Bob is not disturbed by the noise, but any eavesdropper—having a different channel from that of $H_{\mathcal{A}\to\mathcal{B}}$—receives both data signal and noise. As long as Eve does not know the transmit filter $F_{\mathcal{A},\text{TX}}$ used to mix data and noise, she is unable to extract the data from her received signal $\mathcal{E}$. According to [25], this ensures the secrecy of the system. As Alice's transmit filter $F_{\mathcal{A},\text{TX}}$ is based on the knowledge of the channel from Alice to Bob ($H_{\mathcal{A}\to\mathcal{B}}$), which is not available to Eve, Eve cannot generate an optimal receive filter $F_{\mathcal{E},\text{RX}}$.

To still degrade the secrecy of the orthogonal blinding scheme, we assume that Eve knows parts of the transmitted data: the known plaintext. In the cryptography domain, a sound cryptographic algorithm should withstand a known-plaintext attack, amongst other basic attacks, to be considered secure. We apply this consideration also to physical layer security schemes and develop a novel attack model, which is the equivalent to known-plaintext attacks in cryptoanalysis. As a practical example we choose orthogonal blinding to demonstrate the efficacy of our attack methodology.

In our attack model, Eve trains an adaptive filter $F_{\mathcal{E},\text{RX}}$ with known plaintext symbols. The trained filter can then be used to extract the unknown data. For filter training, the error between the filter output and the known plaintext is minimized. Once trained, the filter is independent of the transmitted artificial noise.

The following section focuses on our attack model steps, and how they apply to orthogonal blinding.

A. Communication Phases

Our attack model comprises three phases that we describe hereunder together with examples regarding orthogonal blinding:

1) Channel estimation between Alice and Bob:

- Alice transmits pilot symbols, which Bob uses to estimate the channel $H_{\mathcal{A}\to\mathcal{B}}$ from Alice to him (Section III-A).
- Bob uses an out-of-band channel to send $H_{\mathcal{A}\to\mathcal{B}}$ to Alice. Here, we give orthogonal blinding an advantage; alternatively our system could rely on implicit feedback, as used in IEEE 802.11n [19].

2) Securing transmission:

- Alice uses $H_{\mathcal{A}\to\mathcal{B}}$ to generate her transmit filter $F_{\mathcal{A},\text{TX}}$ (Section IV-B).
- Alice applies $F_{\mathcal{A},\text{TX}}$ to mix data and artificial noise, and transmits the result.
- Bob extracts the data after compensating channel effects in his receive filter $F_{\mathcal{B},\text{RX}}$ (Section IV-B).

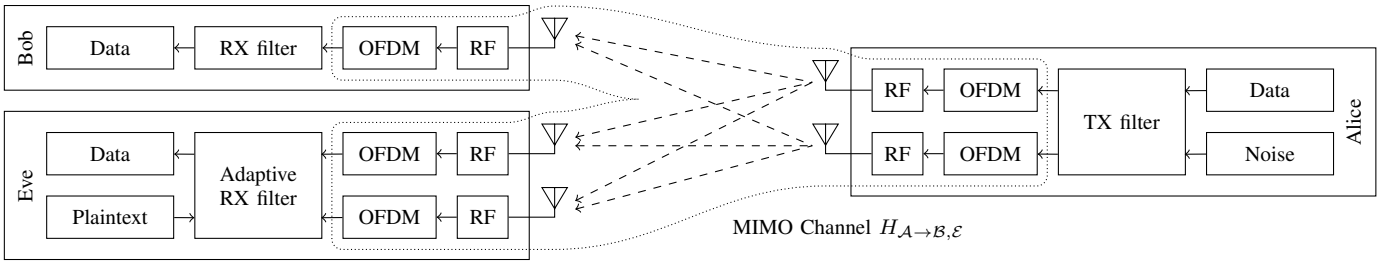


Fig. 1. Our system model illustrating the transmitter Alice, the intended receiver Bob and the passive eavesdropper Eve.

3) Extraction of data by Eve:

- Eve uses her plaintext knowledge and her received signal to train an adaptive receive filter $F_{\mathcal{E},\mathrm{RX}}$ (Section V).
- Eve applies the trained filter to extract the unknown data (Section V).

After performing these steps, we calculate the SERs at Bob and Eve to evaluate our attack's secrecy degradation. Only if Eve's SER does not decrease when carrying out the attack, the secrecy scheme can be considered resistant against known-plaintext attacks.

Before continuing with a mathematical description of the communication system in Section III, we present our adversary model in the following section.

B. Adversary model

Both receivers Bob and Eve, are limited by their receive hardware's sensitivity to detect incoming signals that are further disturbed by Additive White Gaussian Noise (AWGN). Eve might significantly improve her reception compared to that of Bob—in the sense of Signal to Noise Ratio (SNR)—if she shifted her position in a given environment or if she used directional antennas. Thus, we assume that Eve's antennas can be freely positioned. In case this reduced the feasibility of representing Eve as a single node, we suppose that multiple eavesdroppers cooperate and exchange their received signals so that all received signals are available at one point (see [10], [21]). Multiple antennas help to additionally increase the SNR on Eve's channel $H_{\mathcal{A}\to\mathcal{E}}$, as the AWGN is independent of the transmitted signal and can be reduced by destructive interference.

To optionally increase Eve's attack performance with a limited number of antennas (see Section VI-E), we assume that Eve can estimate the channel $H_{\mathcal{A}\to\mathcal{E}}$ from Alice to Eve using the publicly available pilot symbols that Alice transmits to estimate her channel to Bob. Blinding the pilot symbols does not prevent the attack, as pilot symbols themselves can be regarded as known plaintext used for filter training. Furthermore, we require that Eve has partial knowledge of the transmitted data—that could be (but is not limited to) protocol headers and addresses. The amount of data needed to compromise secure information's confidentiality is evaluated in Section VI. The channel information $H_{\mathcal{A}\to\mathcal{B}}$ from Alice to Bob is, however, not disclosed to Eve, and Eve's channel $H_{\mathcal{A}\to\mathcal{E}}$ is revealed neither to Alice nor to Bob.

III. Communication System

We now present our communication system, which draws primarily on a state-of-the-art Multiple Input Multiple Output (MIMO) transceiver using Orthogonal Frequency Division Multiplexing (OFDM) [6], which is employed to abstract from fading channels and to cope with Inter Symbol Interference (ISI). Similar technology is used in the current 802.11ac Wi-Fi standard [19]. This allows to port both the physical layer security scheme as well as our known-plaintext attack against it onto widely available hardware. For the sake of simplicity, we use Software-Defined Radios (SDRs) to implement and assess our system in Section VI.

In what follows, we present the general MIMO channel model (Section III-A) and show how to apply it to our scenario (Section III-C). We conclude with transmit and receive filtering (Section III-D).

A. Channel model

The wireless channel between each pair of antennas is described as a complex number $H_{r,t}$ (the channel coefficient) representing a phase and an amplitude change of the transmitted signal during a transmission. Each of the R receive antennas gets a superposition of all T transmitted signals traversing different channels:

$$\mathrm{RX}_r = \sum_{\tau=1}^{T} H_{r,\tau} \cdot \mathrm{TX}_\tau \tag{1}$$

In matrix form:

$$\underbrace{\begin{pmatrix}\mathrm{RX}_1\\ \mathrm{RX}_2\\ \vdots\\ \mathrm{RX}_R\end{pmatrix}}_{\mathrm{RX}} = \underbrace{\begin{pmatrix}H_{1,1} & H_{1,2} & \dots & H_{1,T}\\ H_{2,1} & H_{2,2} & \dots & H_{2,T}\\ \vdots & \vdots & \ddots & \vdots\\ H_{R,1} & H_{R,2} & \dots & H_{R,T}\end{pmatrix}}_{H} \underbrace{\begin{pmatrix}\mathrm{TX}_1\\ \mathrm{TX}_2\\ \vdots\\ \mathrm{TX}_T\end{pmatrix}}_{\mathrm{TX}} \tag{2}$$

This channel abstraction is only valid if the channel coefficients are equal over the whole transmission bandwidth. Indoor channels with reflections, however, experience frequency selective fading, where the channel coefficients are frequency-dependent. To split wide-band channels into narrow subchannels, where the channel coefficients are considered constant, we apply OFDM, which also avoids ISI.

Instead of transmitting symbols consecutively in the time domain over a band of frequencies, symbols are transmitted in parallel over K frequency subbands—called subchannels. In the time domain, the effect of the wireless channel is modeled

as a convolution of a signal $\text{tx}[n]$ and the channel's impulse response $h_{r,t}[n]$; whereas in the frequency domain (calculated by applying the Discrete Fourier Transform (DFT) of size K), the channel is modeled as a simple multiplication of a symbol $\text{TX}[k]$ by a channel coefficient $H_{r,t}[k]$ on every subchannel k:

$$\text{DFT}_K(\text{tx}[n] * h_{r,t}[n])[k] = \text{TX}[k] \cdot H_{r,t}[k] \tag{3}$$

To estimate the channel coefficient $H_{r,t}[k]$ for transmitter $t = \tau$, pilot symbols $\text{TX}_{\tau,\text{pilot}}$ are transmitted on every subchannel k, while the other transmitters remain silent:

$$H_{r,\tau}[k] = \left.\frac{\text{RX}_{r,\text{pilot}}[k]}{\text{TX}_{\tau,\text{pilot}}[k]}\right|_{\text{TX}_{t\neq\tau}[k]=0} \tag{4}$$

Estimating all $H_{r,t}[k]$, we obtain MIMO channel matrices $H[k]$ for each subchannel k. Therefore, Equation 2 can be generalized to the case of multiple subchannels:

$$\text{RX}[k] = H[k] \cdot \text{TX}[k] \tag{5}$$

B. Modelling channel noise

So far our channel model has not taken into account any disturbances due to noise on the wireless channel, noise in the hardware components, or quantization noise. For our purposes, we apply the common approach to represent the aforementioned effects by complex AWGN η, which is added to the time-domain signal at the receiver:

$$\text{rx}[n] = \text{tx}[n] * h_{r,t}[n] + \eta[n] \tag{6}$$

During the filter design phase, noise effects are disregarded; however, they are considered in our simulations and measurements.

