QR Code Security - How Secure and Usable Apps Can Protect Users Against Malicious QR Codes

Katharina Krombholz*, Peter Frühwirt*, Thomas Rieder†, Ioannis Kapsalis‡, Johanna Ullrich* and Edgar Weippl*
*SBA Research, Vienna, Austria
Email: kkrombholz,pfrühwirt,jullrich,eweippl@sba-research.org
†Vienna University of Technology, Austria
Email: thomas@rieder.io
‡Aalto University, Helsinki, Finland
E-Mail: ioannis.kapsalis@aalto.fi

***Abstract*—QR codes have emerged as a popular medium to make content instantly accessible. With their high information density and robust error correction, they have found their way to the mobile ecosystem. However, QR codes have also proven to be an efficient attack vector, e.g. to perform phishing attacks. Attackers distribute malicious codes under false pretenses in busy places or paste malicious QR codes over already existing ones on billboards. Ultimately, people depend on reader software to ascertain if a given QR code is benign or malicious. In this paper, we present a comprehensive analysis of QR code security. We determine why users are still susceptible to QR code based attacks and why currently deployed smartphone apps are unable to mitigate these attacks. Based on our findings, we present a set of design recommendations to build usable and secure mobile applications. To evaluate our guidelines, we implemented a prototype and found that secure and usable apps can effectively protect users from malicious QR codes.**

***Keywords*-Security, Privacy, Usability, QR Codes**

I. Introduction

QR (Quick Response) codes are two-dimensional matrix barcodes that are used to encode information. In recent years, they have increasingly found their way into urban spaces and mobile communication to make the Web instantly accessible. The most common use case is to encode a link or other textual information to make it instantly available, obviating the need for a user to type a URL manually. Due to their high information density and robustness, QR codes have gained popularity as part of the *Internet of Things*. Applications range from simple web links in billboard advertising to monetary transactions during which highly sensitive data is handled. Besides a broad range of advantages, QR code processing on mobile devices often implies distinct security vulnerabilities of which the average user is mostly unaware. Attackers misuse QR codes to encode malicious links that lead e.g. to phishing sites or to the execution of malicious code. These malicious QR codes can be printed on small stickers and pasted over preexisting QR codes on billboard advertisements [23]. The importance of QR code security has been acknowledged by previous research such as Seeburger et al. [20] and Vidas et al. [24]. However, to the best of our knowledge, we are the first to investigate the impact of usability on risk perception with respect to malicious QR codes. As QR codes are increasingly used as an attack vector, there is a need to develop effective tools that help users to protect themselves against such attacks. In this paper, we provide an extensive study of QR code reader vulnerabilities with an emphasis on usable security measures. In the first phase of this study we focused on the decoder software and in the second phase on the user. During the first phase, we analyzed QR code reader software for mobile devices. We identified the 12 most frequently downloaded QR code reader applications for Android, iOS and Windows Phone and analyzed them with respect to security protection mechanisms and privacy violations. To test these applications, we produced a set of malicious QR codes and observed how the software processed them. Furthermore, we used HTTP(S) interception to monitor the information which was exchanged between the QR code reader application and the related web service. We found that most applications were not only unable to identify malicious QR codes and hence redirected the user to a malicious web site, but also significantly violated the user's privacy by transmitting personal information to third parties. The second part of our survey consists of an intercultural comparative user study. We identified intercultural factors of security awareness with respect to QR code security. To conduct this study, we encoded a link to a short online survey in QR codes and used them to simulate a phishing attack. We deployed 784 QR code stickers in different locations, such as public toilets, bus-stops and cafeterias in four different European capitals, namely Athens (Greece), Helsinki (Finland), Paris (France) and Vienna (Austria). Our findings not only identify the intercultural differences amongst Europeans but also highlight the need for security improvements to make QR code processing a secure user experience. Based on our comprehensive analysis of security and privacy flaws concerning smartphone applications, we propose a set of design recommendations to improve the QR code encoding scheme, the reader software and the app's usability. To evaluate these recommendations, we implemented a prototype application where security, privacy and usability are considered likewise and showed that our recommendations can help to build applications that are both secure and usable.

