Secure User-Friendly Wi-Fi Access Point Joining: International Wireless Communications and Networking Conference

SEIGNEUR, Jean-Marc, BALLESTER LAFUENTE, Carlos, MATOS, Alfredo

Abstract

In this paper, we discuss the remaining security risk of Wi-Fi AP spoofing with current AP joining approaches and how a new solution has been developed as part of the ULOOP project in order to be more user-friendly and secure. It was an important step for increased security because our evaluation shows that even computer aware users do not know or do not bother about this issue although it is a real risk that current approaches do not solve.

Reference


Available at:
http://archive-ouverte.unige.ch/unige:55387

Disclaimer: layout of this document may differ from the published version.
Secure User-Friendly Wi-Fi Access Point Joining

Jean-Marc Seigneur, Carlos Ballestre Lafuente
University of Geneva
Switzerland
Jean-Marc.Seigneur@trustcomp.org

Alfredo Matos
Caixa Mágica Software
Portugal
Alfredo.Matos@caixamagica.pt

Abstract—In this paper, we discuss the remaining security risk of Wi-Fi AP spoofing with current AP joining approaches and how a new solution has been developed as part of the ULOOP project in order to be more user-friendly and secure. It was an important step for increased security because our evaluation shows that even computer aware users do not know or do not bother about this issue although it is a real risk that current approaches do not solve.

Keywords—Wi-Fi; security; user-friendliness; access point

I. INTRODUCTION

The number of Wireless-Fidelity (Wi-Fi) Access Points (APs) in the world has increased significantly: they operate widely across several locations like airports, cafes, businesses and university campuses. This fact coupled with the inherent vulnerabilities of the deployed technologies, can provoke serious security breaches in access networks. In addition to typical network threats, wireless networks present several challenges and specific attack types. This is due to the wide open air nature of the channel, allowing more attacks, spectrum restrictions and constant topology changes because of node mobility. Currently it is common to hear about security threats such as Service Set Identifier (SSID) spoofing using soft APs. To steal credit card numbers and other personal information, thieves are using a soft AP to masquerade as a legitimate wireless AP. For instance, it has been reported [1] that fake Wi-Fi networks have been set up in many airports in order to capture users’ sensitive information as they surf the Web during their connection to those networks. It is important to know whether the Wi-Fi APs within range are trustworthy or not because in some locations it is not rare to find more than five potential Wi-Fi networks to connect to. In previous work [2], we have proposed to compute the trustworthiness in nearby APs based on shared previous interactions between the users of these APs and their outcomes. However, we assumed that these APs were securely authenticated and could not be spoofed. As introduced above, in real life with current Wi-Fi authentication means, being password-less Wi-Fi or password-protected Wi-Fi with WEP or WPA, spoofing of APs happens. In this paper, in Section 2, we survey how the association of a user device to a nearby AP, which we call “joining”, is done, including authentication, and when spoofing can occur. Then, in Section 3, we explain how we have proposed to tackle this problem in the User-Centric Wireless Local Loop (ULOOP) FP7 project [7] through signed contextual QRCode-based. In Section 4, we provide a brief evaluation of the users awareness of the risk of known Wi-Fi AP spoofing and validate embedding signed context for AP joining. Finally, we conclude in Section 5.

II. WI-FI JOINING STATE OF THE ART

A. Current Manual Joining

Nowadays, when a user wants to connect her device to a nearby AP, assuming she had not started the functionality to look for nearby APs, she starts on her device this functionality. If she had agreed to join automatically the APs that she had already joined once in the past, the device selects automatically one of the “known” APs. On most devices, it is also possible for her to specify her list of preferred APs and the AP that is ranked first in the list of “known” APs is automatically selected. That automatic selection has created a security hole and spoofing attacks starting from this hole have successfully been carried out [4]. As there is no authentication of the claimed SSID text string, when the original AP is not switched on, it is easy for an attacker to set up an AP with the same SSID as the original one. The other basic spoofing attacks are also based on this same security hole of no authentication of the claimed SSID text string. If there is no “known” APs, then the list of nearby APs appears on her device and she has to choose one of them without much more information than the SSID text string displayed along with the type of security protection for the network (open, WEP, WPA, WPA2...). Again, the name of an AP previously existing at a location could be spoofed and the spoofing AP could agree on any passwords proposed by the user’s device in order to let the device network requests and responses go through the AP and try to collect any non-encrypted information sent through the spoofing AP. More advanced attacks based on this security hole are possible, for example, if the AP is not switched off, the attack is combined with attacks trying to limit the wireless range of the AP: “WLAN signals are easy to jam, signals from legitimate access points can be easily eliminated” [4]. In public places, a few commercials APs may be available. In this case, joining their SSID does not require a password, but then when Web access is requested, the user’s Web browser is redirected to the AP Web portal where a login and password can be typed or retrieved after being purchased. In home environments, the AP’s owner usually protects the access to her AP through password-protected WPA. If someone wants to use her AP, she has to disclose the AP password to this person, which may be quite difficult if the chosen protecting key is quite long. In order to ease the exchange of long keys, which are difficult to remember, the next subsection describes how different techniques have been proposed to facilitate a more user-