C. Applying the channel model

We have two separate MIMO channels on each subchannel k: $H_{\mathcal{A}\to\mathcal{B}}[k]$ from Alice to Bob and $H_{\mathcal{A}\to\mathcal{E}}[k]$ from Alice to Eve. Each transmitted set of MIMO symbols is denoted by the column vector $\mathcal{A}[k]$ which has length $T_{\mathcal{A}}$. Bob's and Eve's receive vectors have lengths $R_{\mathcal{B}}$ and $R_{\mathcal{E}}$, and are designated by $\mathcal{B}[k]$ and $\mathcal{E}[k]$, respectively:

$$\mathcal{B}[k] = H_{\mathcal{A}\to\mathcal{B}}[k] \cdot \mathcal{A}[k] \tag{7}$$
$$\mathcal{E}[k] = H_{\mathcal{A}\to\mathcal{E}}[k] \cdot \mathcal{A}[k] \tag{8}$$

For channel estimation, Alice transmits pilot symbols $\mathcal{A}_{t,\text{pilot}}[k] \in \{-1, 1\}$ that are known to all nodes. Bob and Eve receive these symbols at all of their $R_{\mathcal{B}}$, respectively $R_{\mathcal{E}}$, antennas and apply Equation 4 to estimate the channels $H_{\mathcal{A}\to\mathcal{B}}[k]$ and $H_{\mathcal{A}\to\mathcal{E}}[k]$.

D. Filtering

Channel estimates are mainly used to generate transmit and receive filters. We derive the following transmit filter equation, using the general channel model from Equation 2:

$$\underbrace{\begin{pmatrix}\text{TX}_1\\ \text{TX}_2\\ \vdots\\ \text{TX}_T\end{pmatrix}}_{\text{TX}} = \underbrace{\begin{pmatrix}F_{1,1} & F_{1,2} & \dots & F_{1,L}\\ F_{2,1} & F_{2,2} & \dots & F_{2,L}\\ \vdots & \vdots & \ddots & \vdots\\ F_{T,1} & F_{T,2} & \dots & F_{T,L}\end{pmatrix}}_{F_{\text{TX}}} \underbrace{\begin{pmatrix}D_1\\ D_2\\ \vdots\\ D_L\end{pmatrix}}_{D} \tag{9}$$

If the channel H to the receivers is available, the transmitter can design a zero-forcing filter that cancels the effect of the channel so that each filter input dimension is directly linked to one receive dimension, assuming that H is invertible:

$$\text{RX} = H \cdot (\overbrace{\underbrace{H^{-1}}_{F_{\text{TX}}} \cdot D}^{\text{TX}}) = D \tag{10}$$

To have less constraints when inverting the channel matrix, the so-called right pseudoinverse might be used. It is given by:

$$H^{-1}_{\text{right}} = H^{\text{H}}(H \cdot H^{\text{H}})^{-1} \tag{11}$$

where $(.)^{\text{H}}$ is the conjugate transpose. Alternatively, filtering can be performed in the receiver to extract the transmitted data D from the received signal RX, by applying zero-forcing at the receiver instead of the transmitter:

$$F_{\text{RX}} \cdot \text{RX} = \underbrace{H^{-1}}_{F_{\text{RX}}} \cdot (\underbrace{H \cdot \overbrace{D}^{\text{TX}}}_{\text{RX}}) = D \tag{12}$$

Again, the inverse of the channel can be replaced with the left pseudoinverse, which is given by:

$$H^{-1}_{\text{left}} = (H^{\text{H}} \cdot H)^{-1} H^{\text{H}} \tag{13}$$

In certain scenarios it is also possible to apply transmit as well as receive filters. This is further discussed in Sections IV and V.

IV. Physical Layer Security Scheme

To wirelessly transmit data, bits are mapped to symbols that are represented by complex numbers defining amplitude and phase of analog sine and cosine waves. The modulation scheme employed in this work to map bits to symbols with different amplitude and phase is called Quadrature Amplitude Modulation (QAM). During transmission, these symbols get disturbed by AWGN so that a receiver has to estimate which symbols were transmitted. The higher the noise power is, the higher the SER gets.

The class of physical layer security schemes we analyze uses this property to increase the secrecy of a transmission. To this end, Alice can transmit both artificial noise and data in a way that Bob successfully receives the plain data, while Eve's reception is disturbed by noise, which prevents a successful demodulation.

In the literature, we find two approaches to achieve the targeted goal. (i) Alice knows both channels to Bob $H_{\mathcal{A}\to\mathcal{B}}$ and to Eve $H_{\mathcal{A}\to\mathcal{E}}$ ([5], [14], [17], [20]). This assumption is unlikely to be fulfilled, as a passive eavesdropper does not share its channel information. However, approach (i) allows to define an upper bound for the achievable secrecy rate. (ii) Alice only knows her channel to Bob and needs a way to still disturb Eve [3], [7], [26]. This is a more practical approach. In the following, we briefly introduce the first approach and describe the second in more detail.

A. Zero-Forcing

The first approach assumes Alice has full channel knowledge and at least as many antennas as Bob and Eve together. Then Alice can combine Bob's and Eve's channels into a single channel matrix $H_{\mathcal{A}\to\mathcal{B},\mathcal{E}}$, which she inverts to cancel out the effect of the complete channel. To meet the transmit power constraints, Alice introduces a normalization factor $1/\alpha_{\text{norm}}$ into her transmit filter $F_{\mathcal{A},\text{TX}}$:

$$\begin{pmatrix}\mathcal{B}\\ \mathcal{E}\end{pmatrix} = \underbrace{\begin{pmatrix}H_{\mathcal{A}\to\mathcal{B}}\\ H_{\mathcal{A}\to\mathcal{E}}\end{pmatrix}}_{H_{\mathcal{A}\to\mathcal{B},\mathcal{E}}} \cdot \overbrace{\underbrace{\frac{1}{\alpha_{\text{norm}}}\begin{pmatrix}H_{\mathcal{A}\to\mathcal{B}}\\ H_{\mathcal{A}\to\mathcal{E}}\end{pmatrix}^{-1}\begin{pmatrix}D_{\mathcal{A}\to\mathcal{B}}\\ D_{\mathcal{A}\to\mathcal{E}}\end{pmatrix}}_{\mathcal{A}}}^{F_{\mathcal{A},\text{TX}}} \tag{14}$$

Bob receives $\mathcal{B} = D_{\mathcal{A}\to\mathcal{B}}/\alpha_{\text{norm}}$ and denormalizes it multiplying by α_{norm}. The data symbols $D_{\mathcal{A}\to\mathcal{E}}$ are intended for Eve, and Alice can choose them to be zero, artificial noise, or any other signal. As long as Eve's channels are independent from those of Bob, each node receives only the signals intended for it.

B. Orthogonal Blinding

As mentioned before, the zero-forcing approach requires Alice to have Eve's channel information. In practical scenarios a passive eavesdropper does not share this information. Nevertheless, Alice knows the channel to Bob so that she can optimally transmit data to him using as many spatial dimensions as Bob receive dimensions has. Alice uses additional dimensions to transmit artificial noise to the null-space of Bob's channel. Since the null-space is orthogonal to Bob's channel, he does not receive the noise. However, any other node in Alice's vicinity, receives a superposition of noise and data. As long as the received noise is powerful enough, Eve is not able to demodulate the QAM symbols she receives.

Following [3], we use the so-called Gram-Schmidt algorithm [9] to compute channels orthonormal to those of Bob. Once we have Bob's normalized channels as well as the orthonormal channels, we combine them into a single channel matrix. Then, we build a zero-forcing filter to transmit into both Bob's spatial dimensions and the orthogonal ones.