978-1-4673-6590-1/15 $31.00 © 2015 IEEE
DOI 10.1109/ARES.2015.84

II. Background & Related Work

QR codes in general consist of different areas reserved for specific purposes. Some areas are functional parts of the QR code and cannot be recovered with error correction. The data is encoded in black and white *modules* [6]. When QR codes are used as an attack vector, they can be either partially modified or completely pasted over. We refer to partial modifications if singular modules are inverted from white to black and vice-versa. In general we distinguish two types of malicious modifications. The first approach includes the modification of black and white pixels, whereas the second one restricts to alterations from white to black. The latter scenario resembles an attacker modifying an existing QR code by using a black pen as described by [12]. Considering the structure of a QR code, the modification of specific modules can influence the code: changing the character encoding forces the QR code scanner to interpret the data differently. The character count indicator reveals the length of the data and a modification thereof tricks the scanner into reading less of the data or into interpreting modules that are not intended to contain data. Furthermore, various encodings can be mixed within the data to change the modes or to insert and delete respective segments. Attackers can also hide malware by misusing control characters [14]. Attacks performed via QR codes are mainly targeted at automated processes and human interactions. According to [21], attacks on automated processes refer to SQL and command injections tricking the system into performing something other than intended by the process owner. Furthermore, a QR code can be misused for browser-based exploits and cross-site scripting attacks [13]. Dabroswki et al. [5] proposed barcode-in-barcode attacks by generating barcode polyglots. In this paper, we focus on the human factor as the most vulnerable component. Humans are not able to decode a QR code by simply looking at it. Hence usable tools are necessary to support the user in deciding whether a QR code is malicious or benign. The importance of human factors in QR code-based attacks has already been acknowledged by the scientific community. Seeburger et al. [20] investigate how users interacted with QR Codes stickers in urban spaces called *PlaceTagz*. These *PlaceTagz* were deployed in different locations in Melbourne such as cafeterias, libraries and public toilets. Their results suggest that curiosity is the main motive for dwellers to scan non-contextual QR codes. Thus, with curiosity being the major motivation to interact with an unknown source, users are ignoring the security threats associated with QR codes from unverified sources or are unaware of them. Vidas et al. [24] described QR code-initiated phishing attacks by conducting two experiments in the city of Pittsburgh, a *surveillance* and a *QRishing* experiment. Within their surveillance experiment, they observed how users interacted with the code and if they scanned the codes or not. Furthermore, they observed the proportion of users who scanned the code but refused to visit the encoded URL by visually monitoring user interactions with QR codes. To do so, they deployed a poster with a QR code and a camera to record the user interactions. In their so-called *QRishing* experiment, they deployed QR codes on three different types of posters and flyers to assess the susceptibility of such a phishing attack. In their codes, a link to a survey was encoded. This survey contained a set of questions to identify the initiatives and the behavior of the people that scanned the QR codes. Similar to Seeburger et al. [20], Vidas et al. [24] found that curiosity is the main motivation for smartphone users to scan a code. The research conducted by Vidas et al. [24] is significantly related to our research. However, it does not include an intercultural comparison of the user awareness and focusses more on how people generally interact with QR codes in urban spaces. Yao et al. [26] proposed a solution called *SafeQR* to harden QR code readers with existing APIs as well as a visual warning scheme for Android apps. Based on the findings by Yao et al. we propose our set of guidelines to harden the QR code itself as well as the reader app and to support the user in verifying the originator of the QR code and to differentiate malicious from benign QR codes. In recent years, intercultural factors of privacy perceptions and security awareness have been examined in scientific literature.

III. Evaluation of QR Code Readers

For our evaluation, we selected the 12 apps with the largest user base [1], [25]. Many QR code reader do not only decode QR codes, but offer additional features. At first we evaluated the functional range of the applications and summarized the main features in Table I. Then we produced a sample QR code with a link to a website. Additionally, we modified this QR code in a way that the encoded link minimally deviates from the original link, simulating a phishing attack. However, we switched only a small number of modules so that the human eye would not be able to detect the deviation from the original QR code. Then we scanned the codes and observed if the decoder application was able to detect the potential phishing attack and if they verified the trustworthiness of the decoded content. Additionally, we reviewed the APIs published by the application vendors, if available. Finally, we used an HTTP and HTTPS interception proxy to observe the communication between the mobile application and the external web services to determine if sensitive information was leaked through the reader application. In the following, we describe the investigated features in detail.

Sample Description and Features

Table I provides an overview of the selected applications, their additional features and estimated user base. The additional features we identified within our application sample are as follows: *(1) Synchronization:* indicates that the application synchronizes the scan history across the Web and all registered devices. This feature may have additional benefits for the user, but also yields distinct privacy and security challenges. *(2) Price Comparison:* After scanning a QR code, the decoded information on a product or service is used for an online price comparison. The application then displays the current best price of the

Table I
FEATURES AND BUSINESS DATA OF THE SAMPLE

	Additional Features			Usage		
App Name	Cloud Sync	Price Compare	Local Shop Finder	Price	Est. User Base	Platform
Scan	✓			$1.99	50-100M	Android/iOS/Win
Barcode Scanner				free	50-100M	Android
RedLaser		✓	✓	free	50-80M	Android/iOS/Win
Bakodo		✓		free	30-55M	iOS
QR Droid				free	10-50M	Android
Quick Scan		✓		free	20-40M	iOS
ShopSavvy		✓	✓	free	15-38M	Android/iOS/Win
QR Code Reader and Scanner				free	10-20M	iOS/Win
Qrafter		✓		free	8-16M	iOS
ScanLife		✓	✓	free	5-12M	Android/iOS/Win
i-nigma				free	5-10M	Android/iOS/Win
AT&T Code Scanner				free	5-10M	Android/iOS

respective product amongst different online shops. This beneficial feature however involves the transmission of sensitive information to a third party and provides this third party with data to track the user. *(3) Local Shop Finder:* In addition to a price comparison, the application finds shops near the user's current and/or requested location that offer the scanned product. This additional feature also requires location information in order to function and enables the third party to track the user.