...
friendly experience for joining an AP being protected by a password or not.

B. Related Work Joining based on Barcodes, QR Codes and NFC

As having to type by hand the necessary password for a selected Wi-Fi AP protected by a password (either through WEP, WPA…) is not so easy, other means to pass the password and even all sufficient information to automatically configure the access to an AP have been thought. One interesting approach has been to encode in a known format that sufficient information into a QR Code in order for an application or a service running on the user’s device to be able, once the QR Code is flashed (the device takes a photo of the QR Code), to decode that information and automatically configure its access to the AP. The known way to encode the sufficient AP security configuration information into a QR Code is available in related literature [3]. Recently, a few projects have started [5,6] to propose mobile applications and services to easily encode and decode QR Codes according to this known format. One of those mobile applications [6] also allows the user who has an NFC-enabled phone and APs with NFC tags to pass the Wi-Fi security configuration through NFC. Although these mobile applications considerably facilitate the access to a password protected AP, they do not solve the AP spoofing problem depicted above.

In order to solve the spoofing attacks mentioned above, [8] has proposed to use 2D barcodes (Data Matrix, PDF417, MaxiCode…) to embed cryptographic materials. These cryptographic materials can easily be flashed or scanned by a user when she is nearby an AP, for example, a public key embedded in the 2D barcode, which is used for bi-directional authentication when both users can scan the generated barcodes. They discuss that embedding a hash of the public key rather than the full public key opens the door for an attack on the hash based on cryptanalytic or brute force. As the 2D barcode they have used for implementation could not store a full public key, they proposed to display several 2D barcodes that each of them would contain a piece of the digest or of the public key. They survey the other means to pass the hash of a public key. However, as Raph Levien underlined, examining key or hash materials is “a serious usability problem” [9]. They conclude that these other means are less easy to use (difficulties to spell very long hash of keys or to recognize visual representations by the user without error…) or prone to Man-in-the-Middle (MiM) attacks. Even previous approaches based on cryptographic materials can be prone to MiM spoofing AP attacks, namely, MiM attacks based on location as depicted in the figure below, whereby a MiM is present in the two locations where the mobile user is located and where the legitimate AP is located. The MiM can set up an AP with the same SSID as the legitimate AP in the other location and simply forward the exchanges between the mobile device and the legitimate AP.

After joining, if the full connection is not encrypted between the mobile device and its remote servers, the MiM AP can see the unencrypted communication, possibly including sensitive information. [10, 11] proposed a simple authentication method that fits into the extensible authentication protocol (EAP) framework under 802.1X based

![Image](image_url)
uncover the concept of trust-on-first-use and uses contextual information, e.g., location of nearby APs and their signal strength, to allow the mobile clients to learn the context of other access points around and use it to recognize the AP the next time the mobile client envisages to use it. Unfortunately, their approach is still vulnerable the first time the AP is used because there is no previous context known. Woodside [13] used 2D barcodes called semacodes to store the URL containing information about the location where the barcodes were installed but no cryptographic materials were used to certify the stored information.

III. ULOOP SECURE SIGNED CONTEXT-BASED QRCODE JOINING

In ULOOP, the main identifier of a user is the hash of the public key of a key pair that the user owns. This main identifier is called a crypto-id because it is possible for the user to prove that she really owns the secret, i.e., the private key, by signing messages she sends through the device. Unfortunately, as mentioned in the previous sections, passing and verifying the hash of a key is not user-friendly. Thus, we have implemented and checked that embedding the full public key in a QRCODE is possible in order to avoid the brute force attack on a key hash as mentioned the previous section. As argued in [8], MiM local attacks when using flashing/scanning 2D barcodes are not considered feasible. Concerning the remaining issue of location-based MiM AP spoofing attacks, instead of letting the mobile clients learning the trusted context of the AP as in [10], we also embed context into the QRCODE but in our approach it is specified and signed by the owner of the AP as depicted in the figure below. In our current implementation, the trusted location of the AP, from the point of view of the owner, is stored and signed along with the SSID and AP password in the QRCODE.

