USER-SIDE DETECTION OF EVIL TWIN ATTACKS

PROJECT FOR CMSC818O

Yusuf Alnawakhtha
Department of Computer Science
University of Maryland
College Park, MD 20740

Mackenzie Kong-Sivert
Department of Computer Science
University of Maryland
College Park, MD 20740

Tamer Mograbi
Department of Computer Science
University of Maryland
College Park, MD 20740

October 4, 2023

ABSTRACT

For over a decade now, users connecting to WiFi access points have been vulnerable to evil-twin attacks. There is an abundance of papers on this topic, but most of their assumptions about user behavior rest on speculation rather than observation or testing. With this study, we observe and quantify users' susceptibility to evil-twin attacks, as well as their efficiency in using some user-side defenses that have been proposed in the past ([1],[2]). In addition, we propose a QR code based defense that allows the user to perform a public key cryptography protocol with the honest access point, and we compare this defense to the other user-side defenses. our results show that users feel more secure by using the QR code defense and that they rated it as easier to use than the other defenses and that most users are able to connect to the access point quicker than they are able to connect to a password-secured access point.

1 Introduction

This study seeks to address the a relatively old type of attack called the "evil twin attack". In such and attack, an attacker will go to a public location with a legitimate wireless access point and create its own malicious access point (AP) with the same name. The attacker's intent in this case is to trick the user into connecting to its AP rather than the honest AP, so the attacker can monitor the user's traffic or redirect it. This allows the attacker to steal sensitive information such as passwords or direct the user to websites containing malware. This attack can present itself as a Man-in-the-middle attack, where the attacker's AP redirects traffic to the honest AP, or not. In the latter case, the attacker's AP forwards the user's traffic separately, as the honest AP would [3]. This allows the attacker to evade detection more easily but also requires more capability from the malicious AP.

An evil twin attack is considered difficult to defend against because the evil twin can mimic an honest AP perfectly in regards to the public information it displays, so a user who has not connected to the honest AP before has no shared information with the honest AP that the attacker does not have access to. These attacks also do not require a hefty infrastructure to deploy and can be used in public places such as restaurants, museums, parks, or airports where there is a large number of potential victims.

Despite its long history and myriad defenses, evil-twin attacks are still very prevalent today, and users are still susceptible to them. This is partly because very few of the proposed defenses have been put into effect and partly because of the lack of research on user behavior when confronted with such an attack. Though many of the proposed solutions are not perfectly secure, they would still offer more protection to the user than is currently put into place. Even certain simple user behavior changes such as not inherently trusting APs named after a location (e.g., an airport) would be helpful if they were advocated for. Hence, it seems important to make a note of users' current vulnerability to attacks when devising more. Thus, the purpose of this study is both to determine empirically how users behave when confronted with such an attack and to test a defense that we propose in Section 3. The defense we propose utilizes QR codes to streamline the detection process We believe that such an approach is not only easier than some of detection processes proposed by other, but it also gives the user an additional sense of security since they take a part in the authentication process by scanning a QR code.

2 Related Work

Most of the proposed defenses for evil-twin attacks can be separated into defenses that depend on detection on the user's and those that depend on detection on the administrator's side. Some of the first defenses to be proposed in previous research have been on admin-side detection. This was mainly because it is significantly easier to implement, but the main disadvantage is that the administrator has less incentive to ensure secure connections than the user has to connect to a secure AP. That is, while the administrator may care about their reputation as a secure institution, they have much less at stake than the user, who is the one being attacked. Hence, some of these early schemes were secure, but they failed to be implemented because of the amount of effort involved on the admin's side.

For example, one of these early defenses was to have administrators systematically verify all the legitimate access points using network enumeration tools. Such scans were time-consuming on the part of the administrator, so they were not done often enough to be effective [4]. Failing to achieve widespread use with this technique, researchers moved on to propose methods by which an outside observer can determine if an access point is legitimate or not. These most often took the form of user-side defenses.