Security Protection Mechanisms

We evaluated the QR codes in reader applications with respect to basic security enhancements to protect the user against malicious QR codes.The results of our evaluation are summarized in Table II. In the following section, we describe these security features in detail.

(1) Modification Detection: We modified a QR code with a link encoded to the website of the credit card company *American Express* (https://www.americanexpress.com) in order to redirect the user to https://www.aoericanexpsers.com (attack proposed in [12]). Just like in a real-world phishing example, the URI is modified in a way to look similar to the original URI. We then scanned both QR codes and examined if the QR code reader application detects the modification and notifies the user regarding a potential fraud.

(2) Website Analysis: We investigated if the respective application uses *Google Safe Browsing* [2] or a similar service. Google Safe Browsing is a service that checks URIs and web content if they for malware or phishing content.

(3) URI Display: Displaying the URI is a powerful feature to notify the user about the actual encoded URI and can prevent the user from becoming a victim of a social engineering attack. As QR codes are not human-readable, users depend on this functionality for awareness of the encoded content.

Privacy Violations

To analyze the leakage of personally identifiable information (PII), we analyzed their network communication. To do so, we used an HTTP(S) interception proxy to monitor information exchanged between mobile reader software and their corresponding web services.

(1) External Communication: QR code reader applications process captured images directly on the device. Nevertheless, many applications transmit the decoded content of the QR codes to the servers of the application vendor.

(2) User Tracking: In general, QR code reader applications do not require personal information and user tracking in order fulfill the user's requirements.

(3) Location Data: QR code reader apps do not require the current location of the user in order to function. However, if they provide additional features as shown in Table I, location information may be required.

Results

In the following we present the results of our security evaluation with respect to our evaluation criteria. Table II shows the results of our evaluation with respect to security enhancements. Fields that are highlighted in red indicate that the (non-) existence of a feature has a negative impact on security, whereas green indicates that the result is beneficial for security. None of the 12 evaluated applications was able to detect a modified QR code. Furthermore, only a single application (QRafter) analyzed decoded URIs with a service that checks URIs and web content for malware and phishing. The majority of the QR code readers within our sample provided a feature to display the decoded URI. The software with the biggest user base in our sample did however not provide this security enhancing mechanism. Some applications provide an option to disable this functionality. Table III outlines our results with respect to privacy violations. 10 out of 12 applications transmit the decoded QR code content to remote servers. While a number of applications require external communication with an associated web service to provide additional features, four applications (QR Droid, ShopSavy, i-nigma and AT&T Code Scanner) use this external communication to leak user information even though the application does not require it. All evaluated apps do not need any login or any other form of authentication. However, most applications within our sample that transmit the decoded content to external servers generate a unique user identifier to track the user's activity. ShopSavvy for example regularly transmits user information together with the decoded content[1] of the QR

[1] https://api.shopsavvy.com/5/cloud/scans/qrCode

Table II
SECURITY FEATURES PROVIDED BY QR CODE READERS

App	Modification Detection	Website Analysis	URI Display
Scan	No	No	No
Barcode Scanner	No	No	Yes
RedLaser	No	No	Yes
Bakodo	No	No	Yes
QR Droid	No	No	Yes
Quick Scan	No	No	Yes
ShopSavvy	No	No	No
QR Code Reader	No	No	No
Qrafter	No	Yes	Yes
ScanLife	No	No	No
i-nigma	No	No	Yes
AT&T Code Scanner	No	No	No

Table III
PRIVACY VIOLATIONS OF QR CODE READERS

App	External Communication	User Tracking	Location Data
Scan	Yes	Yes	No
Barcode Scanner	No	No	No
RedLaser	Yes	Yes	Yes
Bakodo	Yes	Yes	Yes
QR Droid	Yes	Yes	No
Quick Scan	Yes	No	No
ShopSavvy	Yes	Yes	Yes
QR Code Reader	No	No	No
Qrafter	Yes	No	No
ScanLife	Yes	Yes	Yes
i-nigma	Yes	Yes	No
AT&T Code Scanner	Yes	Yes	Yes

code (*content*), the user's current location (*country*) and a generated user identifier (*user*). Quick Scan and Qrafter regularly transmit information to external servers but do not generate a user ID to track the user. Over one third of the tested apps required permission to access the device's location and transmitted location information to a third party, even though some of them do not provide any of the listed additional functionalities. Concerning privacy, only 2 out of 12 surveyed applications (Barcode Scanner, QR Code Reader) were, according to our criteria, sufficiently protecting the user's privacy. The AT&T Code Scanner significantly violates its user's privacy as it scored worst in our privacy as it regularly collects and transmit privacy sensitive information and requires permission to location data. However, as shown in Table I, it does not provide additional functionality that would justify this behavior.

