E-voting Authentication with QR-codes

Stefanie Falkner[1,2,⋆], Peter Kieseberg[1], Dimitris E. Simos[1], Christina Traxler[1,2], and Edgar Weippl[1]

[1] SBA Research, Favoritenstraße 16, 1040 Vienna, Austria
{sfalkner,pkieseberg,dsimos,ctraxler,eweippl}@sba-research.org
[2] University of Applied Sciences Upper Austria, School of Informatics, Communications and Media, Softwarepark 11, 4232 Hagenberg, Austria
{stefanie.falkner,christina.traxler}@students.fh-hagenberg.at

Abstract. In this paper we propose an e-voting authentication scheme combined with QR-codes and visual cryptography. We focus on the usability, in order to supply voters with less technical experience with a usable scheme. The only requirement is that the user needs to handle a device containing a QR-code reader, most probably a smartphone. This approach is based on visual cryptography as the work horse: The e-voting passwords for authentication are encoded as QR-codes and later encrypted into shadow transparencies. Thus, the transparency by itself conveys no information but when the layers are combined, the secret password is revealed.

Keywords: QR-code, e-voting, usability, visual cryptography, visual secret sharing.

1 Introduction

During the last years a lot of different methods and protocols for e-voting have been proposed, most of them relying on the exchange and/or verification of numbers. Still, for practical use, especially considering older citizens or even people without a good technological background in Information Technology (IT), these schemes are rather impractical.

For example, in a traditional Austrian election the voter has to authenticate himself at the polling place to be allowed to cast a vote. After the polling places have closed, the votes are counted by the poll workers. This process usually takes a considerable amount of time and therefore electronic voting schemes have started to become more and more popular. Especially in online voting systems the authentication is an important factor. In this paper we focus on authentication methods for e-voting which can be combined with QR-codes. To log in at the voting platform, there is no technical background needed, only a QR-code reader i.e. on a smartphone or tablet. We chose QR-codes for encoding the passwords necessary for authentication. The codes are designed for a very

⋆ Authors are listed in alphabetical order.

T. Tryfonas and I. Askoxylakis (Eds.): HAS 2014, LNCS 8533, pp. 149–159, 2014.

robust scanning process, incorporating all kinds of methods for detecting and correcting scanning errors, including partially damaged codes, distortions and rotations. The cryptographic primitives used for our scheme come from the field of visual cryptography. The reason for using the latter encryption methods is to gain a high security level while the utilization is still being easily operated. During the encryption process, two shares, which look like random noise and contain no decipherable information, are generated. By overlaying the shares the secret image gets visible to the human eye.

The paper is structured as follows. Section 2 gives a short summary on the background of QR-codes, including the foundation, areas of usage and how they are structured. It also explains visual cryptography as a mechanism for increasing security. The proposed approach for an innovative, robust, secure and above all user-friendly e-voting authentication scheme is explained in detail in Section 3. In Section 4, we evaluate the approach with respect to the usability of our scheme and to the security aspects. Subsequently, in Section 5 we comparing our approach to related works and we conclude the paper in Section 6 by also giving a perspective on further research.

2 Background

2.1 QR-codes

The QR-code is defined as a two-dimensional barcode invented by one of the Japanese Toyota group companies in 1994. The codes have the same function as the traditional barcodes but there is the possibility to store much more data on it. The standard defines 40 versions (sizes) with different capacities. This standard ISO/IEC 18004, an international standard concentrated on QR-codes, has been published in June 2000 [1]. Originally QR-codes were made-up for the application in the production control of automotive parts but since their development they found a wide range of usage in a lot of different areas. In general QR-codes find a popular utilization on advertising, posters and products. In 2010 the International Air Transport Association (IATA) for airports worldwide introduced them for passenger boarding passes. They are also applied in hospitals, i.e. in Hong Kong for patient identification. Some more examples would be QR-codes on bills for e-payment or ticketing systems for trains and airlines. We present below in Figure 1 the structure of a QR-code.

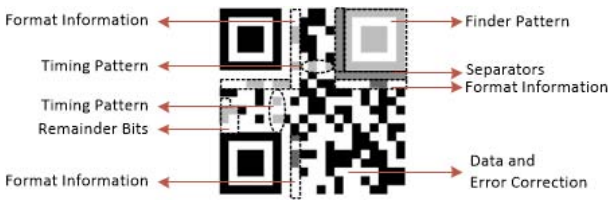

Fig. 1. Structure of version 1 QR-code

A QR-code is divided into modules and each of them is a collection of pixels. For example, version 1 is made of 21x21 modules. In every QR-code there must be three Finder Patterns, which are located in the upper left, upper right and lower left corner. They are used for position detection and identification of possible QR-codes. The Alignment Patterns only occur from version two up to 40, whereby the higher the level is, the more Alignment Patterns exist. For the scanning process a quiet zone, which is defined as a white border, surrounding the QR-code is necessary. This zone is defined in [1] to be at least 4 pixels wide. On a dark background, QR-codes without a quiet zone may be unscannable, because the reader can not distinguish between the dark background and the Finder Patterns. Furthermore, the timing pattern for determining the module coordinates, the Separators for separating the finder patterns from the rest of the code, the data area, the error correction region, and areas which contain the format information as well as the remaining bits are components of a QR-code [1]. The standard defines four different error correction levels (L = 7%, M = 15%, Q = 25%, H = 30%). The error correction is used for recovering the QR-code if parts of the symbol are unreadable or destroyed.