First, we normalize each row $H_{\mathcal{A}\to\mathcal{B},r}$ from $H_{\mathcal{A}\to\mathcal{B}}$:

$$\beta_r = ||H_{\mathcal{A}\to\mathcal{B},r}|| = \sqrt{\langle H_{\mathcal{A}\to\mathcal{B},r}, H_{\mathcal{A}\to\mathcal{B},r}\rangle} \tag{15}$$

$$H'_{\mathcal{A}\to\mathcal{B},r} = \frac{H_{\mathcal{A}\to\mathcal{B},r}}{\beta_r} \tag{16}$$

Then, we create a $(T_\mathcal{A} - R_\mathcal{B}) \times T_\mathcal{A}$ matrix H_{rnd} of uniformly distributed random complex numbers, where $T_\mathcal{A}$ and $R_\mathcal{B}$ are the amounts of Alice's and Bob's antennas. We again take each row $H_{\text{rnd},r}$ from H_{rnd} and subtract the projection onto previously normalized channels:

$$\begin{aligned}\hat{H}_{\text{rnd},r} = H_{\text{rnd},r} &- \sum_{j=1}^{r-1}\left\langle H'_{\text{rnd},j}, H_{\text{rnd},r}\right\rangle H'_{\text{rnd},j}\\ &- \sum_{j=1}^{R_\mathcal{B}}\left\langle H'_{\mathcal{A}\to\mathcal{B},j}, H_{\text{rnd},r}\right\rangle H'_{\mathcal{A}\to\mathcal{B},j}\end{aligned} \tag{17}$$

Then, normalizing again:

$$H'_{\text{rnd},r} = \frac{\hat{H}_{\text{rnd},r}}{||\hat{H}_{\text{rnd},r}||} = \frac{\hat{H}_{\text{rnd},r}}{\sqrt{\left\langle \hat{H}_{\text{rnd},r}, \hat{H}_{\text{rnd},r}\right\rangle}} \tag{18}$$

We combine the resulting normalized row vectors $H'_{\mathcal{A}\to\mathcal{B},r}$ and $H'_{\text{rnd},r}$ into matrices $H'_{\mathcal{A}\to\mathcal{B}}$ and H'_{rnd}, where each row in H'_{rnd} is orthogonal to any other row in H'_{rnd} and to every row in $H'_{\mathcal{A}\to\mathcal{B},r}$. By combining $H'_{\mathcal{A}\to\mathcal{B}}$ and H'_{rnd} into a single matrix, we can thus calculate Alice's zero-forcing transmit filter $F_{\mathcal{A},\text{TX}}$, including the normalization due to transmit power limitations:

$$F_{\mathcal{A},\text{TX}} = \frac{1}{\alpha_{\text{norm}}}\begin{pmatrix}H'_{\mathcal{A}\to\mathcal{B}}\\ H'_{\text{rnd}}\end{pmatrix}^{-1} \tag{19}$$

Using this transmit filter, Bob receives only the signal intended for him, $D_{\mathcal{A}\to\mathcal{B}}$, including normalizations, and Eve receives a superposition of data signal and artificial noise AN:

$$\begin{pmatrix}\mathcal{B}\\ \mathcal{E}\end{pmatrix} = \begin{pmatrix}H_{\mathcal{A}\to\mathcal{B}}\\ H_{\mathcal{A}\to\mathcal{E}}\end{pmatrix} \cdot F_{\mathcal{A},\text{TX}} \cdot \begin{pmatrix}D_{\mathcal{A}\to\mathcal{B}}\\ \text{AN}\end{pmatrix} \tag{20}$$

$$\mathcal{B} = \alpha_{\text{norm}}^{-1} \cdot \beta^{-1} \cdot D_{\mathcal{A}\to\mathcal{B}} \tag{21}$$

$$\mathcal{E} = \gamma_1 D_{\mathcal{A}\to\mathcal{B}} + \gamma_2 \text{AN} \tag{22}$$

where β is a diagonal matrix, whose elements are the normalization factors $\beta_1, \ldots, \beta_{R_\mathcal{B}}$ from Equation 15. To extract $D_{\mathcal{A}\to\mathcal{B}}$, Bob needs the following filter:

$$\hat{F}_{\mathcal{B},\text{RX}} = \alpha_{\text{norm}} \cdot \beta \tag{23}$$

Between the channel estimation phase leading to $H_{\mathcal{A}\to\mathcal{B}}$ and the data transmission phase, Bob's channel can change to $\hat{H}_{\mathcal{A}\to\mathcal{B}}$ [26]. To compensate for this change, Bob's receive filter contains a correction matrix based on those two channel measurements:

$$F_{\mathcal{B},\text{RX}} = \hat{F}_{\mathcal{B},\text{RX}} \cdot H_{\mathcal{A}\to\mathcal{B}} \cdot \hat{H}^{-1}_{\mathcal{A}\to\mathcal{B}} \tag{24}$$

In the following section we describe the design of Eve's receive filter used to attack the system.

V. Known-Plaintext Attack

Eve's goal is to extract as much transmitted data as possible from her signal reception. Therefore, she deploys a filter used to separate the transmitted data from the artificial noise. Assuming that Eve had full system knowledge, the ideal receive filter would be $F_{\mathcal{E},\text{RX}}$:

$$\underbrace{F^{-1}_{\mathcal{A},\text{TX}} \cdot H^{-1}_{\mathcal{A}\to\mathcal{E}}}_{F_{\mathcal{E},\text{RX}}} \cdot \underbrace{H_{\mathcal{A}\to\mathcal{E}} \cdot F_{\mathcal{A},\text{TX}} \cdot \overbrace{\begin{pmatrix}D_{\mathcal{A}\to\mathcal{B}}\\ \text{AN}\end{pmatrix}}^{D_\mathcal{A}}}_{\mathcal{E}} \tag{25}$$

In a practical scenario, we can assume that Eve knows the channel from Alice to Eve. Hence, Eve can calculate $H^{-1}_{\mathcal{A}\to\mathcal{E}}$. Alice's transmit filter $F_{\mathcal{A},\text{TX}}$ is, however, based on her channel $H_{\mathcal{A}\to\mathcal{B}}$ to Bob that is kept secret and thus not available to Eve.

To still extract the data signal $D_{\mathcal{A}\to\mathcal{B}}$, Eve can estimate $F_{\mathcal{E},\text{RX}}$. Consequently, she trains an adaptive filter using her partial knowledge of $D_{\mathcal{A}\to\mathcal{B}}$. Moreover, we do not need to

estimate all rows in $F_{\mathcal{E},\text{RX}}$, as $D_{\mathcal{A}\to\mathcal{B}}$ is the output signal of only the first R_B rows of $F_{\mathcal{E},\text{RX}}$. Hence, our filter estimate $\hat{F}_{\mathcal{E},\text{RX}}$ is an $R_B \times R_{\mathcal{E}}$ matrix; its output $\hat{D}_{\mathcal{A}\to\mathcal{B}}$ is an estimate of $D_{\mathcal{A}\to\mathcal{B}}$:

$$\hat{D}_{\mathcal{A}\to\mathcal{B}} = \hat{F}_{\mathcal{E},\text{RX}} \cdot \mathcal{E} \tag{26}$$

Having knowledge of the symbols at indices l in a transmitted OFDM frame allows Eve to calculate the error between the filter output and the target symbols.

$$\begin{aligned} e(l) &= D_{\mathcal{A}\to\mathcal{B}}(l) - \hat{D}_{\mathcal{A}\to\mathcal{B}}(l) \\ &= D_{\mathcal{A}\to\mathcal{B}}(l) - \hat{F}_{\mathcal{E},\text{RX}} \cdot \mathcal{E}(l) \end{aligned} \tag{27}$$

To improve the filter estimate $\hat{F}_{\mathcal{E},\text{RX}}$, Eve aims at minimizing the mean square error $\mathsf{E}|e_r|^2$ for each row $r \in [1, R_{\mathcal{B}}]$:

$$\min_{\hat{F}_{\mathcal{E},\text{RX}}} \mathsf{E}|e_r|^2 = \min_{\hat{F}_{\mathcal{E},\text{RX}}} \mathsf{E}\left| D_{\mathcal{A}\to\mathcal{B},r} - \hat{F}_{\mathcal{E},\text{RX},r} \cdot \mathcal{E} \right|^2 \tag{28}$$

Iterative training algorithms that fulfill this requirement already exist. We apply the Least Mean Squares (LMS) and Normalized Least Mean Squares (NLMS) algorithms to our filter-training problem. The rationale behind this procedure can be found in [22]. The complexity of both algorithms linearly depends on the number of Eve's receive antennas (according to [22]).

A. Least Mean Squares

The LMS algorithm is defined as follows:

$$\hat{F}_{\mathcal{E},\text{RX}}(i+1) = \hat{F}_{\mathcal{E},\text{RX}}(i) + \mu_{\text{LMS}} \cdot e(l_i) \cdot \mathcal{E}^{\text{H}}(l_i) \tag{29}$$

where

- $\hat{F}_{\mathcal{E},\text{RX}}(i)$ is the filter estimate in the i-th iteration
- μ_{LMS} is the step-size
- $\mathcal{E}^{\text{H}}(l_i)$ is the complex conjugate transpose of $\mathcal{E}(l_i)$
- $e(l_i)$ is the error when applying $\hat{F}_{\mathcal{E},\text{RX}}(i)$
- l_i is the index of the i-th known-plaintext symbol
- $i \in [0, \text{length of } l]$
- $\hat{F}_{\mathcal{E},\text{RX}}(0)$ is the initial guess, e.g., zero vector

B. Normalized Least Mean Squares

The Normalized Least Mean Squares (NLMS) algorithm is similar to the LMS algorithm, but the step-size is normalized according to the currently received training symbol to make the filter less dependent on the energy of the latter:

$$\hat{F}_{\mathcal{E},\text{RX}}(i+1) = \hat{F}_{\mathcal{E},\text{RX}}(i) + \mu(i) \cdot e(l_i) \cdot \mathcal{E}^{\text{H}}(l_i) \tag{30}$$

$$\mu(i) = \frac{\mu_{\text{NLMS}}}{\epsilon + ||\mathcal{E}(l_i)||^2} \tag{31}$$

where

- $\mu(i)$ is the step-size at the i-th iteration
- μ_{NLMS} is the iteration independent part of the step-size
- $||\mathcal{E}(l_i)||^2$ is the quadratic norm of $\mathcal{E}$ to stabilize the algorithm for strongly varying input data
- ϵ is a small value to avoid divisions by zero

Both algorithms are used to train the adaptive filter in our attack scenario. Their evaluation through simulation and real-world measurements are described in the next section. Even though we use the unfiltered receive signal $\mathcal{E}$ in our derivations, $\mathcal{E}$ can be replaced by the prefiltered:

$$\mathcal{E}' = H^{-1}_{\mathcal{A}\to\mathcal{E}} \cdot \mathcal{E} \tag{32}$$

leading to an $R_{\mathcal{B}} \times T_{\mathcal{A}}$ filter matrix $\hat{F}'_{\mathcal{E},\text{RX}}$.