IV. INTERCULTURAL USER STUDY

In order to determine if there are any significant differences in security awareness, we conducted an intercultural comparative user study. In this section, we outline the construction of our study and present our results.

Design and Recruitment

We deployed an online questionnaire through QR codes in four different European countries, namely Austria, Finland, France and Greece. The QR code stickers were placed in highly frequented urban spaces, such as bus stops, public toilets or universities. We did not replace existing QR code stickers. We deployed three different types of QR code stickers: plain QR codes with no additional information, QR codes with a description (respectively translated) and QR codes with cute cat images. The QR codes did not only have the link to the survey encoded, but also a city parameter, a location parameter and a unique ID to measure the performance of each sticker. Our goal was to simulate a QR code phishing attack. In this way, we were able to recruit participants that scan unverified QR codes in public spaces. As our study was conducted in an unobserved manner, it is limited to people who decided to scan the codes. We are therefore unable to determine how many people noticed our codes, but decided not to scan them. However, this selection bias is often exploited within a real world attack scenario as presented in [11]. In our study, we focus on this group of QR code users as they are the most vulnerable group. In our experiment, the participants came across the QR code just like it would be the case in a real-world scenario. Especially with respect to the QR code stickers without further information, the participants did not know about the encoded content before actually decoding it. After scanning a QR code, the participants were redirected to our website with a seven questions-long questionnaire. The survey was designed in a way that it was easy to handle on a smartphone and could be answered within a few minutes. We translated the survey questions to the languages spoken in the respective countries. The study was conducted simultaneously in four cities. We deployed 784 stickers in total, 113 (14,4%) were utilized by participants at least once (273 hits). The data was collected within two months and 83 participants completed our online survey. As all stickers were placed in locations directly in the city center, it is mostly limited to urban dwellers and excludes participants from rural areas. As one third of the stickers was deployed near university campuses, there is a considerable population bias. We measured the performance of our stickers and evaluated how many people scanned our codes but did not fill out the survey, but did not estimate the fraction of people who saw our stickers but decided not to scan them. Simulating a social engineering attack is an ethically sensitive area. Therefore, we decided not collect any individual-related information except for age and gender. Furthermore, we provided the participants with an opt-out option. For ethical reasons we did not make any false pretenses or impersonate someone else. We also refused to replace already existing QR codes to avoid financial damage or reputation loss of the advertising companies. We followed the guidelines from [19].

Results

In all four cities, the stickers with plain QR codes and no additional information performed best, followed by those with a description. In all four cities, the stickers deployed around universities performed better than stickers near public transportation and toilets. According to the survey results, French and Greek participants scan QR codes more often than Finns and Austrians (Figure 1). French participants scan QR codes significantly more often than Finns (Independent two-sample t-test, probability of error $p = 0,043$) and Austrians (Independent two-sample t-test, probability of error $p = 0,03$). The performance measures of our stickers support this statement.

The reasons for scanning the QR codes were also similar amongst the surveyed cultural groups. Most participants scanned the QR codes out of curiosity or boredom (similar to [24] and [20]), regardless of a specific location where it was placed. In Greece, all the participants reported curiosity as the main motivation to scan the codes.

Concerning risk perception, the indicated answers varied among the cultural groups as shown in Figure 2. In Paris, more than 70% of the participants perceived our QR codes as fishy or reported to be sceptical towards QR codes. In comparison, less than 30% of the participants from Athens perceived our stickers as risky. This is effect is statistically significant within our sample (Independent two-sample t-test, probability of error $p = 0,05$).

About half of the French participants stated that they generally suspected a potential threat.

More than 50% of the Austrian, Finnish and Greek participants reported to check the encoded URL before actually visiting it. However, almost the same fraction of our sample used QR code readers which do not provide this option.

Our sample shows a significant gender imbalance between the four European cities. 88% of the entire sample reported being male. Figure 3 illustrates the gender distribution. 55% of all the participants reported being between 18 and 24 years old. Around 25% were between 25 and 30 years old. The oldest participant was 45 years old and participated in Vienna. Our results suggest that young men are the largest user group of QR codes. These findings correspond to results published in [17].