One of the earliest user-side defenses, outlined in [5], was called called "context-leashing", in which a device catalogues the SSIDs and associated distance/location of associated APs in order to compare them against previous AP locations. If there is an AP in an unidentified location, it is to be treated with more suspicion than the AP in the same location. This method, however, requires that the user visit the location at least once before it is useful.

Another proposed method is WiFiHop [4], which was specifically designed to detect MITM attacks. The strategy of WiFiHop is to create a watermark and place it on a packet, which the user will then send to a given AP. If the watermarked packet shows up in both APs, then the user was connected to an adversarial AP. A different team, CETAD [3], distinguishes between honest APs and attackers based on the timing of packets coming from the AP. A more recent paper [6] proposes to accomplish this goal by determining the number of association responses in the handshake between client and AP.

The user has more incentive to put effort into a defense, but, as will be explained later in this paper, they are still unwilling to expend a substantial amount of effort to protect themselves. Further, they may not have the same computing resources or technical expertise as the administrator and thus may not be able to implement all of the same countermeasures. In our review of the previous work, assumptions about users' behavior with respect to these attacks were based largely on speculation. The blinking-light defense from [1] is one of the few to have been tested by users.

2.1 Chosen Defenses

We started off this study by cataloging additional attacks and defenses with the goal of picking out two user-side detection schemes that we would include in the user-survey in addition to a defense which utilizes a scheme involving QR codes. The first defense picked is the "blinking light defense" [1] (we refer to it as such since it is originally proposed to use a router which blinks to display a sequence of light). The defense works by displaying a sequence of colors on the router (any screen capable of displaying two different colors suffices for this scheme) and the user authenticates the access point only if the color sequence matches the color sequence displayed on the device they are connecting with. The color sequence comparison behaves as a short authentication string to allow for authenticated communication through an insecure channel [7]. The way this works is that the public keys of each party are committed along with a nonce. Once each party receives the committed message from the other party, they send information on how to read the commitment. They then xor their nonces together and display the result as a color sequence.

The second defense ([2]) we decided to use in the user study detects if there are multiple APs sharing the same SSID, if that's the case it attempts to inform the user if there is a possible evil twin attack or if it is safe to connect. It does so by comparing the IP addresses, network IDs, and trace routes and uses that to conclude whether one of the APs is mimicking the other or if they are on the same network (the institution could be using two APs for load balancing purposes). It then displays a signal to the user signifying the result (red, yellow, or green). We found issues with this method as it does not tell the user which access point is the attacker and we also believe that this can be used by the attacker to give the user a false sense of security (by hosting two hotspots on the same network to receive a green light from the defense scheme), so we omit the green light feature from the user study.

We compared these schemes to the QR defense using an android app that shows the different scenarios where the defenses were simulated and where we ask the user to connect to the WiFi. We log their decisions and later ask them to complete a survey. We go over the details of the study design and the results in later sections.

3 QR Defense Design

For a user to be willing to execute a defense, it must be easy to use and they must believe that it is effective. We propose a QR code scheme to defend against evil twin attacks because it accomplishes these two aspects easily. Unlike the blinking lights scheme which has the user compare authentication strings, the QR reader will streamline the authentication process for the user making it an easier defense to execute. In addition, if the user receives the QR code through a party or physical source that they trust, they are more likely to trust the connection they establish as opposed to having their device execute a process unknown to them behind the scenes.

3.1 Threat Model and Assumptions

An attacker is able to launch an AP that mimics a restaurant's AP by spoofing any of the honest AP's public information such as SSID and BSSID. The attacker is also able to interact with the honest AP as if they are a client trying to connect. The attacker is also able to be physically in the restaurant as a client and as such have access to any physical information that a client would have. We assume that the attacker does not have access to secret key information held by the honest AP and is unable to impersonate the restaurant's staff. We also assume that the restaurant's staff are not malicious and would not collude with an attacker. This scheme also assumes that the clients will not trust another client (who might be the attacker) who provides them with a QR code and that they would only accept a QR code from the staff.

