Wi-Auth: WiFi based Second Factor User Authentication

Syed W. Shah
School of Computer Science and Engineering
The University of New South Wales
Sydney, NSW 2052
syedwajidali.shah@unsw.edu.au

Salil S. Kanhere
School of Computer Science and Engineering
The University of New South Wales
Sydney, NSW 2052
salil.kanhere@unsw.edu.au

ABSTRACT

While second factor authentication (2FA) is now widely available, user adoption is still very low, as most of 2FA implementations require significant interaction from the user. In this paper, we present a novel 2FA system, called Wi-Auth that requires minimal participation from the user. A user after confirming her credentials with an online service, simply has to place a pre-registered secondary device in close proximity (< 2.5 inches) of the primary device from which the login attempt is being made. Wi-Auth detects the proximity of these two devices by comparing the fine-grained Channel State Information (CSI) of the ambient WiFi signals measured at the two devices. The logic being that two devices that are in such close proximity will exhibit very similar CSI characteristics. Wi-Auth uses a lightweight two-step matching algorithm to compare the two CSI measurements. We also address (for the first time in literature) the issue of targeted attacks where an attacker may be co-located with the victim. We implement Wi-Auth using commodity off-the-shelf 802.11n devices and evaluate its performance in three different practical settings including an open office, an apartment and a large meeting space. Our experiments performed at 90 different location reveal that Wi-Auth can on average achieve 94% authentication accuracy with 5% false positives and 6% false negatives. Moreover, Wi-Auth is very robust in preventing co-located attacks with a 95% attack detection accuracy.

CCS CONCEPTS

•Security and privacy →Multi-factor authentication;

KEYWORDS

Second Factor Authentication, WiFi, CSI

ACM Reference format:
Syed W. Shah and Salil S. Kanhere. 2017. Wi-Auth: WiFi based Second Factor User Authentication. In *Proceedings of 14th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services, Melbourne, VIC, Australia, November 7-10, 2017 (MobiQuitous 2017),* 10 pages.
DOI: 10.1145/3144457.3144468

1 INTRODUCTION

User authentication is critical to numerous online services. Over the past few decades, passwords have been widely used for verifying the legitimacy of user trying to access any online service. However, vulnerabilities of this approach are well documented in literature [19] [26]. An average Internet user has over 118 online accounts [1]. It is very difficult for a user to craft a unique and strong password for each of these accounts. As such, users often tend to use naive passwords such as '12345'which are easy to guess [2]. Alternately, many users reuse a strong password across multiple accounts. The problem with this approach is that, if one of the service providers is hacked (which is not uncommon, for example, recently 500 million Yahoo accounts [25] were reported to be compromised), then all other accounts are susceptible. In order to mitigate these issues, many online services are increasingly offering two factor authentication (referred as 2FA in the rest of the paper) for protecting user accounts. 2FA mechanisms attempt to confirm the user's claimed identity by combining any two of following three factors : (i) Some thing you know (e.g., Password). (ii) Some thing you have (e.g., Phone for receiving one time code). (iii) Some thing you are (e.g., biometrics such as fingerprints). There are various implementations of 2FA. For example, [5] [6] are hardware token based 2FA mechanisms, where the user is given a dedicated piece of hardware (e.g. keyfob) that generates a one-time code (OTC) to serve as the second factor. However, these solutions require users to carry additional hardware, which not only incurs an additional cost for the service provider but is also easy to misplace. A cheaper alternative is to use software token based 2FA. Examples include, Duo Push [20], Encap Security [21] and Google's two-step verification, wherein the OTC is sent to the user's registered mobile device following a successful login attempt. Even though 2FA provides significantly improved security, user adoption rates are still very low. For example, Petsas et al. [3] found that only 6% of 100K Gmail accounts are 2FA enabled. The main reason behind the low adoption rate is that, these mechanisms require user to physically interact with either the hardware token or the device that receives the software token [22] [23].

Several recent works have attempted to develop 2FA mechanisms that reduce the user interaction by exploiting the fact that most of us carry multiple devices (e.g. laptops and smartphones) and that these devices are often in close proximity of each other. For example, [4] presented a 2FA mechanism capable of performing the user authentication by comparing the ambient sound recorded by mobile phone and computer from which login attempt is being made. Similarly, [7] and [8] reduce the interaction needed for 2FA by leveraging the direct communication (via Bluetooth) between user's mobile phone and computer. While these systems reduce the interaction required to perform 2FA, but they have not considered the robustness of their solution in the face of targeted attacks, situations where an adversary who has compromised a victim's password then attempts to hijack the 2FA mechanism by placing his device close to the victim's registered device.

In this paper we set out not only to improve the usability of 2FA by significantly reducing the effort exerted by the user, but also effectively address the issue of close adversaries. We propose *Wi-Auth*, a 2FA mechanism, that leverages the close proximity of user's secondary device (e.g. Phone) to the primary device (e.g. Laptop) from which log-in attempt is being made as a second factor for

MobiQuitous 2017, Melbourne, VIC, Australia

DOI: 10.1145/3144457.3144468

authentication. For establishing this proximity, Wi-Auth relies on WiFi signals which are ubiquitous in our environment. The main idea behind Wi-Auth is that if the two devices are in close proximity, then the WiFi signals received by both devices will be very similar. Wi-Auth triggers both devices to measure fine-grained Channel State Information (CSI) of the received wireless signals. If the CSI data is similar then it can be inferred that the two devices are close and thus serve as the second factor for authentication. Our preliminary experiments revealed that in the 2.4GHz WiFi spectrum, if two devices are within 0.5λ *(aprox. 2.5 inches)* of each other, then their CSI waveforms are similar in shape and amplitude. As such, Wi-Auth requires the user to place the secondary device adjacent to the primary device to ensure a strong match. This is the only active interaction required by Wi-Auth from the user. Moreover, CSI quickly de-correlates as the distance between the two devices increases. Thus, Wi-Auth can readily identify any potential attack from a close adversary, since realistically an adversary (assuming he has compromised the user credentials) can only place his device (primary) several inches (or even feet) away from the secondary pre-registered device without being noticed by the victim. It is important to highlight that the distance over which Wi-Auth can accurately authenticate a device is significantly smaller (2.5 inches) as compared to some of the other proximity based 2FA systems which use acoustics (several feet) or Bluetooth (several meters). These others schemes are thus susceptible to co-located attacks unlike Wi-Auth.

The main contributions of this paper are:

- We propose *Wi-Auth*, a novel *2FA* mechanism that utilizes *WiFi* signals for establishing the close proximity between user's secondary device and device from which log in attempt is being made. We use this proximity as a second factor of authentication while requiring very limited interaction form the user (the user simply has to place the secondary device adjacent to the primary device).
- We propose a lightweight two layer approach for comparing the CSI data measured by the two devices and show that this approach works effectively in different environments.
- Our CSI comparison algorithm also addresses the issue of targeted attacks in which attacker and victim are co-located and show that Wi-Auth can successfully detect these attacks. To the best of our knowledge, this issue has not been addressed before in context of 2FA.
- We implemented Wi-Auth on off-the-shelf WiFi 80211n devices and conducted extensive experiments in three different and practical settings (open office, apartment, large meeting space). Our evaluations revealed that Wi-Auth can achieve 94% authentication accuracy while maintaining very low (5-6%) false positives and negatives. Wi-Auth can also successfully detect a vast majority (95%) of co-located attacks launched by placing the adversary device at different distances from the primary device.

The rest of paper is organized as follows. Section 2 presents an overview of the Wi-Auth system. Section 3 encompasses preliminaries and feasibility of using CSI for achieving our goals. Section 4 expands on the CSI comparison algorithm employed by Wi-Auth. Our extensive experimental evaluations are discussed in Section 5. Related work is presented in Section 6, and finally the concluding remarks appear in Section 7.