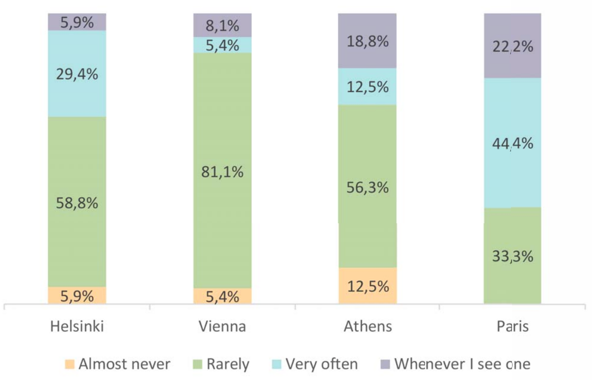

Figure 1. Self-reported QR code scanning frequency in percent. $n_H = 17$, $n_V = 37$, $n_A = 17$, $n_P = 12$

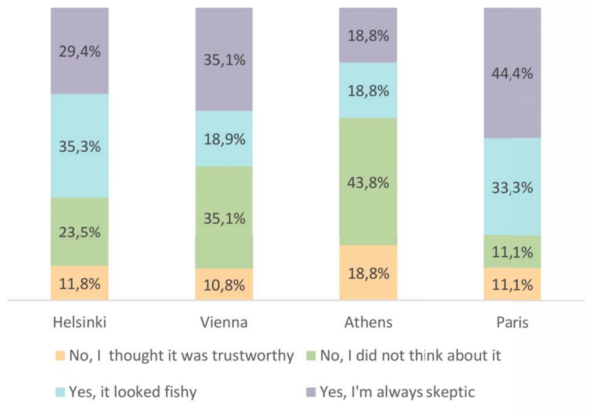

Figure 2. Self-reported considerations on the trustworthiness of our QR code stickers. $n_H = 17$, $n_V = 37$, $n_A = 17$, $n_P = 12$

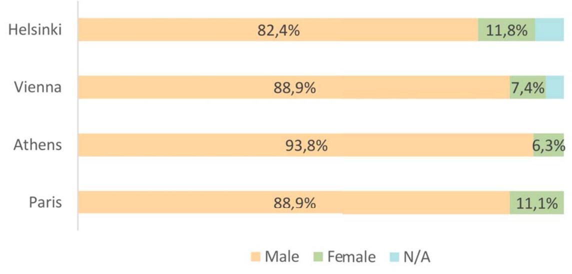

Figure 3. Gender distribution by city in percent.

The most popular operating system within our sample was Android. 70% of the Greek participants accessed our website with a device running Android. The second most popular operating system was iOS. However, almost one quarter of participants from Helsinki were using Windows Phone, which can be explained by the relatively high market share (23%) of Nokia Lumia phones in Finland [22].

V. Design Recommendations

Based on the results from our survey of QR code reader apps, we developed design recommendations to help the user decide whether a QR code is malicious or benign.

A. *QR Code Security Enhancements*

Visual QR Codes. In case of an attack scenario (as described in II) our results suggest that visual QR codes significantly reduce the risk of such an attack. The more complex the theme, the harder it is to modify it in an unobtrusive way. To make it more expensive for an attacker to replace the original QR code (e.g. in billboard advertising), we suggest using a more complex color scheme and embedding it into the color and texture scheme used for the rest of the advertisement.

Digital Signatures. We propose the integration of digital signatures in the QR code standardization to verify the origin of a code check for modifications. The overall idea is to derive a checksum (e.g. SHA-512) and encode it together with the content. The checksum is stored at an online trusted authority. When decoding the QR code, the QR code reader uses the checksum to verify the originator of the code with the trusted authority. A color indication or any other graphical representation should then indicate if verification was successful. To include digital signatures, QR code readers should be adapted to verify and display the result to the user, similar to SSL. For protecting the user against malicious URLs, checking the validity of HTTPS certificates would be an easy-to-implement measure.

B. *Application Enhancements.*

Masking. The distribution of black and white modules of specification-compliant QR codes follows a specific pattern. This pattern is determined by a mask. Due to the error-correcting Reed-Solomon Codes, QR codes with up to a certain amount of corrupted pixels still can be decoded. This threshold is specified by the error correction level. The error correction level is indicated by two modules down in the bottom left-hand corner of every QR code. The higher the deviation from an even distribution of black and white modules, the higher is the probability that the QR code is modified. This is accompanied by a trade-off between error rate and security and is therefore not intended to be a stand-alone countermeasure. In case of a suspicious QR code, this feature can be used as a metric in combination with other measures.

Malicious URL Detection. In general, there are two approaches to successfully distinguish potentially malicious URLs from benign ones. In general, benign URLs have different characteristics than malicious ones. Choi et al. [4] proposed a framework to detect malicious URLs based on text classification. They defined discriminative features to decide whether a URL is benign or malicious. These complex features are as follows: (1)*lexical*, (2)*link popularity*, (3)*webpage content*, (4)*DNS*, (5)*DNS fluxiness* and (6)*network* features. Additionally, we recommend to use frequently updated URL black-/whitelists. Yao et al. [26] proposed the PhishTank API [18] for URL blacklisting.