To implement this defense scheme, the restaurant is required to have an electronic device, such as a handheld digital menu, that has an authenticated communication channel with the honest AP (we will suggest an alternative in the future work section). While the QR code can be displayed by any electronic device that the attacker cannot tamper with, we assumed that it is provided by a waiter through a handheld device since directly receiving the QR code from a member of staff is likely to increase the user's sense of security regarding the defense. The client's device that they wish to connect with must have a camera to scan the QR code provided by the restaurant.

3.2 Defense Scheme

The ability of the attacker to mimic any public information displayed by the honest AP and the lack of prior information shared between the client and the honest AP (such as certificate bootstrapping) makes the evil twin indistinguishable from the honest AP by normal means. Thus, we must rely on some form of out-of-band authentication such as in the Blinking Lights defense scheme. We believe that the downfall of the Blinking Lights defense is that it requires the user to manually compare a sequence which can be time consuming and difficult depending on the distance of the screen displaying the lights. Instead we purpose performing the authentication using a QR code provided to the client by a member of staff. Since the user's device is directly reading the out-of-band information instead of the user manually comparing information, we are able to utilize more information instead of being limited to a short string.

The user and honest AP attempt to perform a Diffie-Hellman protocol using different channels. The honest AP sends its public Diffie-Hellman parameters (including their public key) to a digital device in the possession of a staff member through an authenticated channel, available by assumption, and then the digital device encodes the SSID and public parameters of the honest AP into a QR code. The user reads the QR code and then connects to the AP with the encoded SSID and sends their own public key to that AP through an insecure channel (they also send the public parameters they got from the QR code so the AP knows which private key was used to generate the public key that was shared with this client). Note that there will be multiple APs with the same SSID during an evil twin attack, in which case the device may pick any arbitrary AP to connect to because the user received the honest APs public key through an authentic channel so even if the attacker intercepts the information from the user to the honest AP, they will not be able to share a secret key with the user because the user used the honest AP's public key to generate it and the attacker does not have access to the honest AP's secret key. Once the honest AP receives the client's public key parameters, they can figure out the shared secret key using their own private key. If the user is unable to successfully share a secret key with the AP at the end of the protocol, then it suspects that it is connected to the evil twin and connects to the next AP instead (and performs the protocol again). The attacker may attempt to forward packets to the honest AP so the protocol would be successful, but in that case the user will end up sharing a secret key with the honest AP that the attacker does not know.

4 Study Design

The user-study we conducted is a simulation where the user is a customer at a restaurant and they wish to connect to the restaurant's WiFi. During the simulation, the experimenter takes the role of a waiter/waitress. To keep the simulation invariant of the experimenter performing the user-study, we created a script for introducing each scenario and defense to the participants, as well as some guidelines on what knowledge the waiter/waitress has when answering questions. The experiment consisted of six scenarios, in each of which the user's goal is to connect to the restaurant's WiFi. During each of the scenarios, there is at least one malicious AP. The participants were not told of the existence of an attacker for the first two scenarios.

The first two scenarios did not include any defenses. In the first scenario, the malicious AP was using the same SSID as the honest AP and the only difference that the participant can gleam is the different strengths of the AP. In the second scenario, the malicious AP uses a slightly different SSID and does not require a password (as opposed to the honest AP which requires a password) in an attempt to tempt the user to connect to it.

After the second scenario, we explain (on a surface level) the evil twin attack to the participants. In the third scenario, we explain the blinking light defense to the user. The user is now able to hold an AP's name down and this will issue a blink request to the router. The attacker in this scenario knows about the defense, so whenever the user holds down the malicious AP the attacker will request to perform a short authentication string protocol with the honest AP to make it display a light sequence. This might fool the user into thinking that they connected to the right AP (since the screen at the restaurant displayed a sequence), but the attacker cannot have the router show the same color pattern as displayed on the user's phone (with high probability) unless they forward the message committed by the honest AP (which contains its public key and the nonce required to calculate the correct light sequence).