2 WI-AUTH SYSTEM OVERVIEW

We consider the situation in which user attempts to authenticate himself to a web-server for accessing some online service. The device, from which log in attempt is being made, is referred to as the *primary device*. Without loss of generality we depict this device to be a laptop in Figure 1, but it could very well be any other personal device such as a tablet or smart phone. The primary device will have a Wi-Auth software module installed on it (e.g. as an App). Once the user's credentials (i.e. something the user knows, e.g., user name and password) are verified by the web server, Wi-Auth module on web server will be invoked for confirming 2FA. While we have assumed that the Wi-Auth software module needs to be installed on the web server, it may be possible to implement this as a stand-alone third party service and provide APIs for existing web servers to interface with this service in order to eliminate the overhead involved in modifying the existing web-servers. We assume that the user has registered a *secondary device* with Wi-Auth which is used to verify 2FA. A Wi-Auth software module (e.g. an app) is installed on this device and it is assumed that this app is associated to the user's account using any existing technique for enrolling software token. Without loss of generality, Figure 1 depicts a mobile phone as the secondary device, but one could envision other possibilities such as a smart watch. We also assume that at the end of the enrollment procedure, the server obtains the unique public key of the application installed on the secondary device and associates that key to the user account. Following the verification of the user credentials (Step 1 in Figure 1), the user is prompted to place the secondary device in close proximity (i.e. within 2.5 inches, as discussed in Section 3) of the primary device (Step 2 in Figure 1). The server sends the public key of user's secondary device to the primary device and instructs the Wi-Auth module on both devices to commence the CSI measurement process (Step 3 in Figure 1). Both devices send n ping packets to the connected AP and record the CSI data for the n ping responses received (An investigation into a suitable value for n is conducted in Section 5). Upon completion of CSI measurements, the primary device send its CSI samples encrypted with secondary device's public key to the server (Step 4 in Figure 1). Server in turn sends these samples to secondary device (Step 5 in Figure 1), which uses the methods outlined in Section 4 to compare the two CSI data sets (Step 6 in Figure 1) and informs the server about the outcome (Step 7 in Figure 1). We favor CSI comparison to be done on secondary device instead of web-server, as a typical web-server handles thousands of access requests at a particular time and this additional comparison step can affect the server response time. If Wi-Auth ascertains that the two CSI measurements are similar, then it can be interpreted that these two devices are in close proximity, which can only be the case if the secondary device was placed there by the user (thus verifying the second factor - something you have). We would like to highlight that apart from the initialization process which involves downloading and installing the Wi-Auth software module on the primary and secondary device, the only interaction required from the user is to place the secondary device in close proximity of the primary device. It is worth noting that in this paper we have not implemented the entire end-to-end solution but rather focused on the more challenging issue of developing a robust algorithm for effectively comparing the CSI measurements and demonstrate the efficacy of *CSI* as second factor of authentication.

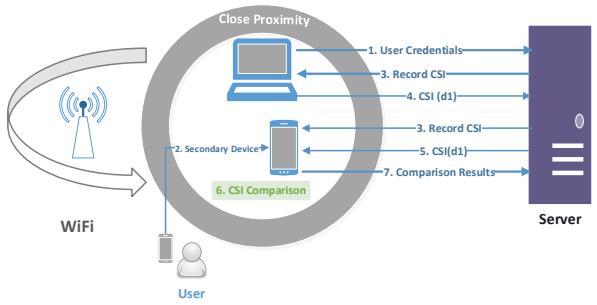


Figure 1: Wi-Auth System Overview

Threat Model: We consider the practical and realistic situation in which an adversary has obtained the victim's credentials (user name and password). Adversaries can easily gain access to this information either by phishing [24] or through various leaked password databases that are increasingly appearing on the Internet. We consider an attack to be successful, if the adversary successfully proves that the device from which he is attempting log-in and user's secondary device have similar CSI measurements, thereby deceiving Wi-Auth to grant access to the online service. We assume that adversary cannot gain access to victim's secondary device as in such a case every 2FA mechanism will fail. Similarly, we also do not consider Man-In-The-Middle attacks (MiTMA) in this piece of work. We consider two types of attackers: those that are located in close vicinity of the the legitimate user and those that are a bit farther from the user. The attack scenarios are illustrated in Figures 11, 12 and 13.

3 PRELIMINARIES AND FEASIBILITY STUDY

In this section, we first present an overview about CSI. Next, we present some results from our feasibility study to demonstrate that CSI has good potential to be used as an indicator of proximity between devices and thus serve as a means to achieve our goals.

3.1 Channel State Information

The current iterations of the WiFi standard such as 802.11n/ac support multiple antennas for MIMO communications and employ OFDM at the physical layer. In OFDM, a single channel is divided into multiple orthogonal sub carriers for improved performance. For example a 20MHz 802.11 channel is divided into 56 sub-carriers. Frequency response of each of these sub-carriers can be monitored by WiFi NICs, which is collectively referred to as the Channel State Information (CSI). CSI captures the effects of various impairments such as scattering, fading and multipath that are experienced by the radio signal as it propagates from the transmitter to receiver. The CSI information is used by the NIC to improve the quality of the WiFi link [11].

Lets assume that we have T_A transmitting and R_A receiving antennas, constituting a $T_A \times R_A$ MIMO System. If we represent the transmitted and received signals by X and Y respectively, then we can write;

$$Y = \mathbb{H}.X + \eta \quad (1)$$

where $\mathbb{H}$ represents CSI and η represents additive white Gaussain Noise. For every received packet, $\mathbb{H}$ will be a complex matrix of order $N_S \times T_A \times R_A$, where N_S represents number of sub-carriers. In this paper, we consider a single $T_A \times R_A$ antenna pair (See Section 4.1 for details). For i_{th} sub-carrier, CSI will be $\mathbb{H}_i = |\mathbb{H}_i| exp\{j \angle \mathbb{H}_i\}$, where $|\mathbb{H}_i|$ represents amplitude and $\angle \mathbb{H}_i$ represents phase response of the i_{th} sub-carrier [12]. In this paper, we rely on CSI amplitude for establishing the proximity between primary and secondary device.

There are many commercial devices that provide open access to CSI using customized drivers. Examples include Intel WiFi Link 5300 [18], Atheros 9390 [28] and Atheros 9580 [29]. In our study, we used Intel 5300 NIC, which reports CSI corresponding to $N_S = 30$ OFDM sub-carriers for every received packet.

3.2 Feasibility Study

For CSI to work as an efficient mean of second factor of authentication in lines with our goals, it must exhibit two essential properties. *(i)* The CSI measured by two devices in close proximity should be similar. *(ii)* The CSI recorded by the two devices should be different if they are further apart.

For visualizing the existence of these properties, we collected CSI from two devices equipped with Intel WiFi Link 5300 NICs. For analyzing first property, we placed two devices in defined close proximity i.e. < 2.5 inches, or in other words $\leq 0.5\lambda$ for *WiFi* operating at 2.4GHz. Once the two devices are separated by a distance greater than 0.5λ the two devices experience different multipath effects and consequently their CSI data is no longer similar. Figure 2 shows the mean amplitude of the CSI measured over 20 packets for each of the 30 sub-carriers for both devices. One can readily observe the high similarity in these two data sets. Next we repeated the experiment, except that we increased the distance between the two devices to 12 inches. One can observe from Figure 3 that the CSI data exhibits low correlation. We repeated our experiments at number of different locations and concluded that two devices in close proximity result in unique and correlated CSI measurements. Our numerous experiments also revealed that CSI quickly de-correlates with the distance (i.e. low correlation was observed if two devices are kept at a distance greater than 2.5 inches). These properties are essential for achieving the goals set out for Wi-Auth and provide the basis for considering the use of CSI as a mean to establish the second factor for authentication.