VI. Experimental Evaluation

We now describe the experimental evaluation of our known-plaintext attack model for physical layer security schemes by means of simulation and testbed experimentation.

First, we give an overview of the technical parameters in Section VI-A, then we investigate the key trade-offs involved in our attack. Using simulation, we cover a wide parameter range and establish the operating area for working attacks. In Section VI-B, we analyze the effect of an increase in artificial noise power by Alice on Bob. In Sections VI-C and VI-D we investigate how Eve can optimize her attack performance. We analyze the filter tuning parameters (step-size μ, see Section VI-C) and the applied filter adaptation (LMS vs. NLMS, see Section VI-D) for a wide range of channel conditions. In Section VI-E a prefiltering approach increases the efficacy of the attack if Eve has channel knowledge. The influence of Eve's antenna count is discussed in Section VI-F. The ability of Eve to improve her efficacy over Bob is discussed in Section VI-G in case the channel from Alice to Eve, $H_{\mathcal{A}\to\mathcal{E}}$, is better than the channel from Alice to Bob, $H_{\mathcal{A}\to\mathcal{B}}$.

Second, by means of experimentation using the WARP software-defined radio platform, we validate the simulation results in practice for selected realistic parameter sets. In Section VI-H, we study the effects of real world channels on our attack methodology and identify subchannel conditions that constrain both the secrecy scheme as well as our known-plaintext analysis. In Section VI-I, we analyze the carrier-dependent attack performance. In Section VI-J, we study the convergence behavior of the employed filtering techniques in practice. Finally, in Section VI-K, we show the reduction in practical secrecy rate of the communication between Alice and Bob.

We summarize our experimental findings in Tables I and II for the simulations and the practical experiments. An extensive discussion follows in Section VII.

A. Technical parameters and test setup

As thoroughly described in Sections II and III, our three nodes Alice, Eve and Bob are multi-antenna nodes using OFDM transmitters to abstract from the physical channel. Without loss of generality, we focus our evaluation on a setup where Alice and Eve have two antennas, and Bob has one. Therefore, Alice has one spatial dimension to transmit to Bob's receive dimension and an additional dimension for artificial noise transmission. Eve has as many antennas as Alice and thus the minimum number of antennas required to perform our attack. To adjust Eve's disturbance by artificial noise AN, Alice

varies the ratio of transmitted AN to transmitted data signal—Noise to Data Ratio (NDR). As Alice's transmit power is limited, a higher NDR reduces the available power to transmit the data signal:

$$D_{\mathcal{A}} = \frac{1}{\text{NDR}+1}\left(\frac{D_{\mathcal{A}\to\mathcal{B}}}{\text{NDR}\cdot\text{AN}}\right) \tag{33}$$

During our simulations, we vary the amount of AWGN η added by the wireless channel. The SNR at Bob varies according to the amount of received signal power, as well as the AWGN: $\text{SNR} = 20\log_{10}(\mathcal{B}) - 20\log_{10}(\eta)$. Due to Alice's transmit power limitation, increasing the NDR reduces the power of the signal transmitted to Bob. To measure the effect of different NDRs at constant AWGNs levels, we reference the SNR_{TX} at the transmitted power: $\text{SNR} = 20\log_{10}(\mathcal{A}) - 20\log_{10}(\eta)$. This leads to results comparable to our simulations.

Our OFDM transceivers work on 40 MHz wide channels in the 2.4 GHz band. The cut-off frequency of the receiver's baseband filters is 18 MHz (36 MHz bandwidth due to IQ-demodulation). The OFDM has K = 64 subchannels with a subchannel spacing of 625 kHz, which is sufficient for the coherence bandwidth of the channels in our indoor scenario illustrated in Figure 2. Channel measurements showed that most of the transmitted energy is concentrated on three to five taps in the channel impulse response of the received signal. Therefore, we simulate similar channels during our simulations. Each transmitted wireless frame consists of an 802.11a short preamble to detect the start of the frame, followed by ten pilot symbols for channel estimation. Payload including packets additionally contain 150 payload symbols (Alice's filter output). The cyclic prefix length to avoid ISI equals 12 samples. Carrier Frequency Offset (CFO) correction at Bob and Eve is avoided by synchronizing the Radio Frequency (RF) clock generators by cable. Note that practical CFO correction algorithms are available [6]; however, they would have added unnecessary complexity to our experiments. To prevent gain fluctuations and to increase the reproducibility of our results, we opted for manual instead of automatic gain control.

The transmitted data symbols $D_{\mathcal{A}\to\mathcal{B}}$ are normalized 4-QAM symbols. We choose the SER to compare the performance of our adaptive filter to Bob's receive performance, since it is a practical measure of the amount of data that can be correctly extracted at both Bob and Eve. In simulation, we run 100 Monte Carlo experiments with different channels, calculate the SERs and average over 100 experiments, 64 subchannels and 150 OFDM symbols.

B. Effect of Alice's artificial noise on Bob

The higher the NDR is, the lower is the signal energy received at Bob (Equation 33). Additionally, the AWGN degrades Bob's reception performance. Figure 3 illustrates Bob's SER over Alice's NDR for different SNR_{TX}, which represents the amount of AWGN. We clearly observe that the SNR_{TX} has a major influence on Bob's SER. If the NDR increases, the SER approaches 75 %, which is equal to guessing uniformly distributed 4-QAM symbols.

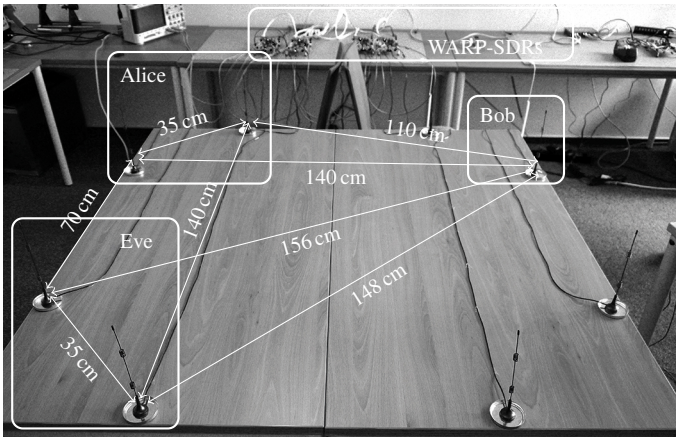


Fig. 2. Antenna setup for practical measurements.

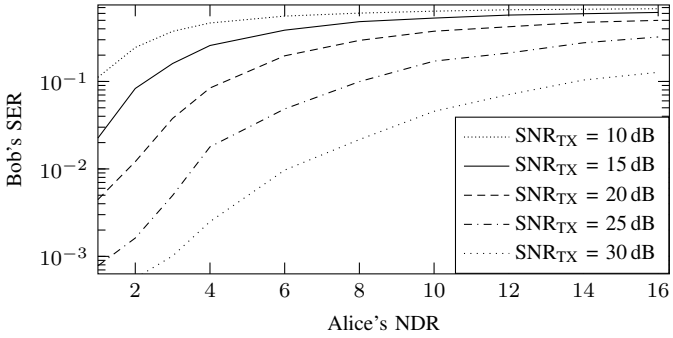
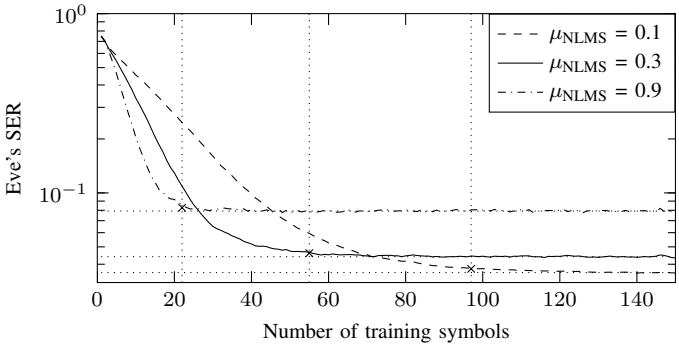
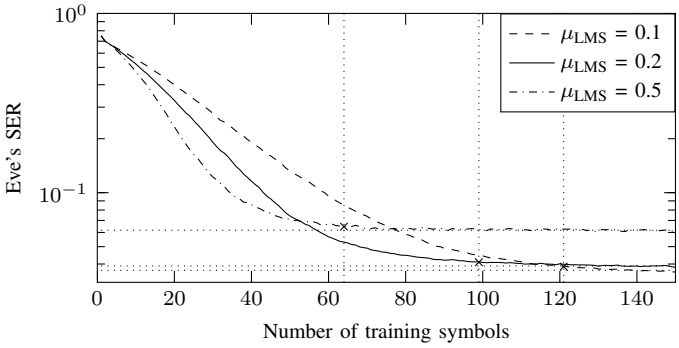


Fig. 3. Bob's SER over Alice's NDR for several SNR_{TX}.



(a) NLMS



(b) LMS

Fig. 4. Eve's SER over the number of used training symbols for multiple step-sizes μ of the (N)LMS algorithm. SNR_{TX} = 30 dB and Alice's NDR = 4.

C. Convergence behavior of Eve's filter

When using adaptive filters, the step-size μ influences how fast a filter converges to its targeted ideal filter. Small step-sizes lead to slower convergence but also to smaller deviations from the ideal filter, whereas high step-sizes lead to faster convergence with higher errors and potentially no convergence. To measure the filter's training performance, we choose to compare SERs ranging from 0 % (for no errors) to 75 % (for randomly guessing 4-QAM symbols). A non converging filter just maximizes the SER. Furthermore, this metric can be compared to Bob's receive performance.