The fourth and fifth scenarios use the color defense. In the fourth scenario, an attacker who is mimicking the SSID of the honest AP is performing a man-in-the-middle attack and is recognized by the defense. Thus, the malicious AP is labeled with a red dot. In the fifth scenario, the attacker launches two malicious APs. One AP spoofs the honest APs SSID, MAC, and IP address so the defense labels both the first malicious AP and the honest AP with yellow dots. The second malicious AP has a slight variation in the SSID, so it is not labeled by the defense, and it requires no password. The attacker does this to make the clients suspicious of the honest AP, which we conjectured would make the clients trust the second malicious AP more.

In the sixth scenario, the attacker is performing an evil twin attack again. The user is instructed to perform the QR code defense which consists of opening a QR reader (through a button provided at the bottom of the WiFi list) and scanning the QR code which behaves as an authenticated communication channel that delivers the honest AP's public key to the user. Since the user receives the public key through an authenticated channel, the attacker cannot end up with a shared secret with the user (the attacker is able to perform the key exchange with the honest AP, but this is not special since the attacker is allowed to behave as a customer in any case).

For each scenario we used a different password because we are timing the users, so this way they remain consistent between scenarios. Each time the user connects to an AP, we record which AP they connected to and the time it took them. We wrap this data in a json object and send it over to our node js server hosted on heroku. This server acts both as our logging server and it simulates the blinking router for the third scenario. We used socket.io for the communication between the android phone and the node js server.

After the user is done with all the scenarios, they will be directed to a survey.

5 Results

We recruited 24 participants for our user study to evaluate the usefulness of the QR code defense. One of the logs was not recorded, so we have 23 logs (the logs include information on which AP they connected to and the time it took them to connect. However, all 24 users took the survey. This section will discuss the results of the study, focusing mainly on the user's response to the survey.

One of the questions that we asked in the survey was "How willing are you to connect to a public hotspot with no password". the answer is a score from 1 (very unlikely) to 10 (very likely) and figure 1 shows the results.

Figure 2 gives a good overview of the number of users that connected to the adversary vs those who connected to the authentic access point in each of the 6 scenarios. For example in the first scenario we see that most users connected to the honest AP. However, it is important to note that in this case the honest AP and attacker behave identically so it is arbitrary which we choose as the legitimate AP for the sake of reporting the results. We chose the honest AP to be the one with the higher signal strength, but this tells us that user's are more willing to connect to the stronger AP when presented with identical ones (since they are unable to tell which is the attacker, or even know that an attack is occurring). In the second scenario we can see that a significant number of users connected to the adversary. In this scenario, the adversary had a similar name to the actual access point, had no password and also had a stronger signal. In the third scenario where the blinking light defense is deployed, we can see that some users still connected to the adversary. This is because for some of the users, when they sent a blink request to the router, they were satisfied that the router showed a color sequence, even though it was not the same color sequence that was displayed on the phone.

Figure 1: the results when users were asked, “How willing are you to connect to a public hotspot with no password?" A score of 1 signifies very unwilling and a score of 10 means very willing