Google Safe Browsing. [2] provides a framework to detect phishing attacks and malware infected content. Google provides an API that is publicly available.

C. Usability Enhancements

Content Display. We argue that displaying the content to the user is an easy yet effective tool to help users make decisions on whether to proceed with encoded content or not. The results from our QR code reader survey from Section III show that almost 50% of the most downloaded QR code readers already include this feature. However, in many cases this feature is not feasible as a stand-alone measure. QR codes often have shortened URLs encoded. Also, subdomains or longer URLs make it hard for users to approve them and users are often unable to distinguish similar looking malicious URLs from benign ones [7], [8].

Predisplay Analysis. As an additional feature, we recommend to pre-process the encoded content (e.g. resolving shortened URLs) in the background and then display the final URL to the user. For subdomains, we recommend to highlight the second-level domain.

Content Verification. Additionally, we suggest to check the encoded content with the help of black- and whitelists as proposed in [26]. As the results from [8] and [3] suggest, warnings are in many cases not effective to inform the user about threats and their implications. Therefore we recommend to graphically indicate whether the encoded content is trusted or not according to the respective black or whitelist and propose to follow a participatory design approach in designing icons.

Privacy. None of the apps evaluated in this paper informed the user about callbacks and the hidden and unexcpected traffic. We therefore recommend to keep the decoded content locally and to not transmit it to a third party.

VI. Evaluation

To evaluate the design guidelines defined in Section V, we implemented both a QR code encoder and decoder.

To evaluate the digital signature scheme, we implemented a QR code generator that produces QR codes with digital signatures. The QR code generator is a web application that allows user to generate signed QR codes and to verify already existing ones. In addition to an admin console to sign or revoke signatures, it features a responsive user interface to make it accessible on desktop and mobile devices. Our application is implemented in Javascript, MongoDB is used for persistence. For our proof of concept and to ensure backward compatibility, we implemented a basic signature scheme based on base64 encoded signatures/SHA-256 as a fragment of the URL.

Our smartphone app also supports digital signatures but is backward-compatible to standard-compliant QR codes. In addition to the Android API, we used ZXING, a barcode image processing library for Java and dnsjava, a library for sending DNS queries.

We implemented our prototype in a way that it complies with our design recommendations. We ran the same tests as presented in III with the same testbed of QR codes and our prototype successfully passed all tests. Additionally, we created an extended testbed to determine if the strict security mechanisms produce false positives, i.e. detect codes as potentially malicious even if they are benign. False positives may occur, especially when the code to be scanned is not sufficiently illuminated. To minimize this, the threshold for error rate checks was modified. Based on our findings we aimed to optimize the interplay between human factors and the security mechanisms of our application. To make our application both secure and usable, we actively involved potential users in the design process, using user-based usability testing. Furthermore, we based decisions on the user interface design on state of the art findings from scientific literature on security warnings [3], [10], [9]. Based on the scientific literature, we decided to reduce the burden on users to a minimum and designed only two different warnings, i.e. only if a certain number of security checks failed and if there is no Internet connection. We performed a summative, user-based usability test [15] (n=8) to measure the effectiveness of the security warnings from our secure QR code reader application. Our usability study was conducted in two iterations. First, the participants had to scan ten different QR codes in order to solve a word puzzle. Every QR code used in this game had a link to a site encoded that displayed a letter and its position in the word to be found. The goal was to scan enough codes to know the secret word. Three out of the ten codes were malicious and yielded a security warning. For this study, we did not encode real malicious links but modified the application in a way so that it issued warnings when particular codes were scanned just as if they were malicious. We did not tell the participants that the overall aim of this application was security and privacy. We then investigated how many codes were scanned and how many of the URLs were actually visited. After the game, we started the second iteration where we debriefed the participants and then actively engaged them in the design phase. The goal of this participatory approach was to design improved, user-friendly warnings. To do so, we first interviewed the participants on how they liked the app and how they reacted to the warnings. Subsequently, we interviewed them about their suggestions and preferences concerning the design of our user interface and the warnings. Based on this, we encouraged them to modify our warning design using pen and paper. As we did not tell the participants that this was a secure QR code reader application at the beginning of the study, 6 participants scanned the malicious codes more than once, because they did not notice the security warnings as such in the beginning. 3 participants had noticed and read the warning when it was issued for the second time and reacted appropriately. All of the participants checked the displayed URLs and said that this was an important feature, which a QR code reader should have. Two participants stated that they would not want to use an app without this feature as they in general do not trust codes. During the first round, 3 participants immediately tapped "yes" when the warning was displayed and thus opened the malicious URL. In the second phase, all participants claimed that warnings should be issued as little as possible as already implemented in our prototype application. Furthermore, most participants preferred warnings that describe the possible implications from a potentially malicious URL. Most participants said that they observe a habituation effect in their behavior and therefore prefer warnings that deviate from standard warning designs. One participant said that *"a small difference, such as e.g. multiple red exclamation marks or stop signs instead of one would be a sufficient deviation to attract my attention"*. They said they prefer color indicators to highlight security warnings in general. Based on the design suggestions by our participants, we designed the initial scan result display in a more striking way, using three red exclamation marks to attract the user's attention. Furthermore, we rephrased the warning message including a possible consequence to make the implications more understandable to the user.