In Figure 4, we illustrate how the SER reduces when the number of available training symbols increases. We regard a filter as convergent, when the SER differs less than 5 % from the average SER, and there are 130 to 150 available training symbols. This averaged SER is the achievable SER of an adaptive filter with a given step-size. The points of convergence are marked as crosses in Figure 4. We use these convergence points in the following to compare the performance of different adaptive filter settings. In Figure 5 we illustrate how the chosen NLMS step-size influences the convergence characteristics for different channel $\mathrm{SNR}_{\mathrm{TX}}$. Small step-sizes drastically increase the convergence time but also allow a minimum SER. A μ_{NLMS} of 0.3 is a good compromise for our scenario.

D. Choosing Eve's adaptive filter technique

In Section V, we introduced the two training algorithms LMS and NLMS. Figure 4 illustrates their convergence characteristics. Regarding comparable SERs at convergence, we observe that the NLMS algorithm converges faster (with respect to required training symbols) than the LMS algorithm. The normalization of the filter update allows the application of higher step-sizes in the NLMS filter, which reduce the convergence time. Figure 6 illustrates the training performance for different NDRs. We observe that the LMS algorithm requires more training symbols than the NLMS algorithm to achieve a similar SER at a certain NDR. Due to the advantages of the NLMS algorithm, we use it in what remains of this paper.

E. Prefiltering at Eve

As described at the end of Section V-B, Eve can use her channel estimate $H_{\mathcal{A}\to\mathcal{E}}$ to enhance the filtering performance. Prefiltering is applied for all of our presented results. In Figure 7, we illustrate the advantage of prefiltering at Eve. Even though similar SERs can be achieved, the convergence time without prefiltering is significantly higher than with prefiltering. Prefiltering maps Eve's spatial receive dimensions to Alice's transmit dimensions, which reduces the complexity of the filter training. For higher NDR the prefiltering advantage decreases.

F. Effect of multiple receive antennas

Figure 8 shows our simulation results for many-antenna eavesdroppers on high and low $\mathrm{SNR}_{\mathrm{TX}}$ channels. The SER decreases if the number of antennas increases. Hence, many antennas are useful if Eve has a noisy channel. Prefiltering generally leads to a faster filter convergence.

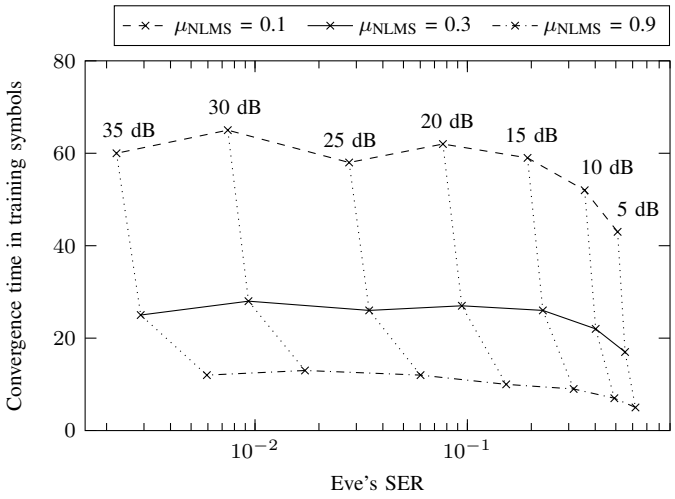


Fig. 5. Linking Eve's SER to the convergence time of NLMS filters with different step-sizes μ and NDR = 2. The black dotted lines refer to equal $\mathrm{SNR}_{\mathrm{TX}}$ values.

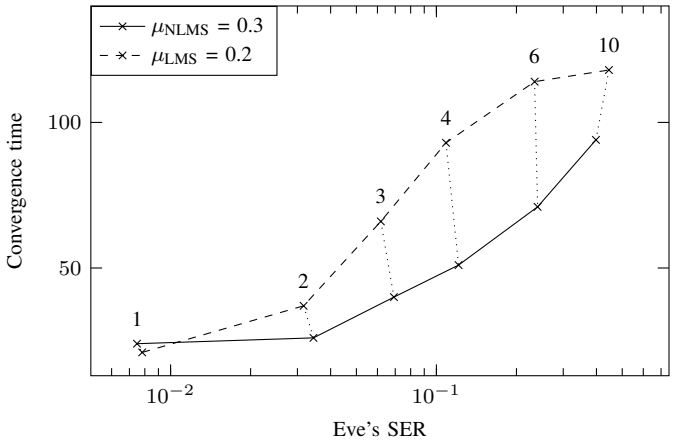


Fig. 6. Linking Eve's SER to the convergence time (LMS dashed; NLMS solid). The black dotted lines refer to equal NDR values. $\mathrm{SNR}_{\mathrm{TX}}$ = 25 dB.

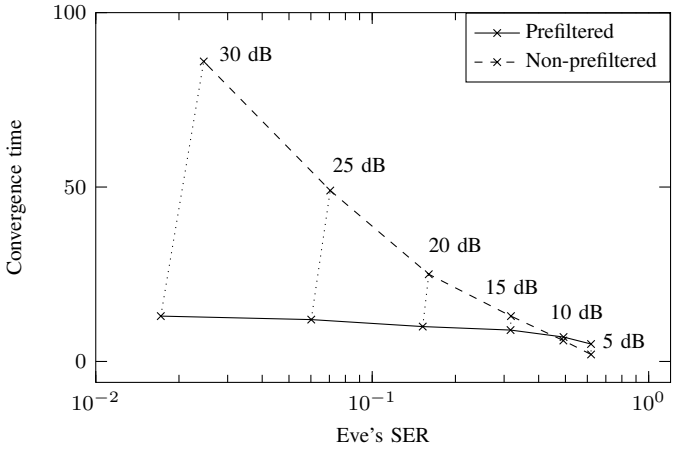


Fig. 7. NLMS without prefiltering (dashed) vs. NLMS with prefiltering (solid). The black dotted lines refer to equal $\mathrm{SNR}_{\mathrm{TX}}$ values. NDR = 2 and μ_{NLMS} = 0.9.

TABLE I. SUMMARY OF THE MAJOR CONTRIBUTIONS OF THIS PAPER VALIDATED IN SIMULATION.

Experiment	Section	Conclusion
Effect of artificial noise	Sec. VI-B	Since Alice is power-limited, increasing her artificial noise implies decreasing the power of the data signal. Hence, the SER at *both* Eve and Bob increases, but is worse at Eve for similar SNRs.
Convergence behavior	Sec. VI-C	The smaller the step-size μ, the better the SER, but the more known plaintext is needed. For example, $\mu = 0.9$ slashes convergence time to less than half of $\mu = 0.3$, but doubles the SER.
Adaptive filter technique	Sec. VI-D	The LMS algorithm converges slower and requires more training symbols than NLMS, e.g., for NDR = 4, NLMS requires only about half the amount of training symbols of LMS.
Prefiltering at Eve	Sec. VI-E	If Eve knows the channel to Alice, she can reduce the convergence time. For a 25 dB channel, $\mu = 0.9$, NDR = 2, prefiltering divides convergence time by more than three at comparable SER.
Multiple receive antennas	Sec. VI-F	Especially, on low SNR_{TX} channels multiple eavesdropper antennas reduce Eve's SER.
Eve's attack performance	Sec. VI-G	Eve can severely compromise secrecy if she has a good channel to Alice. If the SNR_{TX} to Bob is 15 dB and to Eve 25 dB, the secrecy rate is *negative*, i.e., Eve can extract more data than Bob.

G. Assessing Eve's attack performance

When theoretically analyzing physical layer security schemes, the secrecy rate [25] is used to measure how much more data Bob receives compared to how much Eve can extract. This secrecy rate definition is not directly applicable to system simulations and measurements. Therefore, we define the practical secrecy rate S_{prac} to compare Eve's advantage over Bob based on their SERs and a maximum 4-QAM SER of 75 %:

$$S_{\text{prac}} = \frac{\text{SER}_{\mathcal{E}} - \text{SER}_{\mathcal{B}}}{75\,\% - \text{SER}_{\mathcal{B}}} \tag{34}$$

Depending on Bob's and Eve's channel quality, the practical secrecy rate can be negative if Eve's channel has a higher SNR_{TX} than that of Bob's channel. That is possible as Eve can freely position her antennas. An exemplary result is illustrated in Figure 9.

H. Influence of radio hardware

To analyze the applicability of the security scheme and our attack in realistic environments, we implemented Alice, Bob and Eve on separate WARP nodes using WARPLab. The exemplary setup in our lab is illustrated in Figure 2. We additionally performed experiments with antennas in multiple office rooms (roughly 10 m apart), but we did not experience significant changes in the filter training performance, apart from a drop in SNR and a need for increased receive gains. Therefore, our evaluation concentrates on the lab scenario. We first compare Bob's SER measurements to the simulations. According to Figure 10, the channel quality lies somewhere between 20 and 25 dB, at least for NDR > 4. For NDR < 4, the simulatively determined SERs are not achieved. In the following, we explain the deviation of Bob's SERs.

I. Effect of real-world channels

First, we illustrate Bob's SER over subchannels in Figure 11. The error rate depends on the subchannels; here, especially subchannels 25 to 37 exhibit high SERs. To explain this effect, we analyze a set of 100 channel measurements in Figure 12. The channel coefficients are stable in most experiments; only the subchannels 25 to 37 constantly change. These subchannels lie outside of the cut-off regions of our baseband receive filter. Additionally, the high error rate at subchannel 7 can be explained by a deep fade. Removing subchannels with extraordinary high SERs, reduces Bob's SER and allows it to converge against the simulated 25 dB channel, as illustrated in Figure 13.