Thus, the mobile client can check that embedded location, compare it to its actual location and decide if it seems close enough in order to avoid location-based MiM attacks as described above. The mobile client is currently implemented in Android. It may be argued that the location provided by mobile phones in some places, for example, inside buildings, may be not very accurate or even impossible to retrieve. However, new means to improve mobile phone location are nowadays intensively researched, for example, based on a combination of GPS and other measurements such as signal prints. By the way, signal prints have been proven effective to detect masquerading and resource-depletion attacks with high probability in a building [12]. Signal prints are another example of context that could be embedded and signed in the QRCODE. Anyway, we have also embedded a textual field that can be displayed to the user in case the location accuracy seems too low or is not available, e.g., due to unavailable GPS information. In this case, the mobile device would show the text location to the user and she would decide if it corresponds enough to her current location, for example, “Starbucks, Geneva Plain Palais street”.

At QRCODE creation time, the following template is used to embed the necessary information in the QRCODE.

```
ssid=ap_ssid_value;password=ap_password_value;lat=ap_location_latitude_value;lon=ap_location_longitude_value;user_readable_location=user_readable_location_value;signature=signature_of_the_previous_fields;public_key=full_user_crypto_id_public_key
```

![Figure 2. Signed Location-based QRCODE against Man in the Middle Wi-Fi Joining Attack](image-url)
In Section 4, we validate that this required information can fit into one QRCode. Then, once a mobile user wants to join a specific AP, which is advertised in proximity of the user with a specific sticked QRCode generated previously with another mobile client, she can simply flash that QRCode and if the location matches the location signed in the QRCode, the mobile client displays that the QRCode is valid along with the SSID and password that can be used to connect to this AP. We have used a 10-meter radius so far based on GPS and display the textual location information if the GPS is not available. To mitigate the remaining risk of an attack based on cloaking or limiting the range of the legitimate AP [4] within these 10 meters and relaying between the mobile client and the AP, we either assume that those remaining attacks can be detected as argued in [12] or we encrypt all communication between the mobile client and the AP as it is possible in ULOOP because ULOOP also modifies the AP in addition to its mobile client application. The following UML sequence diagram in Figure 3 depicts the process occurring after the user has clicked on a button asking to join a specific nearby AP.

As in [10], until now, we have proposed to identify the AP with its self-signed certificates and we do not require centralized certificate authorities, pre-shared secrets or pre-existing trust technologies. Thus, one may wonder how a mobile device can decide whether or not choosing an AP even if it is now strongly authenticated because there is no link to a real world entity that could be sued. Indeed, the certificate is self-signed and no third party has certified it to be linked to a real world entity. As mentioned in the introduction, computing trust in the AP is beyond the scope of this paper. However, we have proposed to compute the trustworthiness in nearby APs based on shared previous interactions between the users of these APs and their outcomes in previous work [2] although we assumed that these APs were securely authenticated and could not be spoofed, which is what we solve in this paper. In [11], they not only envisaged collaborative reporting of AP trustworthiness but also assumed mitigating Sybil attacks [17] “by requiring that reporting users first register with an account authority that can perform micro-transactions out-of-band”. For information, in ULOOP, we also assume that a user can have only one validated crypto-id based on the validation of a ULOOP certified identity validator authority. There may be several certified identity validator authorities in ULOOP. In ULOOP, until the crypto-id is not validated by such an identity validator authority, the crypto-id is considered in “quarantine” and only self-signed. A crypto-id in quarantine is considered less trustworthy and is granted fewer privileges. In [8], they consider Trusted Platform Module (TPM) compliant computing platforms and may use Attestation Identity Keys (AIK) certified by the TPM ecosystem certification authorities (although such an ecosystem has still to really emerge).

IV. EVALUATION

In this section, we first show our validation results regarding the embedding of signed context information in a QRCode for secure and easy Wi-Fi AP joining. Then, we show the results of our investigation regarding the awareness of the users about Wi-Fi AP spoofing and its related risk that we try to mitigate with the solutions presented in the previous section.

A. Validation Results of Signed Context Information in a QRCode

The number of characters in a QRCode is limited, even
more when the QRCode is flashed by a smartphone and it is not the original image generated by software that is directly processed. As shown in the following Figure 4, which corresponds to a reading of the QRCode with the standard Mobiletag Iphone application, we have been able to store all the required information with a decent number of characters per field but only for RSA signatures based on 1028 bits keys. As argued in [8], for greater keys length, we may use a sequence of QRCodes to be flashed, but for the moment, we keep 1028 bits keys that still provide security without having to flash multiple QRCodes because it is far less user-friendly to flash many QRCodes in sequence.