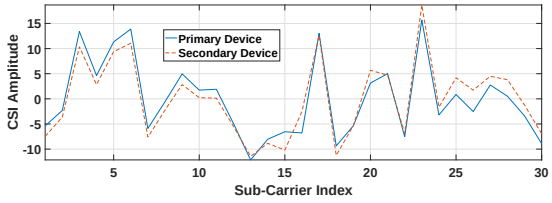


Figure 2: CSI measured by 2 devices in close proximity

4 CSI COMPARISON ALGORITHM

Figure 4 illustrates the work flow of proposed algorithm. We employ a two layer similarity analysis approach to enhance the robustness of overall authentication mechanism. Each step is discussed in the detail in following sub-sections.

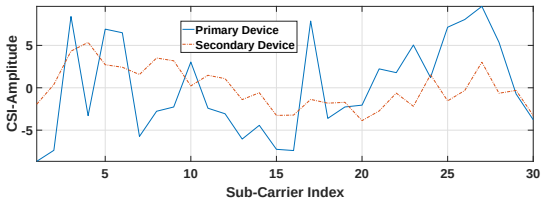


Figure 3: CSI measured by 2 devices separated by 12 Inches

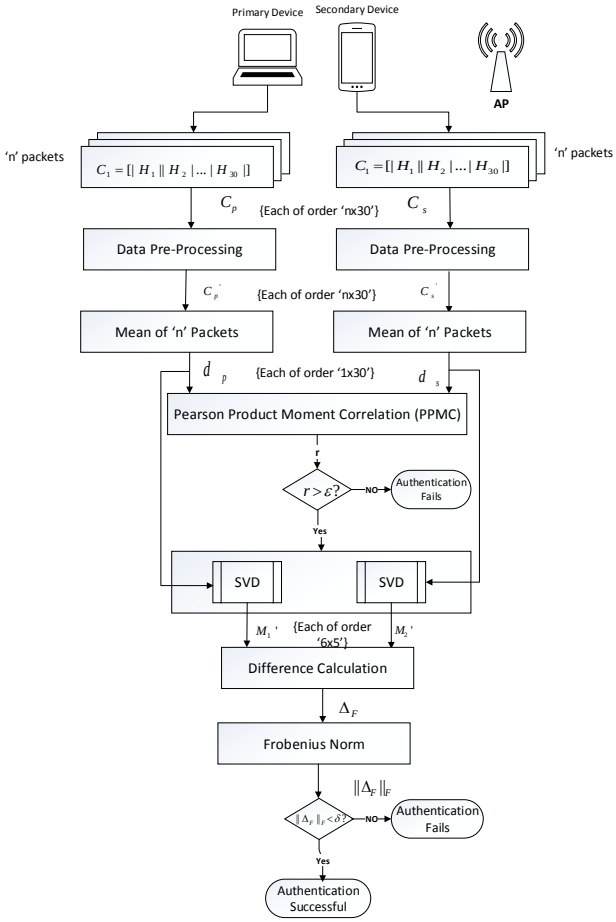


Figure 4: Wi-Auth Work Flow for Authentication

4.1 CSI Measurement

As noted in Section 2, the server Wi-Auth module instructs the primary and secondary device to record CSI for a sequence of n packets. These devices can be synchronized using existing Time Synchronization Function. It should be noted that precise time synchronization is not essential since in our experiments we observed that the mean of CSI data (for each sub-carrier, expressed as a vector) for a sequence of n packets from two closely located devices will match up even if their clocks are slightly misaligned. A similar observation was noted in [15]. Current commodity 802.11n devices are equipped with 3 antennas. Although we expect our system to work as desired with multiple antennas interfaced with each device, however, instead of a 3 x 3 MIMO system, we consider a 1 x 1 SISO system. This reduces the size of our data matrix from $30 \times 3 \times 3$ to $30 \times 1 \times 1$ for every received packet, which helps in reducing the computational complexity of Wi-Auth. There are two main reasons for considering a SISO system. First, having a single antenna pair enables us to position two devices very close to each other (i.e. within 2.5 inches of each other). Having multiple antennas would complicate the setup. Second, it allows us to verify that our approach would also work in a setting where a device only has a single antenna, which often is the case with wearable devices. Therefore, we obtain CSI data of order n×30 from each device, where 30 represents number of reported sub-carriers.The CSI data sets for the primary and secondary devices are represented by C_p and C_s respectively. Each row of C_P and C_S represents *CSI* amplitudes measured at each sub-carrier for one particular received packet.

4.2 Data Pre-Processing

In our feasibility study (outlined in Section 3), we observed that the CSI amplitudes for a sequence of back-to-back packets varies ever so slightly. Figure 5(a) shows the CSI amplitude of 30 sub carriers for a a sequence of 50 packets. Each line in the Figure 5(a) denotes a different packet. While the amplitudes vary slightly, the shape of the curves are similar. We attribute these slight differences to variable interference [13]. If we use the 'raw'CSI data (as depicted in Fig 5(a)) to determine the similarity between the two data sets, then we observed an increase in the False Negative Rate (FNR). To reduce this effect, we first subtract the mean CSI of each packet from the CSI of each individual sub-carrier for that packet. This is a standard normalization procedure which is widely used in literature (e.g. [13]). Mathematically it can be denoted as:

$$C_{P(i)}^{'} = C_{P(i)} - 1_{1\times30}\{\frac{1}{30}\sum_{k=1}^{30} C_{P(i)}(k)\} \tag{2}$$

$$C_{S(i)}^{'} = C_{S(i)} - 1_{1\times30}\{\frac{1}{30}\sum_{k=1}^{30} C_{S(i)}(k)\} \tag{3}$$

$C_{P(i)}$ and $C_{S(i)}$ represents i_{th} rows of C_P and C_S respectively and $1_{1\times30}$ is a vector of all ones. This process effectively adjusts the mean of every packet to zero and reduces the overall variance which is illustrated in Figure 5(b).

4.3 Similarity Analysis

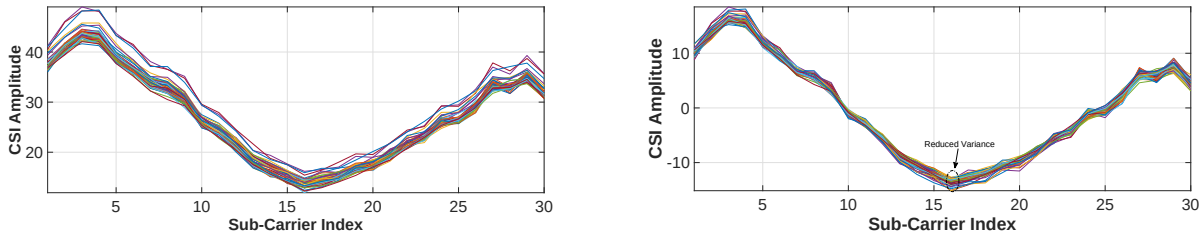
The output of data pre-processing module are two matrices $\acute{C_P}$ and $\acute{C_S}$ each of order $n \times 30$ with i_{th} row computed in accordance with equations 2 and 3 respectively. For similarity analysis, we first compute the mean of each column of $\acute{C_P}$ and $\acute{C_S}$ which represents the mean of CSI amplitudes corresponding to an individual sub-carrier for n packets and represent the final CSI data as vectors d_p and d_s each of order 1×30, corresponding to the primary and secondary device respectively. For simplicity, we represent d_p and d_s as;

$$d_p = [h_{p1}\ h_{p2}\ h_{p3}\ h_{p4}\ h_{p5} \cdots h_{p30}] \tag{4}$$

$$d_s = [h_{s1}\ h_{s2}\ h_{s3}\ h_{s4}\ h_{s5} \cdots h_{s30}] \tag{5}$$

Where h_{pi} and h_{si} represents the mean of i_{th} column of $\acute{C_P}$ or $\acute{C_S}$ respectively.