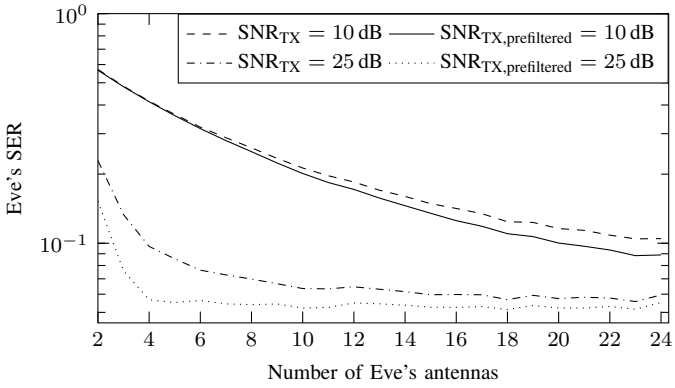


Fig. 8. Eve's SER in dB after 100 training iterations on one subchannel for different number of receive antennas with and without prefiltering. NDR = 10, μ_{NLMS} = 0.3.

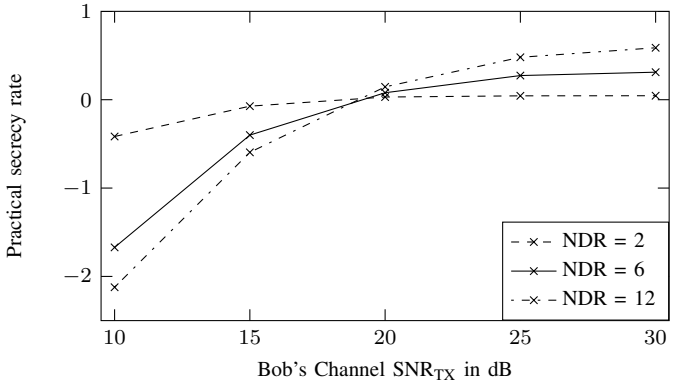


Fig. 9. Practical secrecy rates for different NDRs and SNR_{TX} at Bob. Eve's SNR_{TX} is fixed to 25 dB and she uses the NLMS algorithm with μ = 0.3.

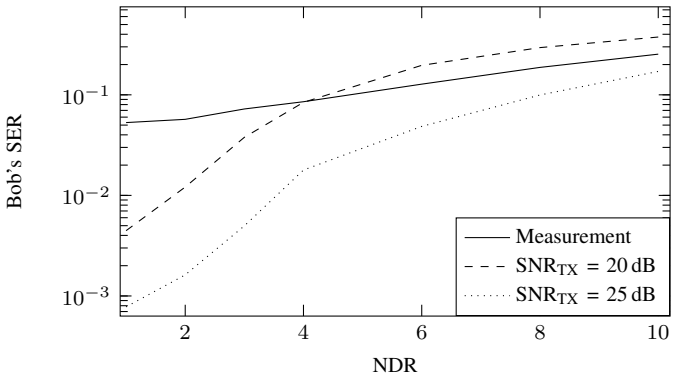


Fig. 10. Practical and simulation results of Bob's SER over Alice's NDR

After omitting the aforementioned subchannels, the measurements still deviate from the simulations—especially in the low NDR regions. Figure 12 hints at an explanation. For most experiments the channel coefficients are very similar. Nevertheless, certain experiments show severe outliers that deviate from most channel measurements. These outliers are either due to interfering transmissions, or due to wrong estimations of the frame preamble, which leads to erroneous OFDM demodulations that affect both Bob and Eve.

In Figure 14 we illustrate symbol errors at selected subchannels. The rows correspond to the 100 experiments, the columns to the 150 symbols per frame. Subchannel 53 has mostly randomly distributed symbol errors. Some experiments, however, show clustered errors, that can be explained by the aforementioned problems. On subchannel 7 we observe that error clustering especially occurs at certain experiments and the deep fade at this subchannel seems to emphasize the probability of error clusters. To further approximate the simulated results, we decide to ignore certain experiments that introduce symbol error clusters, as this effect is not considered in the simulations.

Figure 15 illustrates Bob's SER over various replications of the experiment. The SER is constantly low for experiments where the errors are not clustered but randomly distributed. Error clusters, however, lead to outliers. Removing experiments with outliers, allows to further reduce Bob's SERs, so that the measured results are more similar to the simulated results, which is illustrated in Figure 16.

J. Convergence behavior in practice

We illustrate Eve's practical convergence behavior in Figure 17 (compare Figure 4 for simulation results). We observe, that Eve's filter takes longer to converge and that Eve's SER is higher in the measurement compared to the simulation. The reasons are twofold. On the one hand, the plots in Figure 4 are based on channels with an $\mathrm{SNR_{TX}}$ of 30 dB, which is higher than the $\mathrm{SNR_{TX}}$ we approached when analyzing Bob's channel in Section VI-H. On the other hand, we considered all measurements in this evaluation, as error clusters and high error rate subchannels are part of our practical setup, i.e., outliers are not removed.

K. Eve's attack performance in practice

Figure 18 summarizes the attack performance of Eve in terms of convergence time and SER. For larger step-sizes the convergence time is smaller, but the SER becomes larger, as predicted by our simulations in Section VI-C. Figure 18 also shows Bob's results in terms of SER. For larger values of NDR the SER worsens for both Eve and Bob. Bob's convergence time is constantly zero, as he does not need to train any filter to suppress noise. In this experiment, Bob and Eve were placed close to each other, leading to similar channel SNRs. Hence, Bob's SER is better than Eve's.

This is directly reflected in Figure 19, which shows the practical secrecy rate between Alice and Bob. Since Bob performs better than Eve, the rate is always positive, as opposed to our experiment in Section VI-G. The practical secrecy rate improves with increasing NDR, as the additional noise makes it more difficult for Eve to decode symbols

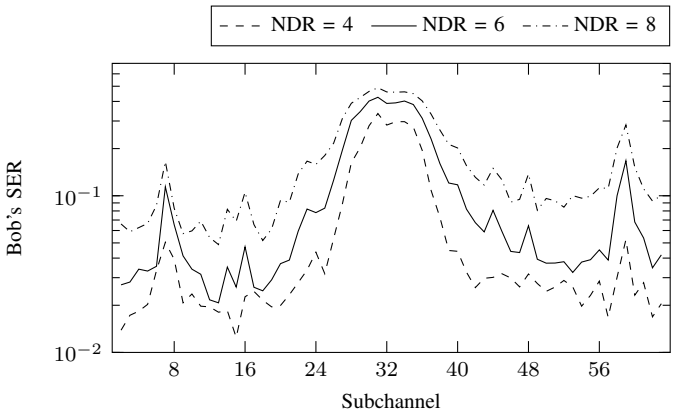


Fig. 11. Bob's SER over different subchannels for different NDR values.

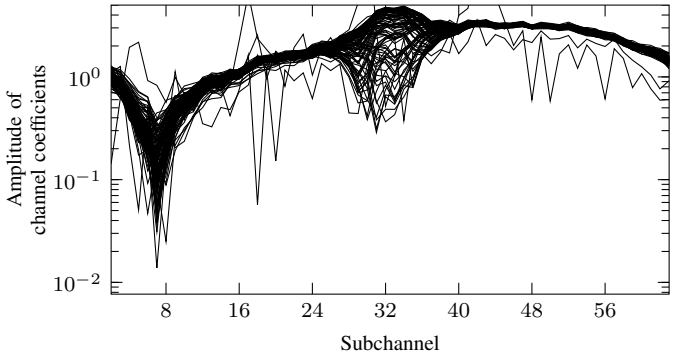


Fig. 12. Amplitudes of the channel coefficients between Alice and Bob plotted for 100 experiments carried out in a time-frame of 30 minutes.

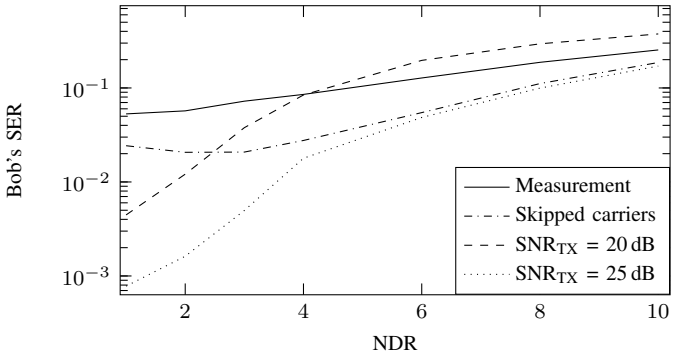
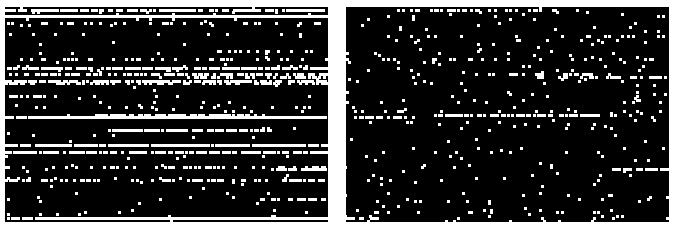


Fig. 13. Bob's SER over Alice's NDR in practice with all carriers and skipping carriers 25 to 37. Additionally, two simulations for $\mathrm{SNR_{TX}} = \{20, 25\}$ dB are shown.



(a) Subchannel 7 (b) Subchannel 53

Fig. 14. Bob's symbol errors (white pixels represent errors; black pixels, absence of errors). Experiments are on the vertical axis; symbols per frame on the horizontal one. NDR = 6.