We have used two phones (an iPhone 4S and a Samsung Galaxy Nexus) to flash QRCodes generated with a Java-based implementation. The two phones were consistently able to read the generated QR Codes with a size of 250 pixels per 250 pixels when using RSA 1028 bits keys and the following fields but never with RSA 2056 bits keys (too much information was present in the QRCode). Trying with a bigger image of 400 pixels per 400 pixels did not help either and we did not try bigger images to as they could not fit on a smartphone screen to the contrary of a 250 pixels per 250 pixels. Trying to GZip compress, the resulting text to be embedded in the QRCode did not allow us to store 2056 bits RSA key along with all other required information either. In the following image, the RSA signature has a fixed length and is Base64 encoded. This signature results in 172 characters for RSA 1028 bits keys. For a 2056 bits RSA key, it resulted in 344 characters. We then use the suitable maximum following length for the other required information: 70 characters for the AP password (e.g., WPA maximum key length security is 63 characters); 20 characters for the SSID; 10 characters for the latitude; 10 characters for the longitude; 60 characters for the textual description of the location.

In total, including the public key encoded as PEM, the content was 697 characters long, which is within the 800 characters that are usually recommended not to go over in a QRCode.

B. Wi-Fi AP Spoofing Users Online Survey

During summer 2012, we created a short survey and sent it to a list of users who are subscribed to a marketing database and who are interested in computer programming and speak English or French. 1767 user answered, which is quite a large number of answers. We asked them the following question “Do you know that a Wi-Fi hotspot public access point name can be easily impersonated and that it can be a security risk for you?” They could reply one of the following answers “Yes; No; I don’t care” and optionally add a textual comment. 5 of them used that comment option and answered: yes with the following comment “but it is possible to secure the link” ; yes with the following comment “VERY COMMON AND IT CAN CAUSE HAVOC!!!!!!”; yes with the following comment “Obvious .. :P”; yes with the following comment “Honeypot :- )”; no with the following comment “Yes, Now i know. :P”. Among the English speaking people, 540 replied “yes”, 185 replied “no” and 1017 replied “I don’t care”. The image below indicates the percentages for each answer type.

V. CONCLUSION

Although these users are interested in computer programming, it is surprising to see that 58.4% of 1743 English-speaking users did not care really care about this issue and that 10.6% did not know it. Concerning the comment on securing the connection, few users would know how to really secure their connection. Furthermore, the fact that many of them answered that they do not care, leaves us to think that they would not take the time to secure it if it is not automated, which is not the case today with current Wi-Fi connections.

Figure 5. Spoofing Awareness Results

Although these users are interested in computer programming, it is surprising to see that 58.4% of 1743 English-speaking users did not care really care about this issue and that 10.6% did not know it. Concerning the comment on securing the connection, few users would know how to really secure their connection. Furthermore, the fact that many of them answered that they do not care, leaves us to think that they would not take the time to secure it if it is not automated, which is not the case today with current Wi-Fi connections.

V. CONCLUSION

In this paper, we have underlined the remaining security risk of Wi-Fi AP spoofing with current approaches and how a new solution developed as part of ULOOP could be more user-friendly and secure. The remaining security risk seems still
quite important because our evaluation shows that even computer aware users do not know or do not bother about this issue although it is a real risk that current approaches do not solve. Thus, our new approaches that transparently mitigate this issue without bothering the users seem quite suitable and necessary.

ACKNOWLEDGMENT

The research leading to these results has received funding from the EU IST Seventh Framework Programme (FP7/2007-2013) under grant agreement n 257418, project ULOOP (User-centric Wireless Local Loop), participants: Alcatel-Lucent Bell Labs, (FR), COFAC/University Lusófona (PT), Huawei Technologies Duesseldorf GmbH (DE), ARIA S.p.A (IT), Caixa Mágica Software (PT), FON Wireless Ltd (UK), Technische Universität Berlin (DE), University of Kent (UK), Université de Genève (CH), Level7srlu (IT), University of Urbino (IT). The concepts described in this paper are under investigation in this EU IST FP7 ULOOP project but do not reflect consensus of the consortium nor of the project towards the discussed theme.

REFERENCES

[5] https://sites.google.com/site/easywificonnect/