VII. Discussion

In the course of this evaluation, we found that security enhancing features are very rarely provided by QR code reader applications. Considering the fact that malicious QR codes are a potential threat [23] it is surprising that most QR code reader applications ignore this danger. Services, such as Google Safe Browsing, anti-phishing or malicious URI detection tools (e.g. [16]) provide effective security measure. Even though these tools are mostly free and easy to use, 11 out of 12 applications do not use them. Due to the fact that QR codes are not human-readable, the user relies on effective tools to identify whether the QR code is malicious or benign. Therefore, we consider URI display as essential to support the user's decision making process on whether to visit the encoded link or not. Therefore, we consider applications that do not display the URI as containing a major security weakness. Our results show that many applications leak user-related information. However, some of them require the transmission of this information to external web services in order to provide full functionality. Nevertheless, 4 out of 12 applications leaked information without any functional benefit for the user. In general, the results of our intercultural user study suggest that Finnish QR code users have the highest level of security awareness and successfully apply their knowledge in a real world scenario. When comparing our results to those published by Seeburger et al. [20] and Vidas et al. [24] the overall attitude towards QR codes as a medium to access Web content is comparable for European, American and Australian smartphone users. However, the awareness of security-related challenges varies, even among European nations. Due to the different perceptions of privacy in non-western countries, we plan to expand our survey to not only a more in-depth analysis of the European data, but also Asian countries such as Japan and China, where the overall performance of QR codes in public spaces is significantly higher. This is due to the fact that in comparison to one-dimensional barcodes, two-dimensional barcodes have the ability to encode complex Chinese characters. As our results suggest, young men are the most considerable user group of QR codes. This gender imbalance motivates more gender-specific research concerning awareness, technology acceptance and software design. We showed that our design recommendations can, if considered in the architectural design decisions of an application architecture, effectively protect users from QR code-based attacks. In particular, the findings from the usability study were very helpful in designing effective warnings and a user interface that supports the user in the decision-making process on whether to trust or not to trust a certain source of information. Our findings also highlight the importance of user-centric design approaches for secure applications.

VIII. Conclusion

In this paper, we presented an extensive analysis of QR code security from a holistic point of view. We analyzed why users are still susceptible to QR code based attacks and why currently deployed smartphone applications are unable to mitigate these attacks. Additionally, we proposed a set of design guidelines to build secure, usable and privacy-preserving apps. We showed that if security, privacy and usability aspects are considered evenly, smartphone applications can effectively protect users against malicious QR codes.

Acknowledgements

The research was funded by COMET K1, FFG - Austrian Research Promotion Agency, the Austrian Science Fund (FWF): P26289-N23 and the Josef Ressel Center for User-Friendly Secure Mobile Environments (Usmile).

REFERENCES

[1] Google Play. Available Online: https://play.google.com/store, last accessed 02/09/2014.

[2] Google Safe Browsing API. Available Online: https://developers.google.com/safe-browsing/, last accessed 02/09/2014.

[3] D. Akhawe and A. P. Felt. Alice in Warningland: A Large-scale Field Study of Browser Security Warning Effectiveness. In *Proceedings of the 22Nd USENIX Conference on Security (SEC 2013)*, pages 257–272, 2013.

[4] H. Choi, B. B. Zhu, and H. Lee. Detecting Malicious Web Links and Identifying Their Attack Types. In *Proceedings of the 2Nd USENIX Conference on Web Application Development (WebApps 2011)*, pages 11–11, Berkeley, CA, USA, 2011. USENIX Association.

[5] A. Dabrowski, K. Krombholz, J. Ullrich, and E. Weippl. Qr inception: Barcode-in-barcode attacks. In *Proceedings of the 4th Annual ACM CCS Workshop on Security and Privacy in Smartphones and Mobile Devices (SPSM 2014)*. ACM, 11 2014.

[6] Denso Wave. The QR Code Standard. Available Online: http://www.qrcode.com/en/about/standards.html, last accessed 12/02/2014.

[7] J. S. Downs, M. Holbrook, and L. F. Cranor. Behavioral Response to Phishing Risk. In *Proceedings of the Anti-phishing Working Groups 2Nd Annual eCrime Researchers Summit (eCrime 2007)*, pages 37–44, New York, NY, USA, 2007. ACM.