TABLE II. SUMMARY OF THE MAJOR CONTRIBUTIONS OF THIS PAPER VALIDATED IN PRACTICAL EXPERIMENTATION.

Experiment	Section	Conclusion
SDR hardware limitations	Sec. VI-H	Bob's SER is worse compared to simulation due to errors at the receive filter's cut-off regions. After removing these outliers, simulation and real-world results come closer, but still differ.
Real-world channels	Sec. VI-I	Due to interference, deep fades and wrong preamble detections, some experiments contain clustered errors, increasing Bob's SER. After removing them, simulation and practice match nearly perfect.
Convergence behavior	Sec. VI-J	Simulation and practice also match regarding filter convergence. Eve can trade off between fast convergence (small μ) yet higher SER or slow(er) convergence (large μ) yet low SER.
Attack performance	Sec. VI-K	Our attack drastically reduces the practical secrecy rate between Alice and Bob. Increasing the NDR does not lead to a linear increase in secrecy rate, which significantly limits the practical strength of existing implementations of orthogonal blinding.

correctly. A similar effect is caused when increasing the step-size, since the error when training the filter becomes larger. Finally, Figure 19 shows that the practical and simulation results for $\mu = 0.3$ match nearly perfect, which again validates our experiments. We conclude that our attack on orthogonal blinding is successful and exhibits a good performance.

VII. DISCUSSION

Our performance evaluation in simulation and practice shows that the known-plaintext attack model can be successfully translated from cryptoanalysis to the analysis of physical layer security schemes. We now discuss its benefits and limitations, with a special focus on its application to current state-of-the-art systems.

In our evaluation, we use the SER as our **main metric** to quantize the performance of our attack. We also study a number of additional metrics, such as the convergence time. However, we consider the SER to be most representative, as it fully captures the main goal of our attack, i.e., correctly decoding the symbols sent by Alice at Eve. Moreover, it is agnostic to our filtering approach. Thus, it allows us to compare our results to other attacks based on other techniques.

While the SER captures the efficacy of our attack, **speed** is also a critical factor, since filter training should converge fast in order to require as less known plaintext as possible. A key aspect with respect to speed is the chosen type of filter and its parameters. Our experiments show that the normalization of the NLMS filter can halve the convergence time of the LMS, which makes it the technique of choice for our attack. Moreover, it is well suited for scenarios where the filter input originates from an antenna array.

The speed of our attack also determines its **applicability to real-world systems**. Hence, we present a brief example on how our attack would work in an 802.11ac [19] system that applies orthogonal blinding on the OFDM symbols in the Physical-layer Service Data Unit (PSDU). We assume a bandwidth of 20 MHz divided into 64 subchannels (48 usable for data transmission). Additionally, we consider Binary Phase Shift Keying (BPSK) with 1/2 Forward Error Correction (FEC). If Eve can guess the 30 byte Wi-Fi Media Access Control (MAC) header as well as the 40 byte IPv6 header, she knows the plaintext of the first 23 to 24 OFDM symbols on each subchannel. Hence, an NLMS filter with step-size 0.9 leads to a SER below 10 %, according to Figure 4 (a) with $\mathrm{SNR_{TX}}$ = 30 dB and NDR = 4. Both headers sum up to 70 byte, which is 3 % of the 2304 byte MAC Service Data Unit (MSDU) [1] plus 30 byte MAC header.

While our attack would be successful in the aforementioned example, Alice has a number of options to make the attack

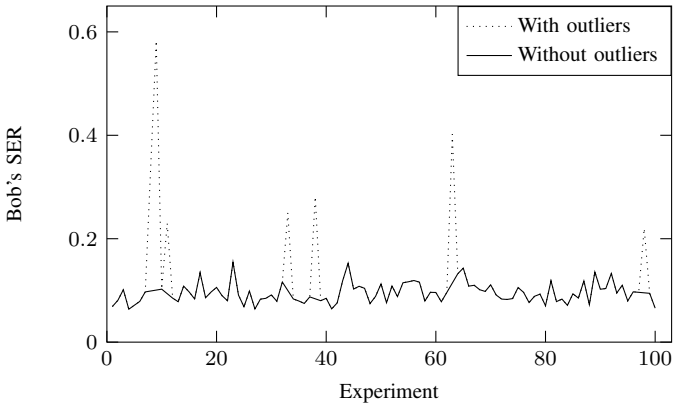


Fig. 15. Bob's SER in several experiments (NDR = 8).

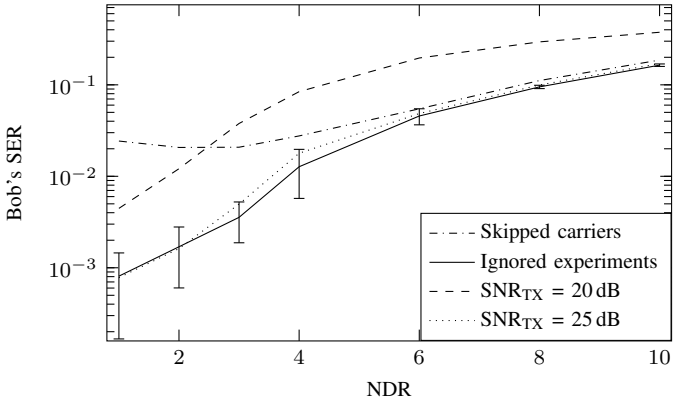


Fig. 16. Bob's SER over Alice's NDR skipping carriers 25 to 37 compared to additionally ignoring outliers. Two simulations for $\mathrm{SNR_{TX}}$ = {20, 25} dB are also shown.

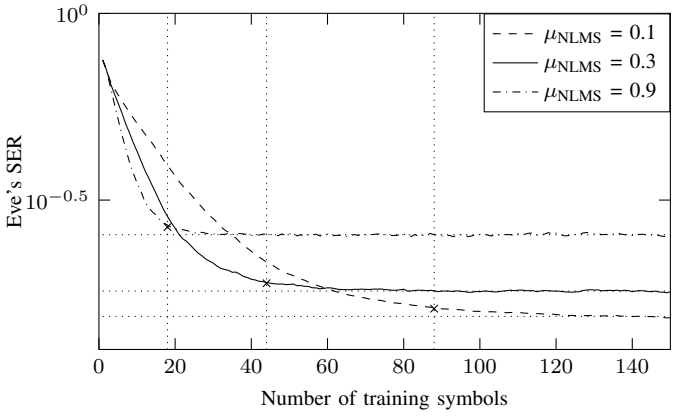


Fig. 17. Eve's SER over the number of used training symbols for multiple step-sizes μ (NLMS; NDR = 4).

harder for Eve. First, she could use a higher order modulation scheme, which translates into smaller headers in terms of symbols and thus less known plaintext. However, Eve can **maximize the knowledge** she gets from the amount of symbols she can guess by exploiting the coherence bandwidth of the channel. Specifically, neighboring subcarriers can typically be considered to be similar. Therefore, the resulting filters are also highly similar for both, which means that the known symbols on both subcarriers can be combined to train the filter.

Second, Alice could increase the NDR. Still, as shown in our evaluation, this does not only lead to a worse SER at Eve but also at Bob. To keep a certain SER, Alice would have to increase the **FEC**, which in turn implies more known symbols. Eve could use these additional known symbols to train her filter with a smaller step-size and thus reduce her SER. Furthermore, she could exploit slowly changing channels in order to train her filter over **consecutive frames**, which reduces the amount of known symbols required per frame. Also, if Eve can force Alice to send a specific well-known frame, Eve could train her filter on that frame only and apply the same filter on all consecutive frames, as long as the channels are constant. This would be analog to a chosen plaintext attack.

Third, Alice could avoid the blinding of any data usable by Eve as known plaintext. However, Alice's physical layer generally does not know which **upper layer data** Eve knows. Alice could also apply **encryption** in addition to orthogonal blinding, but if Eve could get access to the ciphertext, she could use it for filter training.

VIII. Related Work

In information theory, multiple publications base on Wyner's work on the wiretap channel [25]. Wyner introduces the secrecy rate that describes how much more information the intended receiver can extract compared to an eavesdropper when communicating over wireless channels. The secrecy capacity refers to the maximum theoretically achievable secrecy rate of a channel.

This basic scheme has been extended for multiantenna scenarios, where both transmitters as well as receivers use multiple spatial dimensions [17]. Other extensions focus on the secrecy rate in multi-hop or relay scenarios [5], [14]. Our attack can be extended to such scenarios, but is not directly related to them. The authors of [5] focus on zero-forcing beamforming, where knowledge of the CSI to the eavesdroppers is required. However, this is not a realistic assumption in practical systems, since the eavesdropper typically does not disclose its CSI to well-behaved nodes. Another shortcoming of many theoretical analyses is the limitation of the transmitted data signals to Gaussian distributed waveforms. The authors of [15] relax this shortcoming by considering QAM quantized data symbols, as also used in this work. Additionally, our physical layer is based on an OFDM transmitter; its secrecy rate is analyzed in [20].