For checking the similarity between two CSI measurements (i.e. d_p and d_s), we first calculate the *Pearson's Product moment Correlation*



(a) Raw Data **(b) Data after Subtracting Mean**

Figure 5: Effect of Data Pre-processing on CSI (Colored Lines Indicate Individual Packets)

(PPMC) in accordance with equation;

$$r = \frac{\sum_{i=1}^{30}(d_{p(i)} - \overline{d_p})(d_{s(i)} - \overline{d_s})}{\sqrt{\sum_{i=1}^{30}(d_{p(i)} - \overline{d_p})}\sqrt{\sum_{i=1}^{30}(d_{s(i)} - \overline{d_s})}} \quad (6)$$

$\overline{d_s}$ and $\overline{d_p}$ represents the mean value of two data sets. The value of r varies from -1 to 1, with a value close to 1 indicating a strong positive association between two data sets. The intuition behind using PPMC is that, it provides a quick way to ascertain if the two data sets d_p and d_s exhibit a similar trend or not [30]. However, PPMC by itself is not sufficient to determine similarity. While PPMC can ascertain if the trend (i.e. shape) of the two data sets is similar, it does not consider the amplitude of the signals. To demonstrate this we illustrate the CSI data collected from two devices which are separated by $5ft$ in Figure 6. Observe that while the shape of the two data sets are similar, their amplitudes are vastly different. The PPMC as computed by Eq.6 for this example is 0.9, suggesting a strong positive association. Hence, we propose to use a second level similarity check which compares the amplitudes of the two data sets. This helps us in reducing the False Positive Rate (FPR) of overall authentication system. However, this second layer similarity check is only invoked if r is greater than certain threshold ε (explained in Sec. 5) , so that we can bypass the additional computations if there is no strong positive association observed between two CSI measurements. This helps in reducing the overall processing time when there is no strong positive association between two data sets.

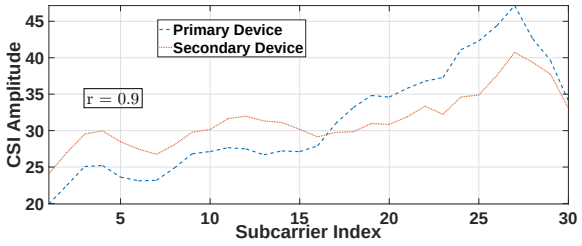


Figure 6: CSI of two devices not in Close-Proximity

Our second layer similarity check computes the magnitude of difference between two CSI measurements d_p and d_s. One would anticipate that amplitude of the CSI data for two devices in close proximity should be equal. However, our observations revealed that, there are some slight differences in the amplitudes as demonstrated in Figure 7, which shows the mean CSI data for 20 packets after pre-processing (as per Section 4.2) for devices in close proximity. These difference are due to noises and manufacturing variations in different devices. It is important that these variations are minimized

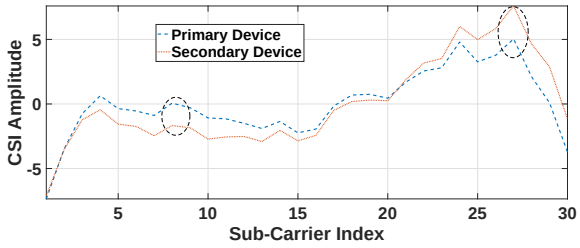


Figure 7: Difference in CSI amplitudes measured by two devices in close proximity

to reduce the chance of false negatives. We suggest that these differences can be reduced using Singular Value Decomposition *(SVD)*. SVD is a powerful technique that can be applied to any matrix, regardless of whether it is rectangular or square, unlike other decomposition techniques [14]. For any matrix M of order $m \times n$, SVD will produce,

$$M_{m\times n} = U \sum V^T \quad (7)$$

where as U and V are orthogonal matrices of order $m \times m$ and $n \times n$ respectively, while $\sum$ is a rectangular matrix (with only diagonal non-zero entries) with same order as that of M [14]. The diagonal entries (i.e. positive values) of $\sum$ are arranged in descending order and referred as *singular values*. Typically, the large singular values point to important and interesting information , while the remaining (which are generally very small as compared with top few singular values) can be assumed to be due to noise [16]. Hence, the singular values contained in a diagonal matrix $\sum$ are represented as $\sigma_1, \sigma_2, ..., \sigma_n$ (for $m \times n$ matrix with $n \leq m$). Out of these n singular values, top $\acute{n}$ values represent the actual useful information, while rest of $(n - \acute{n})$ values represent noise components. De-noised matrix $\acute{M}$ can be written as;

$$\acute{M} = U_1\sigma_1V_1^T + U_2\sigma_2V_2^T + \cdots + U_{\acute{n}}\sigma_{\acute{n}}V_{\acute{n}}^T \quad (8)$$

For applying *SVD* on our data vector d_p and d_s, first of all we construct a matrix M_i of order $m \times n$ from each data vector d_p and d_s, where $i = 1$ for primary device and $i = 2$ for secondary device. We refer this matrix as CSI Image (as SVD is extensively utilized to eliminate imperfections in images due to noise) with $m = 6$ *and* $n = 5$ (i.e. keeping $n < m$). CSI image formulation is shown below;

Data vector $d_p \curvearrowright$

h_{p1}	h_{p2}	h_{p3}	h_p	h_{p5}	h_{p6}	h_{p7}	$\cdots$	h_{p30}

From this data vector, six consecutive CSI values (e.g. CSI amplitude $h_1 \cdots h_6$) are grouped to form a column of CSI image matrix M_1 and so on as depicted below,

$$\underbrace{(h_{p1} \cdots h_{p6})}_{column1}, \overbrace{(h_{p7} \cdots h_{p12})}^{column2}, \overbrace{(h_{p13} \cdots h_{p18})}^{column3} \cdots \underbrace{(h_{p25} \cdots h_{p30})}_{column5}$$

The final *CSI image matrix* M_1 looks like;

$$M_1 = \begin{Bmatrix} h_{p1} & h_{p7} & . & . & h_{p25} \\ h_{p2} & h_{p8} & . & . & h_{p26} \\ . & . & . & . & . \\ . & . & . & . & . \\ h_{p6} & h_{p12} & . & . & h_{p30} \end{Bmatrix}$$

h_{pi} in this matrix represents mean CSI amplitude of i_{th} sub-carrier, while i_{th} column contains values corresponding to sub-carriers $\{((i \times 6) - 6) + 1\}$:$\{i \times 6\}$. Similarly we compute the CSI matrix M_2 using the data vector d_s corresponding to secondary device. Once the CSI images (matrices) are formulated, we apply *SVD* on each CSI matrix to find $M_{i(6\times5)} = U_{6\times6} \sum_{6\times5} V^T_{5\times5}$, where $i = 1$ for primary device and $i = 2$ for secondary device. We form a row vector S_v containing the 5 singular values corresponding to each device i.e. $S_v = [\sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5]$. Out of these five values, we consider the first two values (which are the largest), as our empirical study reveal that these singular values correspond to the most useful information in the CSI measurements, while other three singular values refer to noise components. We represent each de-noised matrix as $\acute{M}_{i(6\times5)} = u_1\sigma_1 v_1^T + u_2\sigma_2 v_2^T$. This method effectively reduces the noise power P_N by a factor of $\sum_{i=3}^{5}(\sigma_i^{\omega})^2$, where $P_N = \sum_{i=1}^{5}(\sigma_i^{\omega})^2$ represents the corresponding noise matrix for our *CSI* matrix, while σ_i^{ω} represents i_{th} singular value of noise matrix . The final result of applying SVD gives us two matrices $\acute{M}_1$ and $\acute{M}_2$ corresponding to primary and secondary device respectively.