[8] S. Egelman, L. F. Cranor, and J. Hong. You've been warned: An empirical study of the effectiveness of web browser phishing warnings. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, pages 1065–1074, New York, NY, USA, 2008. ACM.

[9] S. Egelman, L. F. Cranor, and J. Hong. You'Ve Been Warned: An Empirical Study of the Effectiveness of Web Browser Phishing Warnings. In *Proceedings of the 2008 SIGCHI Conference on Human Factors in Computing Systems (CHI 2008)*, pages 1065–1074, 2008.

[10] M. Harbach, S. Fahl, P. Yakovleva, and M. Smith. Sorry, i don't get it: An analysis of warning message texts. In *Financial Cryptography and Data Security*, pages 94–111. Springer, 2013.

[11] C. Herley. Why do nigerian scammers say they are from nigeria? In *WEIS*, 2012.

[12] P. Kieseberg, M. Leithner, M. Mulazzani, L. Munroe, S. Schrittwieser, M. Sinha, and E. Weippl. Qr code security. In *Proceedings of the 8th International Conference on Advances in Mobile Computing and Multimedia*, pages 430–435. ACM, 2010.

[13] K. Krombholz, P. Frühwirt, P. Kieseberg, I. Kapsalis, M. Huber, and E. Weippl. Qr code security: A survey of attacks and challenges for usable security. In *Human Aspects of Information Security, Privacy, and Trust*, pages 79–90. Springer, 2014.

[14] M. Labs. Android Malware Spreads Through QR Code. Available Online: http://blogs.mcafee.com/mcafee-labs/android-malware-spreads-through-qr-code, last accessed 02/09/2014.

[15] J. Lazar, J. H. Feng, and H. Hochheiser. *Research methods in human-computer interaction*. John Wiley & Sons, 2010.

[16] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker. Learning to detect malicious urls. *ACM Transactions on Intelligent Systems and Technology (TIST 2011)*, 2(3):30, 2011.

[17] MarketingCharts. QR Code Scanners Likely Male, Young. Available Online: http://www.marketingcharts.com/direct/qr-code-scanners-likely-male-young-21019/, last accessed 12/02/2014.

[18] OpenDNS. PhishTank API. Available Online: http://www.phishtank.com/api_info.php, last accessed 01/20/2014.

[19] S. Schrittwieser, M. Mulazzani, and E. Weippl. Ethics in security research - which lines should not be crossed? In *Proceedings of the 2013 Cyber-Security Research Ethics Dialog & Strategy Workshop (CREDS 2013)*, 2013.

[20] J. Seeburger. No cure for curiosity: linking physical and digital urban layers. In *Proceedings of the 7th Nordic Conference on Human-Computer Interaction: Making Sense Through Design (NordiCHI 2012)*, pages 247–256. ACM, 2012.

[21] V. Sharma. A Study of Malicious QR Codes. *International Journal of Computational Intelligence and Information Security*, 3(5), 2012.

[22] Surur. Nokia Lumia had around 232013. Available Online: http://wmpoweruser.com/nokia-lumia-had-around-23-smartphone-market-share-in-finland-in-q1-2013/, last accessed 12/02/2014.

[23] theregister.co.uk. That square QR barcode on the poster? Check it's not a sticker. Available Online: http://www.theregister.co.uk/2012/12/10/qr_code_sticker_scam/, last accessed 11/31/2013.

[24] T. Vidas, E. Owusu, S. Wang, C. Zeng, L. F. Cranor, and N. Christin. QRishing: The Susceptibility of Smartphone Users to QR Code Phishing Attacks. In *Proceedings of the 2013 Workshop on Usable Security (USEC 2013)*, 2013.

[25] Xyologic. XYO - Apps to the people. Available Online: http://xyo.net, last accessed 12/09/2013.

[26] H. Yao and D. Shin. Towards Preventing QR Code Based Attacks on Android Phone Using Security Warnings. In *Proceedings of the 8th ACM SIGSAC Symposium on Information, Computer and Communications Security (ASIA CCS'13)*, pages 341–346, 2013.

APPENDIX

List of evaluated QR code reader applications:

- **Scan:** https://scan.me
- **Barcode Scanner:** https://code.google.com/p/zxing/
- **RedLaser**: http://redlaser.com
- **Bakodo:** http://bako.do
- **QR Droid:** http://qrdroid.com
- **Quick Scan:** http://www.ihandysoft.com/apps
- **ShopSavvy:** http://shopsavvy.com
- **QR Code Reader:** https://scan.me
- **Qrafter:** http://keremerkan.net/downloads/
- **ScanLife:** http://www.scanlife.com
- **i-nigma:** http://www.i-nigma.com
- **AT&T Code Scanner:** https://www.wireless.att.com/businesscenter/solutions/mobile-marketing/products/mobile-barcode-service.jsp