The aforementioned papers focus on finding upper bounds for the achievable secrecy rates. Thus, the CSI to the eavesdropper is assumed to be known. To overcome this limitation while still achieving positive secrecy rates, orthogonal blinding is used. The authors of [3] analyze the performance of orthogonal blinding in comparison to zero-forcing beamforming

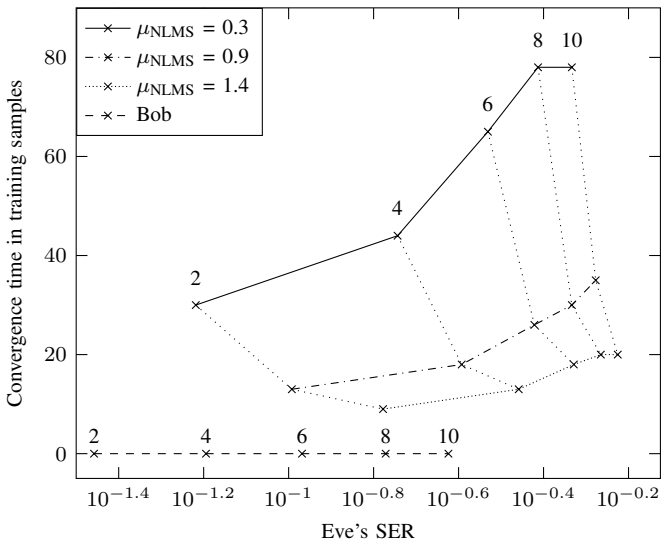


Fig. 18. Eve's SER and convergence time compared to Bob's SER. The numbers represent the NDR at each point. The dotted black lines connect points with equal NDR.

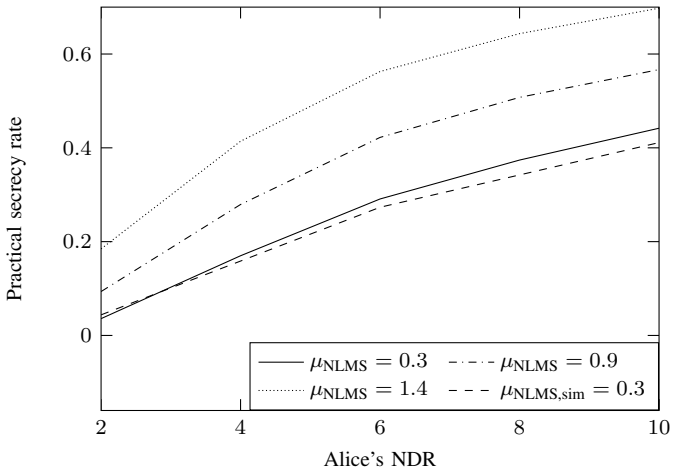


Fig. 19. Practical secrecy rates for different NLMS step-sizes μ over Alice's NDR. For comparison, a simulation with SNR_{TX} = 25 dB is also shown.

and the use of directional antennas. They validate their results in practice using software-defined radios [2], similar to our evaluation. However, they assume each eavesdropper to be equipped with only one antenna, thus reducing her spatial capabilities. The authors of [10] consider cooperation, i.e., eavesdroppers that combine their spatial dimensions to become a more powerful attacker. Colluding eavesdroppers come with the cost of communication overhead, as analyzed in [21]. Other extensions to orthogonal blinding consider delayed feedback from CSI measurements, which is key for transmit filter generation [26], and the use of a separate node to transmit the artificial noise [7].

The concept of jamming an eavesdropper is also considered in the related area of friendly jamming. The authors of [11] use jamming to protect the confidentiality of unencrypted communication of medical devices. This scheme was broken in [23] by smart placement of multiple antennas. Additionally,

jamming can also be used to perform secret key exchanges [12] or to protect a network against intruders [24].

Our work stands apart from the related work discussed in this section since—to the best of our knowledge—our approach is the first practical attack on orthogonal blinding and the first approach to apply known-plaintext attacks against physical layer security schemes.

IX. Conclusion

We present a physical layer attack model which is inspired by the concept of known-plaintext attacks from the cryptography domain. Specifically, we instantiate our model to design an attack on orthogonal blinding, which is a physical layer security scheme based on artificial noise. In a setup with transmitter Alice, receiver Bob and eavesdropper Eve, Alice sends artificial noise into a channel orthogonal to the channel of Bob. Hence, while Bob does not receive any noise, Eve cannot decode the signal since she gets a superposition of signal and noise. Our attack assumes that Eve may guess part of the data sent by Alice, such as protocol headers. We use this known plaintext to train an adaptive filter at Eve. Once the filter is trained, Eve can use it to decode the unknown data. We implement our attack on software-defined radios and additionally perform an extensive simulation to determine the operating area of our technique.

Our experiments show that our attack can successfully compromise orthogonal blinding. By carefully selecting the filter step-size μ, Eve can reduce the convergence time of the adaptive filter by a factor of two. If Eve knows her channel to Alice, the convergence of the attack can be sped up by more than a factor of four over our basic, not optimized attack. Furthermore, we demonstrate that a *negative* secrecy rate (Eve extracts more data than Bob) can be achieved if Eve has a better channel to Alice than Bob. Future work includes extending our attack with techniques such as training filters over multiple frames, and investigating how further attack models from the cryptography domain can be applied to physical layer security.

Acknowledgment

This work has been funded by the German Research Foundation (DFG) in the Collaborative Research Center (SFB) 1053 “MAKI – Multi-Mechanism-Adaptation for the Future Internet”, by the LOEWE Priority Program Cocoon and by LOEWE CASED. Many thanks go to our shepherd, Srdjan Capkun.

References

[1] *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification*. IEEE Std. 802.11, 2012.

[2] (2013) Rice university WARP project. [Online]. Available: http://warp.rice.edu

[3] N. Anand, S.-J. Lee, and E. Knightly, “Strobe: actively securing wireless communications using zero-forcing beamforming,” in *Proc. INFOCOM’12*, 2012, pp. 720–728.

[4] E. Barkan, E. Biham, and N. Keller, “Instant ciphertext-only cryptanalysis of GSM encrypted communication,” *J. Cryptol.*, vol. 21, pp. 392–429, Mar. 2008.

[5] R. Bassily and S. Ulukus, “Secure communication in multiple relay networks through decode-and-forward strategies,” *IEEE Trans. Commun., Netw.*, vol. 14, pp. 352–363, 2012.

[6] Y. S. Cho, J. Kim, W. Y. Yang, and C.-G. Kang, *MIMO-OFDM Wireless Communications with MATLAB*. John Wiley & Sons, 2010.

[7] S. Fakoorian and A. Swindlehurst, “Solutions for the MIMO gaussian wiretap channel with a cooperative jammer,” *IEEE Trans. Signal Process.*, vol. 59, pp. 5013–5022, 2011.

[8] S. R. Fluhrer, I. Mantin, and A. Shamir, “Weaknesses in the key scheduling algorithm of RC4,” in *Revised Papers SAC’01*, 2001, pp. 1–24.

[9] S. Friedberg, A. Insel, and L. Spence, *Linear Algebra*. Prentice Hall, 1989.

[10] S. Goel and R. Negi, “Secret communication in presence of colluding eavesdroppers,” in *Proc. MILCOM’05*, 2005, pp. 1501–1506.

[11] S. Gollakota, H. Hassanieh, B. Ransford, D. Katabi, and K. Fu, “They can hear your heartbeats: non-invasive security for implantable medical devices,” in *Proc. SIGCOMM’11*, 2011, pp. 2–13.

[12] S. Gollakota and D. Katabi, “Physical layer wireless security made fast and channel independent,” in *Proc. INFOCOM’11*, 2011, pp. 1125–1133.

[13] T. Jager, K. G. Paterson, and J. Somorovsky, “One bad apple: backwards compatibility attacks on state-of-the-art cryptography,” in *Proc. NDSS’13*, 2013.

[14] J. Kim, A. Ikhlef, and R. Schober, “Combined relay selection and cooperative beamforming for physical layer security,” *IEEE Trans. Commun., Netw.*, vol. 14, pp. 364–373, 2012.

[15] Z. Li, R. Yates, and W. Trappe, “Achieving secret communication for fast rayleigh fading channels,” *IEEE Trans. Wireless Commun.*, vol. 9, pp. 2792–2799, 2010.

[16] K. Nohl, E. Tews, and R.-P. Weinmann, “Cryptanalysis of the DECT standard cipher,” in *Proc FSE’10*, 2010, pp. 1–18.

[17] F. Oggier and B. Hassibi, “The secrecy capacity of the MIMO wiretap channel,” *IEEE Trans. Inf. Theory*, vol. 57, pp. 4961–4972, 2011.

[18] D. A. Osvik, A. Shamir, and E. Tromer, “Cache attacks and countermeasures: the case of AES,” in *Proc. CT-RSA’06*, 2006, pp. 1–20.

[19] E. Perahia and R. Stacey, *Next Generation Wireless LANs - 802.11n and 802.11ac*, 2nd ed. Cambridge University Press, 2013.

[20] F. Renna, N. Laurenti, and H. V. Poor, “Achievable secrecy rates for wiretap OFDM with QAM constellations,” in *Proc. VALUETOOLS’11*, 2011, pp. 679–686.

[21] W. Saad, Z. Han, T. Basar, M. Debbah, and A. Hjorungnes, “Physical layer security: coalitional games for distributed cooperation,” in *Proc. WiOPT’09*, 2009, pp. 1–8.

[22] A. H. Sayed, *Adaptive Filters*. Wiley-IEEE Press, 2008.

[23] N. O. Tippenhauer, L. Malisa, A. Ranganathan, and S. Capkun, “On limitations of friendly jamming for confidentiality,” in *Proc. S&P’13*, 2013.

[24] M. Wilhelm, I. Martinovic, J. B. Schmitt, and V. Lenders, “Wifire: a firewall for wireless networks,” in *Proc. SIGCOMM’11*, 2011.

[25] A. D. Wyner, “The wire-tap channel,” *Bell Systems Technical Journal*, vol. 54, pp. 1355–1387, 1975.

[26] Y. Yang, W. Wang, H. Zhao, and L. Zhao, “Transmitter beamforming and artificial noise with delayed feedback: secrecy rate and power allocation,” *IEEE Trans. Commun., Netw.*, vol. 14, pp. 374–384, 2012.