After applying *SVD*, we calculate the difference between $\acute{M}_1$ and $\acute{M}_2$ and represent the difference matrix as Δ_F. Next, we calculate the magnitude of difference matrix using the simple form of matrix norm i.e. *Frobenius norm*. The intuition behind using Frobenius norm is that, it is simple to calculate and generally preffered for real time computations [27]. Frobenius norm (some times also referred as Euclidean or Hilbert-Schmidt or Schur norm) is defined as square root of sum of squared individual elements [17]. For any difference matrix Δ_F, Frobenius norm can be expressed mathematically as;

$$||\Delta_F||_F = \sqrt{\sum_{i=1}^{6}\sum_{j=1}^{5}|m_{ij}|^2} \tag{9}$$

where m_{ij} represents an individual entry contained in i_{th} row and j_{th} column of matrix Δ_F. As with PPMC, if the magnitude of difference is less than the threshold δ (i.e. $||\Delta_F||_F < \delta$) (See Section 5 for threshold value), we treat two measurements as similar and declare two devices in close proximity which results in successful authentication. On the other hand, if the magnitude of the difference is greater than the threshold (i.e. $||\Delta||_F > \delta$), then the two devices are deemed to not be in close proximity and thus authentication is denied .

5 EXPERIMENTAL EVALUATION

In this section, we present the evaluation setup, experimental methodology and performance of Wi-Auth. We also investigate the impact of various factors on the performance of Wi-Auth. As discussed in Section 2, we focus on evaluating the most challenging aspect of the system, which is the CSI comparison algorithm. We will implement the complete end-to-end system as part of our future work.

5.1 Evaluation Setup

All of our experiments are conducted using two HP 6930P laptops as the primary and secondary device [1]. Both the laptops are equipped with Intel 5300 NIC and run Ubuntu 12.04 with Kernel 3.13.0. These laptops are connected to a 802.11n WiFi AP operating in the 2.4GHz band. We used another HP 6930P laptop equipped with Intel 5300 NIC as an AP. This is because currently available CSI tool works reliably if both transmitter and receiver are equipped with Intel 5300 NIC [18]. We connected the WiFi NICs of the primary and secondary laptops to external antennas so that we could easily maintain a distance of less than 2.5 inches between the two devices. Additionally we only used a single antenna from the three available on the Intel 5300 NIC of both primary and secondary device (the motivation for using a SISO system is outlined in Section 4.1).

Experimental Methodology: We conduct extensive experiments in three different indoor environments. Our first test scenario is an open office area which consists of a number of cubicles. Layout of this scenario is shown in Figure 11. Second scenario is a typical apartment. Figure 12 presents layout of this scenario. While the third scenario is a large meeting area used for student interaction containing a number of chairs and desks. Figure 13 depicts the layout of this scenario. We selected these scenarios as they represent practical settings in which a 2FA system such as Wi-Auth could be used. In each scenario we conducted two types of experiments. The first set of experiments test whether Wi-Auth performs as expected in a typical usage scenario. In these experiments the primary and secondary devices are placed in close proximity of each other (i.e. $<$ 2.5 inches apart) and at a distance of 1m from the AP. We have evaluated the effect of varying the distance from the AP in Section 5.3. The AP broadcasts 20 packets and both devices record the CSI corresponding to these packets. We expect that WiAuth should achieve a positive match between the two CSI data sets for each iteration of this experiment. The second set of experiments evaluates the effectiveness of WiAuth in preventing attacks. Here we assume the primary device to be the victim. We assume that an adversary has obtained the credentials for the victim in some manner (as discussed in Section 2). The adversary is now aiming to falsely complete the second factor authentication employed by Wi-Auth. The adversary won't be able to place his device (referred to as the attacker) in close proximity (i.e. within 2.5 inches) of the victim without being noticed. However, the attacker could be placed out of sight of the victim but still in the vicinity of the victim (e.g, in a neighboring cubicle in our first scenario or outside a room in our second scenario or an adjacent table in our third scenario). We consider two types of attack scenarios. In the first instance, the attacker is placed at a distance of approximately 5-7 feet from the victim (referred to as 'relatively close'in the results). In the

[1] We used laptops in our implementation as the CSI tool [18] that we used for recording the CSI only works with the Intel WiFi link 5300 NIC. However, we anticipate that our solution could readily work with other devices provided we have access to the CSI data.

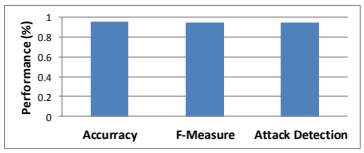


Figure 8: Wi-Auth Performance in Scenario 1

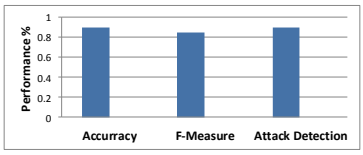


Figure 9: Wi-Auth Performance in Scenario 2

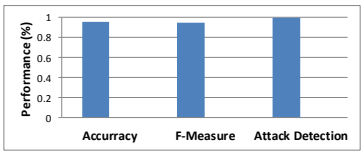


Figure 10: Wi-Auth Performance in Scenario 3

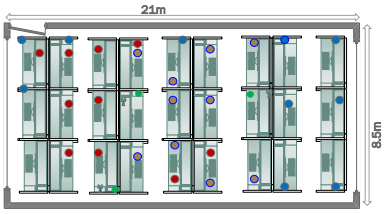


Figure 11: Layout of Scenario 1

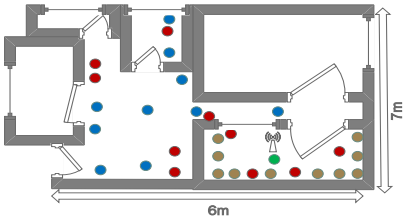


Figure 12: Layout of Scenario 2

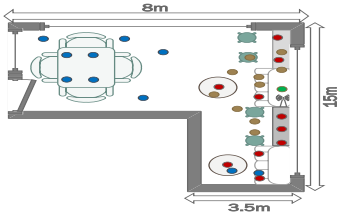


Figure 13: Layout of Scenario 3

second instance the attacker and victim are separated by over 25 feet (referred to as 'far'in the results). This allows us to study the effect of both close and remote adversaries (as discussed in the threat model in Section 2). For the attack experiment, the AP is located at a distance of 1m from the victim and a sequence of 20 packets is used to measure the CSI. Figure 14 depicts the setting for the various experiments.

In each scenario, we conducted a total of 30 experiments. 10 experiments were conducted to test the accuracy of Wi-Auth with the primary and secondary device in close proximity. The red dots in Figures 11, 12 and 13 depict the locations at which each experiment was conducted (a single dot represents both the primary and secondary device as they are co-located). The next 10 experiments consider the relatively close attack scenario with the victim and attacker separated by approx 5-7 feet. Brown dots in Figures 11, 12 and 13 represent locations of these experiments (the brown dot represent location of an attacker, while victim is represented by green dot). Finally, the last 10 experiments considers the far attack scenario where the victim and attacker are separated by 25 feet or greater. Blue dots in Figures 11, 12 and 13 represent locations of these experiments (blue dot represent location of an attacker, while victim is represented by green dot).

Performance Evaluation: We use the following metrics to evaluate the performance of Wi-Auth:

Accuracy: Accuracy of *Wi-Auth* is defined as;

$$A = \frac{n^{tp} + n^{tn}}{\check{n}} \tag{10}$$

where n^{tp} and n^{tn} represents the number of true positives and negatives respectively, while $\check{n}$ represents total number of tests. A true positive refers to the instance where Wi-Auth declares that two CSI measurements are similar when the primary and secondary device are placed in close proximity (i.e. < 2.5 inches apart). A true negative is when Wi-Auth declares two CSI measurements to be

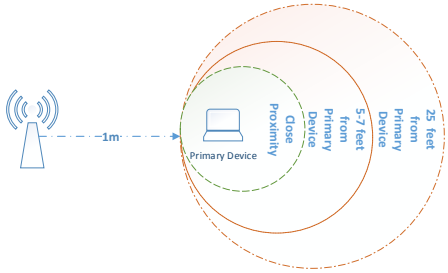


Figure 14: Experimental Methodology Overview

dissimilar when the primary and secondary device are not in close proximity (i.e. > 2.5 inches apart). A ranges from 0 to 1, with a value of 1 implying 100% accuracy.