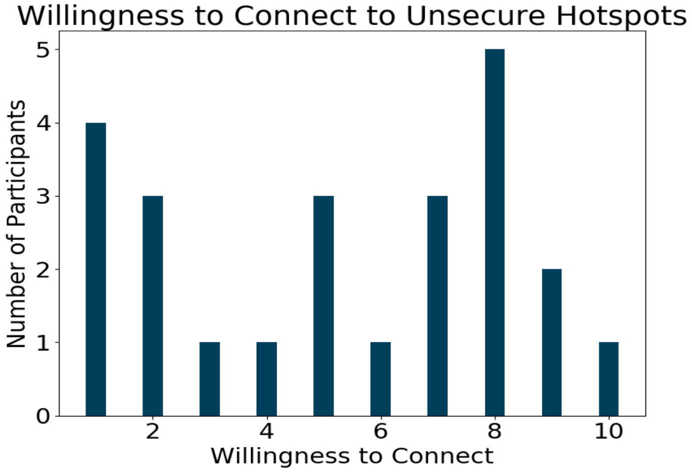
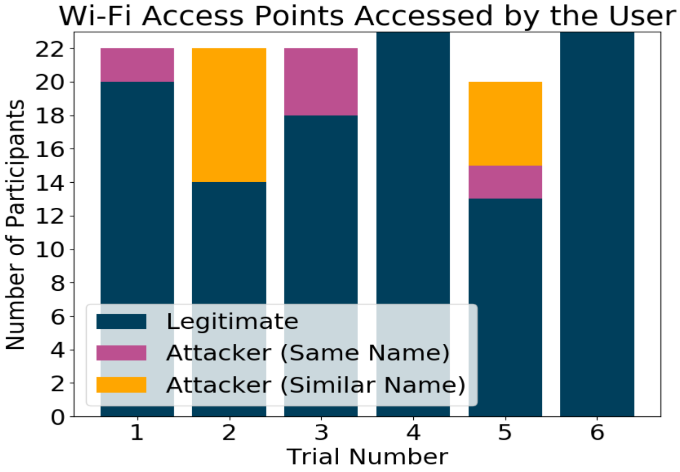


Figure 2: a graph of the access points to which users connected during the simulation

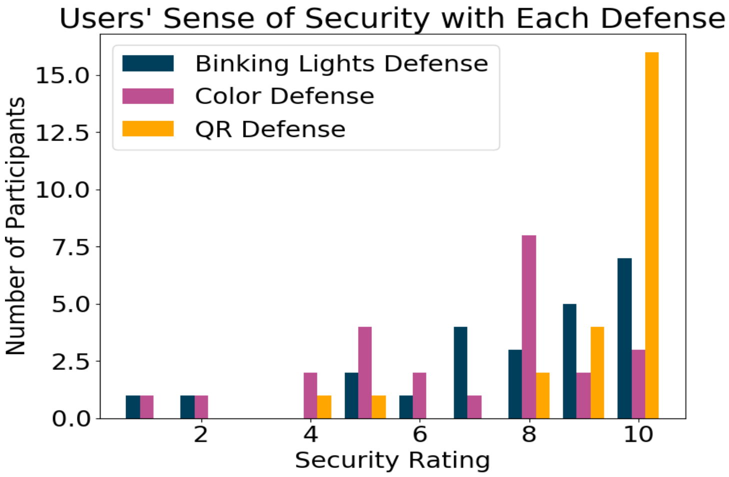


This was despite the fact that it was explained to users that the pattern must be the same on both devices. In the next two scenarios, the color defense was deployed, in the 4th scenario, there was one AP marked in red so users knew not to connect to it. However, in the 5th scenario, there were two APs with the same name and each was marked in yellow so the user did not know which to choose. This can account for why so many of them ended up connecting to the adversary. In the last scenario the QR defense is deployed and this defense does not leave the decision up to the user and automates the connection process, thus each user connected to the right access point when they scanned the QR code. The small gaps are due to users choosing not to connect to one of those networks.

We can see in figure 1 that most people gave a score of 5 or higher. This means that people are likely unaware of the risk of WiFi phishing or simply prefer to gain internet access despite the risk. That is why ease of use of a user-side detection scheme is an important aspect, as those who are willing to connect to public APs with no concern are unlikely to be willing to perform a cumbersome scheme.

We can clearly see in figure 3 that the color defense did not give as much of as sense of security as the other defenses. This is likely due to the fact that one of the scenarios in the study, which we described earlier, had an attacker perform an evil twin attack in which they spoofed their IP address to match that of the honest AP. The color defence described by [2] is able to detect an evil twin attack, but is unable to distinguish which one of them is the attacker. The inability of the defense from distinguishing the attacker understandably raises doubts amongst regarding the usefulness of the defense.

Figure 3: a graph of the users' sense of security when using each defense



We can see that most people gave the QR defense a score of 8 or higher and the blinking light a score of 7 or higher. Noticeably, two thirds of participants gave the QR defense the maximum possible rating for their trust in its success.

Some people also asked what would happen if more than one person is trying to connect to the wifi with the blinking light defense, this is indeed one of the limitations of the defense.

Figure 4: ease of use of each defense, according to participants. A score of 1 signifies very hard, while a score of 10 signifies very easy

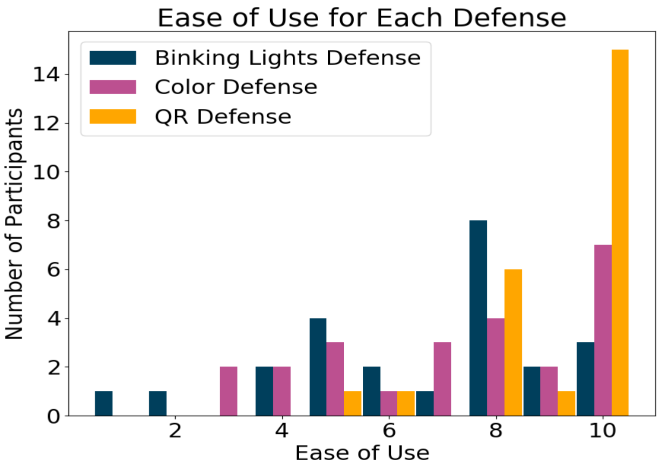
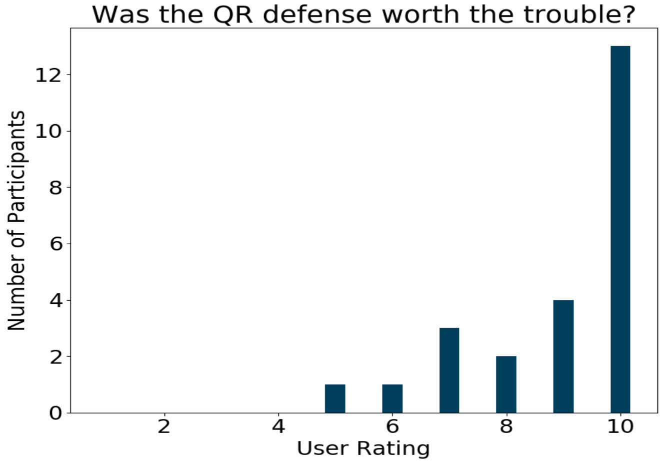


Figure 4 showcases the ease of use of each defense (a score of 1 being very hard and a score of 10 being very easy). While the Color Defense seems intuitively to be the simplest, it received the worst ranking. We conjecture that the discrepancy between the reported ease of use of the Color defense by the users and the actual simplicity of the defense might be due to the confusion caused when the defense is able to detect an evil twin attack occuring but is unable to distinguish the evil twin (labeling both as yellow). If the user expects the defense to help them connect to the honest AP, but it at best gives them an even chance of connecting to the attacker then they might feel that the defense is difficult to use. The QR defense received much higher scores, with almost two thirds of the participants giving it the maximum score. One of the main reasons we mentioned that we purposed the QR code because of is that it streamlines the authentication process that the Blinking Lights defense conducts, and we see that reflected in the results since it received much more favorable results in comparison.

The main thing about a user-side defense is that the users must feel that the security they gain from using the defense is worth the effort it takes of them to use it. Thus, we directly asked the users if to rate the statemtn "The security gained

Figure 5: The user's were asked if they agreed with the statement "The QR defense was worth the trouble of using it". A score of 1 signifies "strongly disagree" and a score of 10 signifies "strongly agree"



from QR code defense was worth the trouble of using it". Figure 5 shows that most people did think that the security gained from the defense was worth the trouble of using it.

We also recorded the time that it took each user to connect to an AP. The median time it took customers to connect to the AP using the QR defense was 9.567 seconds (including the time it took for them to request the QR code). Which is a reasonable time to expect users to take to connect to private WiFi. In fact, the majority of users recorded a lower time connecting using the QR code than connecting with the absence of any defense scheme. This is because the user still needs to inquire about the WiFi password and type it in, which typically takes longer than scanning a QR code. However, it is not very helpful to compare these two scenarios directly because some delay might have also been incurred due to the existence of an evil twin attack which causes some hesitation on the user's part.