F-measure: For combining the precision and recall rate of *Wi-Auth*, we present F-measure, which is defined as;

$$F = \{2 * \frac{Precision * Recall}{Precision + Recall}\} \tag{11}$$

where $precision = \frac{n^{tp}}{n^{tp}+n^{fp}}$, while $recall = \frac{n^{tp}}{n^{tp}+n^{fn}}$. n^{fp}, n^{fn} represents number of false positives and negatives respectively. F ranges between 0 and 1, with a value of 1 implying that no false positive or negative decisions were made during the experiments. On the other hand, a value of 0 suggests that Wi-Auth cannot make any true positive decisions.

Attack Detection Ratio: With this we evaluate if Wi-Auth can successfully detect that the secondary device trying to verify the second factor is not in close proximity of the primary device (i.e. victim) and thus an attacker. The ratio measures the fraction of attacks that are successfully detected.

5.2 Evaluation Results

Threshold Values: For setting the threshold values of PPMC and Frobenius Norm, we conducted a total of 30 experiments (10 in each scenario) by placing primary and secondary device in close proximity. Threshold value of *PPMC* (ε) was set using the expression $\mu_p - n\sigma_p$, where μ_p is mean value of PPMC in all experiments, σ_p represents standard deviation of PPMC, while n represents the a constant multiplier. Similarly, the threshold value of norm (δ) is set in accordance with expression $\mu_n + m\sigma_n$, where μ_n is mean value of norm, σ_n is standard deviation of norm, while m represents constant multiplier. Based upon our experiments, we set $n = 1$ for PPMC and $m = 1.5$ for Norm. These parameters compute $\varepsilon \approx 0.75$ and $\delta \approx 10.5$. For successful authentication, *PPMC* should be greater than ε while *norm* should be less than δ. We also analyzed the trade off between FP and FN by varying the different threshold values for ε and δ. Figures 15 & 16 show that the selected threshold values return the lowest FP and FN rates.

Scenario 1: In first scenario, Wi-Auth achieves 96% accuracy, F-measure of 95% and attack detection ratio of 95% as depicted in Figure 8. Calculated similarity indexes are shown in Table. 1. It is evident that the values for PPMC and Norm for almost all experiments meet the threshold criteria (i.e. for successful authentication, $r > 0.75$ and $||\Delta_F||_F < 10.5$). The results highlighted in blue show the validity of our two-step approach. Consider the experiment for the relatively close attack referenced by index 4. Here the PPMC is 0.9 (which is greater than the threshold of 0.75) and thus the algorithm would progress to the second level check. However, the Norm (24.5) is much greater than the corresponding threshold (10.5) which correctly prevents a successful authentication. The result highlighted in red depicts the only experiment with a false positive decision. In summary, for 29 out of 30 experiments, Wi-Auth worked as expected; it correctly identified the device when in close proximity with 100% accuracy, and detected 19 out of 20 attacks.

Index	Close Proximity		Relatively Close (5-7 ft)		Far (25 ft)	
	PPMC	Norm	PPMC	Norm	PPMC	Norm
1	0.91	6.44	0.18	26.9	0.34	35.61
2	0.95	4.3	-0.59	20.3	0.44	21.81
3	0.78	7.5	-0.42	16.18	-0.63	32.05
4	0.79	8.1	0.9	24.5	0.43	44.48
5	0.91	9.6	-0.52	8.98	0.72	27.4
6	0.92	8.2	0.27	12.44	0.66	43.3
7	0.83	8.1	0.27	13.5	0.69	71.5
8	0.94	8.35	-0.44	77.07	0.83	38.6
9	0.84	7.4	0.83	10.3	-0.54	72
10	0.77	6.9	0.38	19	0.63	43.9

Table 1: Evaluation Results in Scenario 1

Scenario 2: As depicted in Figure 9, Wi-Auth achieves 90% accuracy, a F-measure of 85% and attack detection ratio of 90% in scenario 2. Detailed values of similarity indexes are shown in Table. 2. These results depict that, in 90% of all experiments conducted in this scenario, Wi-Auth established the correct proximity between two devices. Highlighted results in red color show the false negative and positive decisions of Wi-Auth in this scenario, while those highlighted results in blue color represent the effectiveness of two layer similarity approach as described in section 4.

Scenario 3: Figure 10 illustrates the performance of Wi-Auth in scenario 3, depicting 96% accuracy, 95% F-measure and 100% attack detection. Detailed values of results of these experiments are shown in Table 3. These results show that Wi-Auth made only one false negative decision in this test scenario, which is highlighted in red color in Table 3. While in all other experiments, Wi-Auth correctly established the true proximity between primary and secondary device. Results highlighted in blue color in Table 3 show the validity of our two layer similarity approach in this scenario.

Index	Close		Same Room (5-7 ft)		Outside room	
	PPMC	Norm	PPMC	Norm	PPMC	Norm
1	0.94	2.98	0.58	10.98	-0.53	29.6
2	0.75	10.1	-0.15	35.15	0.48	69.2
3	0.97	8.1	-0.6	61	0.87	27.15
4	0.9	9.6	0.86	5.63	-0.45	44.96
5	0.78	8.6	-0.54	11.25	-0.0009	23.13
6	0.7	11.4	0.06	13.44	0.8	13.6
7	0.74	7.8	0.43	12.89	0.62	8.71
8	0.99	1.75	0.94	4.98	0.06	23.2
9	0.86	10.4	0.35	10.15	-0.65	15.9
10	0.97	7.11	0.83	16.2	0.45	8.82

Table 2: Evaluation Results in Scenario 2

Index	Close Proximity		Relatively Close (5-7 ft)		Far (25 ft)	
	PPMC	Norm	PPMC	Norm	PPMC	Norm
1	0.75	7.55	0.61	13.35	-0.72	42.12
2	0.97	2.39	0.35	15.5	-0.77	48.4
3	0.71	7.86	0.0663	10.35	-0.49	37.6
4	0.76	6.41	0.72	13.1	0.62	46.6
5	0.86	5.5	0.65	17.2	-0.52	26.3
6	0.77	6.75	0.41	9.52	0.8	30.6
7	0.82	9.4	-0.52	28.6	-0.44	31.8
8	0.97	3.22	-0.16	36.4	-0.19	24.9
9	0.97	4.1	-0.27	13.3	-0.48	39.9
10	0.93	7.62	-0.39	24.9	0.92	18.6

Table 3: Evaluation Results in Scenario 3

Overall Performance of Wi-Auth: To summarize, we tested Wi-Auth at 90 different locations in three different environments and obtained the average accuracy of 94%, F-measure of 91.6% and attack detection of 95%. Overall False Positive Rate (FPR) of Wi-Auth is 5%, while False Negative Rate (FNR) is 6.67%. Higher accuracy and low values of *FPR and FNR* indicate that Wi-Auth is very effective in a variety of environments.

5.3 Analysis and Discussion

5.3.1 Impact of Distance between AP and primary and secondary devices. An important factor that can affect the performance of Wi-Auth is the distance between AP and the primary and secondary devices that measure the CSI, as an increase in distance can introduce different mutipaths which may affect CSI. We analyzed the effect of this distance by conducting a set of experiments in Scenario 2. We placed the primary and secondary devices in close proximity and varied the distance between the AP and these two devices from 0.7m to 5m (in multiples of 0.7m). This environment also had a wall at a distance of about 2.8m from the two devices. As such, the final 3 experiments (where the distance from the AP was 3.5, 4.2 and 4.9 m respectively) were conducted with a wall between the AP and two devices. Figure 17 show the variations of PPMC and Norm for different distances. These results show that for different distance, although PPMC stays relatively constant, the norm fluctuates significantly in same environment. However, an important point to note here is that, both of these values stay within the defined thresholds for all experiments, even those containing an intermediate wall. This shows that Wi-Auth can achieve good accuracy irrespective of the distance of the devices from the AP although there are variations in norm with distance.

5.3.2 Impact of varying the mounting height of the AP. The position where an AP is installed can vary in many practical settings. As such, we investigate the effect of varying the mounting height of the AP on the performance of Wi-Auth. We should point out that in all our previous experiments the AP and primary and secondary devices were placed at the same height. We test the impact of

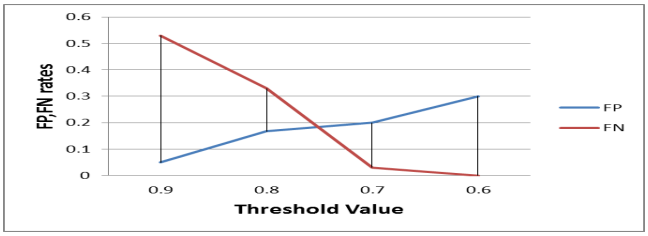


Figure 15: FP vs FN tradeoff at different PPMC thresholds

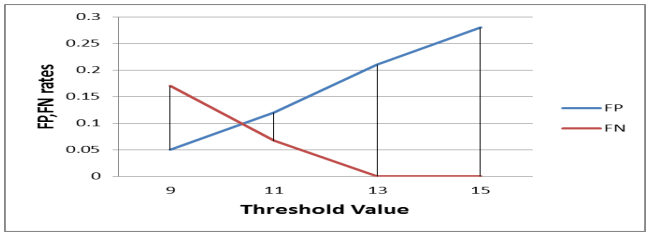


Figure 16: FP vs FN tradeoff at different F-Norm thresholds

different mounting heights by conducting experiments in Scenario 2, where the AP is placed at a distance of 1m from the primary and secondary devices which are in close proximity of each other. We vary the relative height between the devices and the AP from 0ft to 5ft (in increments of 1 ft). Results shown in Figure 18 reveal that, while Norm values fluctuates, both PPMC and Norm stay within the defined threshold for different heights. This portrays the efficacy of Wi-Auth for different practical scenarios where AP may be mounted in different places (e.g. on the ceiling in an office, on top of a table in a home, etc.).

5.3.3 Impact of Number of Packets. For analyzing the impact of number of packets used for the CSI measurements, we conducted experiments in Scenario 2 where the two devices were placed in close proximity with the AP placed at a distance of 1m. We varied the number of packets from 20 to 200 (in increments of 20). Results depicted in Figure 19 show that, there is no definite trend in Norm values with number of injected packets, while PPMC stays relatively same. However, for different packets injected, both PPMC and norm stays within the defined threshold. Therefore, we choose, 20 packets for most of our experiments to reduce both the time for conducting CSI measurements and the overall processing time.

5.3.4 Practical Considerations: Recall from Section 2, that in Wi-Auth, the CSI data recorded by the primary device has to be first transmitted to the server which then forwards it to the secondary device. We observed that the CSI measurements collected for 20 packets only take up 3KB. The overheads of transmitting this are thus very low. One of the reasons for the small size of the CSI data is that we use a SISO system which reduces the order of the CSI data from 30 x 3 x 3 to 30 x 1 x 1 for each packet. Additionally, Wi-Auth employs a fairly efficient two step algorithm (as described in Section 4) for comparing the two data sets. Therefore, we anticipate that the run time complexity of Wi-Auth to be relatively low, even on mobile devices.

6 RELATED WORK

As described in Section 1, traditional 2FA mechanisms utilize either hardware or software tokens. Hardware tokens like [5] [6] requires user to carry a dedicated hardware and also they incur additional manufacturing cost. As a result, they are not very widely used. Instead, a software based approach is preferred where a one time token is dispatched to a registered mobile device of the user if the first step of the authentication is successful. One popular example of a software token based 2FA is Google's two-step verification. However, adoption rate for these systems is quite low due to the non-trivial interaction required by the user (i.e. entering the one time code every time a user authenticates). Number of recent works have tried to reduce the interaction associated with the typical software tokens based 2FA mechanisms. For example, [4] presented a 2FA mechanism referred as sound-proof. Sound-proof relies on ambient sound recorded by the microphones of primary and secondary device for establishing the proximity between two devices. Devices are considered to be in close vicinity, if sound recorded by two devices show high similarity index. This solution effectively reduce the user interaction required in traditional 2FA mechanisms. However, there are two main problems associated with this solution. First one is that, this solution requires user to generate some sound (e.g. clearing throat) in situations under which no ambient sound is present. Note that, it is not uncommon that the ambient sound in a typical office setting is very low. Another problem associated with this approach is that, sound-proof does not consider attacks under which adversary and victim are co-located. Similarly, few works like [7],[8] and [9] have used Bluetooth for direct communication between user phone and device from which log-in attempt is made. This solution can also be considered effective for reducing the interaction required for proving the possession of second factor of authentication. However, co-located adversaries can not be thwarted out (as discussed in Section 1) in this mechanism as well. SlickLogin [10] is another solution that can effectively reduce the user interaction. This mechanism requires user to place his smart phone just few inches away from the device from which user is attempting log-in. In this solution, a uniquely generated sound (in-audible to human ear) is played by the speakers and an App on the user's smart phone records it by utilizing its microphone. Once the sound signature is verified by the phone, it requests server to allow access to the user attempting to log-in. While this solution is promising, it may fail if noise is present in the surrounding. Also, sounds generated in these frequencies may be disruptive to pets and younger individuals who can hear sound in these frequencies.

7 CONCLUSION

In this paper we presented a novel 2FA system called Wi-Auth wherein the user has to simply place a previously registered secondary device in close proximity of the primary device from which the user is logging in to an online service. Wi-Auth detects the close proximity of the two devices by comparing the CSI of the WiFi signals recorded at both devices to complete the second factor of authentication. We designed a lightweight and robust two step algorithm for comparing the CSI measurements from the two devices. Our extensive evaluations in three practical settings demonstrated that our system can achieve an average accuracy of 94%. Moreover, we tested the robustness of our system to co-located attacks

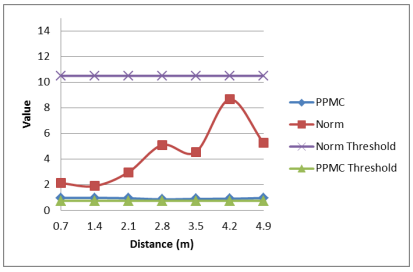


Figure 17: Impact of Distance between AP and CSI Measuring Devices

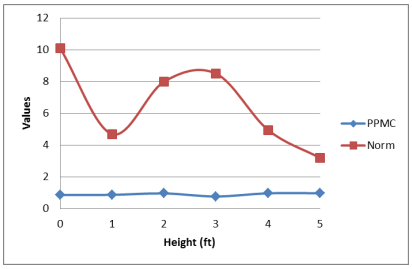


Figure 18: Impact of Height between AP and CSI Measuring Devices

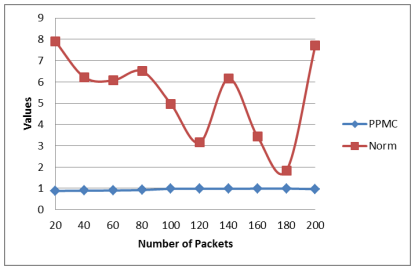


Figure 19: Impact of Number of Packets

and found that Wi-Auth can prevent 95% of attacks. This paper is primarily focused on the most challenging aspect of the system, i.e., the CSI matching algorithm. In our future work, we plan to implement and evaluate the complete end-to-end system and also undertake an extensive usability study.