6 Future work

The current design of the QR code defense scheme restricts the strategy to institutions that have staff who are able to respond to clients in a timely manner, and it requires that the staff carry an electronic device with an authentic communication channel with the honest AP. The reason a digital device is required is that the honest AP is regenerating the parameters used for the Diffie-Hellman protocol with each user. However, it is possible to perform instead a modified TLS protocol where instead of authenticating the server's public key with a certificate authority, the user receives the public key from a QR code in a trusted location or with a trusted party. The user then completes the rest of a TLS protocol with the server. Since the public key does not need to change frequently in this protocol, a printed out QR code would work. This requires the institution hosting the honest AP to update the QR code when the public key needs to be changed. Work on this kind of scheme will require evaluating how frequently the public keys need to be changed and, if the QR code is not held by staff members, how to place the QR code securely so the attacker cannot replace it with their own QR code. The user-study we conducted had the QR code be presented by the waitstaff through a digital screen, so future work would be developing a QR code scheme that uses non-digitally rendered QR codes. This requires an evaluation on how printed QR code (and mounted QR codes in the case they are not presented by staff) affect the sense of security the users receive from the scheme.

7 Conclusion

From the data we collected, we can draw two main conclusions. First, despite the prevalence of evil-twin attacks, users generally have not changed their behavior to avoid these attacks, nor do they seem to understand the risks associated with connecting to unsecure public hotspots. Thus, we recommend that efforts be put into place to inform users about evil twin attacks and how to spot suspicious access points, in a similar manner to how users are now encouraged to develop strong passwords. This finding also underscored that ease of use is vital, as several users neglected to use the blinking-light defense or used it incorrectly because it was slightly harder to use than the others. On the other hand, every user connected to the right access point in the QR-code trial and the first trial of the color defense because they were so easy to use.

Second, we conclude from this study that our proposed QR scheme was perceived to be easier and more secure than the other two defenses that we tried, and the users connected to the right access point every time. With the increasing ubiquity of QR-code scanners, connecting to an access point using this method would not be very foreign to the user. Thus, we conclude that this defense is worth pursuing and, if put into practice, can decrease the risk of evil-twin attacks generally.

References

[1] Volker Roth, Wolfgang Polak, Eleanor Reiffel, and Thea Turner. Simple and effective defense against evil twin access points. *Proceedings of the first ACM conference on Wireless network security*, pages 220–235, 2008.

[2] Somayek Nikbakhsh, Azizah Bt Abdul Manaf, Mazdak Zamani, and Maziar Jaanbeglou. A novel approach for rogue access point detection on the client-side. *26th International Conference on Advanced Information Networking and Applications Workshops*, pages 684–687, 2012.

[3] Hossen Mustafa and Wenyuan Xu. Cetad: Detecting evil twin access point attacks in wireless hotspots. *IEEE Conference on Communications and Network Security*, 2014.

[4] Diogo Mónica and Carlos Ribiero. Wifihop - mitigating the evil twin attack through multi-hop detection. *Proceedings of the 16th European conference on Research in computer security*, pages 21–39, 2011.

[5] Harold Gonzales, Kevin Bauer, Janne Lindqvist, Damon McCoy, and Douglas Sicker. Practical defenses for evil twin attacks in 802.11. *2010 IEEE Global Telecommunications Conference*, pages 1–6, 2010.

[6] Mayank Agarwal, Santosh Biswas, and Sukumar Nandi. An efficient scheme to detect evil twin rogue access point attacks in 802.11 wi-fi networks. *International Journal of Wireless Information Networks*, 25:130–145, 2018.

[7] Serge Vaudenay. Secure communications over insecure channels based on short authenticated strings. In *Annual International Cryptology Conference*, pages 309–326. Springer, 2005.