REFERENCES

[1] Darren Allan. 2015 . We all have too many online accounts and can't remember the passwords (June 2015). Retrieved April 20, 2017 from http://www.itproportal.com/2015/07/23/we-all-have-too-many-online-accounts-and-cant-remember-the-passwords/

[2] Ashlee Vance. 2010. If your password is 123456, Just Make It Hack Me. (Jan. 2010) Retrieved April 22, 2017 from http://www.nytimes.com/2010/01/21/technology/21password.html

[3] Thanasis Petsas, Giorgos Tsirantonakis, Elias Athanasopoulos, and Sotiris Ioannidis. 2015. Two-factor authentication: is the world ready?: quantifying 2FA adoption. *In Proceedings of the Eighth European Workshop on System Security (EuroSec '15)*. ACM Press, New York, NY, USA, 4-7 pages. DOI: https://doi.org/10.1145/2751323.2751327

[4] Nikolaos Karapanos , Claudio Marforio , Claudio Soriente and Srdjan Capkun. 2015. Sound-Proof: Usable Two-Factor Authentication Based on Ambient Sound. *In Proceedings of 24th USENIX Security Symposium (USENIX Security'15)*. USENIX Association, Berkeley, USA, 483-498. DOI: https://www.usenix.org/conference/usenixsecurity15/technical-sessions/presentation/karapanos .

[5] RSA. RSA SecureID. Retrieved 26 June, 2017 from https://www.rsa.com/en-us/products/rsa-securid-suite/securid-hardware-tokens

[6] yubico. Yubikeys . Retrieved 26 June, 2017 from https://www.yubico.com/products/yubikey-hardware/

[7] Alexei Czeskis, Michael Dietz, Tadayoshi Kohno, Dan Wallach, and Dirk Balfanz. 2012. Strengthening user authentication through opportunistic cryptographic identity assertions. *In Proceedings of the 2012 ACM conference on Computer and communications security (CCS'12)*. ACM, New York, NY, USA, 404-414. DOI: http://dx.doi.org/10.1145/2382196.2382240

[8] Bryan Parno, Cynthia Kuo, and Adrian Perrig. 2006. Phoolproof phishing prevention. *In Proceedings of the 10th international conference on Financial Cryptography and Data Security (FC'06), Giovanni Crescenzo and Avi Rubin (Eds.)* Springer-Verlag, Berlin, Heidelberg, 1-19. DOI: http://dx.doi.org/10.1007/11889663

[9] Maliheh Shirvanian, Stanislaw Jarecki, Nitesh Saxena and Naveen Nathan. 2014. Two-factor authentication resilient to server compromise using mix-bandwidth devices, *In proceedings of Network and Distributed System Security Symposium (NDSS'14)*. Internet Society, San Diego, CA, USA,1-16. DOI: http://dx.doi.org/10.14722/ndss.2014.23167.

[10] Greg Kumparak. 2014.Google Acquires SlickLogin, The Sound-Based Password Alternative. Retrieved 10 May, 2017 from https://techcrunch.com/2014/02/16/google-acquires-slicklogin-the-sound-based-password-alternative/

[11] IEEE Std. 2009 . 802.11n-2009: Enhancements for higher throughput, 2009., Retrieved 01 Dec. 2016 from https://www.ieee802.org.

[12] Zheng Yang, Zimu Zhou, and Yunhao Liu. 2013. From RSSI to CSI: Indoor localization via channel response. *ACM Comput. Surv.*.46, 2(25 (Dec, 2013), 32 pages. DOI: http://dx.doi.org/10.1145/2543581.2543592

[13] Hongbo Liu, Yan Wang, Jian Liu, Jie Yang, and Yingying Chen. 2014. Practical user authentication leveraging channel state information (CSI). *In Proceedings of the 9th ACM symposium on Information, computer and communications security (ASIA CCS '14)*. ACM, New York, NY, USA, 389-400. DOI: http://dx.doi.org/10.1145/2590296.2590321

[14] Dan Kalman. 1996 . A Singularly Valuable Decomposition: The SVD of a Matrix, *The College Mathematics Journal*. MAA 27,1 (1996), 2-23. DOI: http://www.math.umn.edu/ lerman/math5467/svd.pdf

[15] Wei Xi, Chen Qian, Jinsong Han, Kun Zhao, Sheng Zhong, Xiang-Yang Li, and Jizhong Zhao. 2016. Instant and Robust Authentication and Key Agreement among Mobile Devices. *In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security (CCS '16)*. ACM, New York, NY, USA, 616-627. DOI: https://doi.org/10.1145/2976749.2978298

[16] David Austin. We Recommend a Singular Value Decomposition. American Mathematical Society (AMS).Retrieved June 20, 2017 from http://www.ams.org/samplings/feature-column/fcarc-svd

[17] Gene H. Golub and Charles F. Van Loan. 1996. Matrix Computations (3rd. ed.), The John Hopkin Press Ltd. London.

[18] Daniel Halperin, Wenjun Hu, Anmol Sheth, and David Wetherall. 2010. Predictable 802.11 packet delivery from wireless channel measurements. *In Proceedings of the ACM SIGCOMM 2010 conference (SIGCOMM '10)*. ACM, New York, NY, USA, 159-170. DOI: http://dx.doi.org/10.1145/1851182.1851203

[19] Drew Thomas. 2016. The Current State of Authentication:We have a Password Problem (June 2016) . Retrieved 10 May, 2017 from https://www.smashingmagazine.com/2016/06/the-current-state-of-authentication-we-have-a-password-problem/

[20] Duo Security, INC. Duo Push. Retrieved 10 Jan,2017 from https://duo.com/product/trusted-users/two-factor-authentication/authentication-methods/duo-push

[21] ENCAP SECURITY. Encap Security. Retrieved 15 Jan, 2017 from https://www.encapsecurity.com/

[22] Nancie Gunson, Diarmid Marshall, Hazel Morton and Mervyn Jack. 2011. User perceptions of security and usability of single-factor and two-factor authentication in automated telephone banking. *Computers & Security*. ELSEVIER 30,4 (2011) 208-220. DOI: https://doi.org/10.1016/j.cose.2010.12.001

[23] Catherine S. Weir, Gary Douglas, Martin Carruthers1 and Mervyn Jack. 2010. User perceptions of security, convenience and usability for ebanking authentication tokens., *Computers & Security*. ELSEVIER 28 (2009), 47-62. DOI: https://doi.org/10.1016/j.intcom.2009.10.001

[24] Email, recorded messages and SMS Scams. Retrieved December 18, 2016 from https://www.commbank.com.au/news/netbank-news-30.html

[25] Mike Snider and Elizabeth Weise. 2016. 500 million Yahoo Accounts breached (Sep. 2016). Retrieved March 13, 2017 from http://www.usatoday.com/story/tech/2016/09/22/report-yahoo-may-confirm-massive-data-breach/90824934/

[26] Sourav Kumar Dandapat, Swadhin Pradhan, Bivas Mitra, Romit Roy Choudhury, and Niloy Ganguly. 2015. ActivPass: Your Daily Activity is Your Password., *In Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*..ACM, New York, NY, USA, 2325-2334. DOI: https://doi.org/10.1145/2702123.2702457

[27] Jadran Lenarcic, and Vincenzo.P. Castelli. 1996. Recent Advances in Robot Kinematics, *Springer* .

[28] Souvik Sen, Jeongkeun Lee, Kyu-Han Kim, and Paul Congdon. 2013. Avoiding multipath to revive inbuilding WiFi localization. *In Proceeding of the 11th annual international conference on Mobile systems, applications, and services (MobiSys'13)*. ACM, New York, NY, USA, 249-262. DOI=http://dx.doi.org/10.1145/2462456.2464463

[29] Yaxiong Xie, Zhenjiang Li, and Mo Li. 2015. Precise Power Delay Profiling with Commodity WiFi.*In Proceedings of the 21st Annual International Conference on Mobile Computing and Networking (MobiCom'15)*. ACM, New York, NY, USA, 53-64. DOI: http://dx.doi.org/10.1145/2789168.2790124

[30] Derrick TR, Bates BT, Dufek JS. 1994.Evaluation of time-series data sets using the Pearson product-moment correlation coefficient. *Med Sci Sports Exerc*. NCBI 26,7 (Jul 1994), 919-28. DOI: https://www.ncbi.nlm.nih.gov/pubmed/7934769