Furthermore, there are also different modes available for encoding the data in QR-codes, depending on which data should be encoded, ranging from plain numbers (numeric mode), simple content like URLs (alphanumeric mode) up to Japanese Kanjis (Kanji mode), amongst others. In particular the 8-bit byte mode, also called binary mode, handles all the character values from $(00)_{HEX}$ to $(FF)_{HEX}$, so far this includes the whole Kana and Latin character set. On the contrary, the alphanumeric mode makes use of a set of 45 characters, containing upper letters, numbers and some predefined special characters.

In order to provide a good contrast for reading a QR-code, it is important that black and white modules are well distributed over the whole symbol. This is done by XORing masks onto the encoded data [1]. Note that, there are eight masks defined in the standard, which are responsible for an optimal distribution of white and black pixels.

2.2 Visual Cryptography

Visual cryptography spawned as a branch of secret sharing cryptography, mainly due to the pioneering idea of Naor and Shamir to split a secret image into n images called shares or transparencies and afterwards if (some) of these shares are stacked using the OR-operation, the original image is revealed [9]. In this paper we use an instantiation of the visual secret sharing (VSS) scheme given in [9], and in particular we use the so-called $(2, 2)$ scheme. The parameters for a (k, n)-threshold scheme [9] are defined as follows.

Let n be the total number of shares and $k \leq n$ the number of shares that need to be stacked in order to reveal the original image. Furthermore, let m be the number of subpixels, v the stacked m-vector, α the relative difference and t be a fixed threshold.

Each share is a collection of subpixels and M are the matrices determining the m subpixels, which are needed for generating the shares. The resulting matrix

can be described as a $n \times m$ binary matrix $S = [s_{ij}]$ where $s_{ij} = 1$ if the j-th subpixel in the i-th share is black, and $s_{ij} = 0$ otherwise.

It is important that the stacked shares have a good contrast between the black and the white pixels, because this is needed for a successful scanning process. Therefore the following requirement should be considered; that the parameter m should be as small as possible to gain an optimal resolution. Furthermore, let $w(v)$ be the Hamming weight of a vector v where $v \in \{0,1\}^s$. The Hamming weight $w(v)$ of the m-vector on which the OR operation is executed, is proportional to the grey level of the stacked shares. While the black pixels remain black in the stacked shares, the white ones are represented with grey color.

For a better visibility the contrast α, which is the relative difference between the minimum and maximum Hamming weight $w(v)$ needs to be as large as possible. The minimum is represented as white (see formula 1) and the maximum as black (see formula 2). For example in the $(2, 2)$ scheme we have used the best contrast α is $\frac{1}{2}$.

$$w(v) \leq t - \alpha \cdot m \tag{1}$$

$$w(v) \geq t \tag{2}$$

The parameter t is the threshold $1 \leq t \leq m$ to construct the shade of grey. The difference between the shades of grey is crucial for the visibility of the stacked shares, to enable the scanning process.

For example in the $(2, 2)$ scheme, two matrices M_0 and M_1 are needed to define the color of the m subpixels in order to generate the two shares S_1 and S_2. We use complimentary matrices to share a black pixel and identical matrices to share a white pixel. Stacking the shares we have all the subpixels associated with the black pixel now black while 50 percent of the subpixels associated with the white pixel remain white. Below we give an example containing two possible matrices for generating the shares.

$$\mathrm{M}_0 = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix} \qquad \mathrm{M}_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Following this selection, each pixel of the secret image is selected one by one and the white ones are represented with M_0 and the black ones with M_1 respectively. During this process the columns of the matrices are permuted randomly. From the resulting matrix, each second row is used for creating share number 2 (S_2) and the other rows result in share number 1 (S_1).

For decrypting the secret image in the stacking process, the matrices are stacked using OR-operations, thus revealing the original image.

3 QR-code Based E-voting

The initialization process As can be seen in Figure 2, the first step in our approach is to generate n passwords for the n voters, which are needed for the

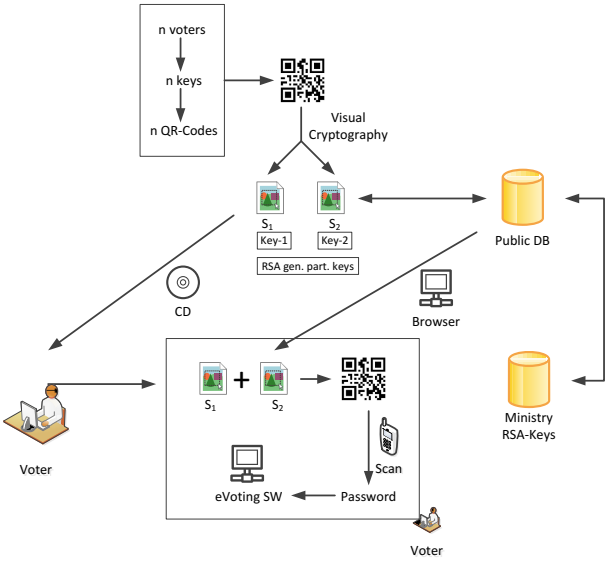

Fig. 2. General structure of the approach

authentication on the voting platform. To create secure passwords we use a pseudorandom number generator (PRNG) [7] with the social security number as unique seed.

Afterwards the n passwords are encoded into n QR-codes. Regarding the QR-code specification, we make use of version 1 and the whole capacity to avoid padding. If there is free space, the QR-code adds fixed binary patterns as padding. Therefore, if voters have the same password length and the same mask, the padding area in the QR-code would appear identical and could therefore be easily spotted, maybe leading to a security breach. For that reason we generate a QR-code without the white border, which is also called the *quiet zone*. Instead of the white border, we use a light uncolored background for the voting platform to further support an unproblematic scanning process.

With regard to the reconstruction of distorted and destructed parts of the code Figure 3 shows the capacity of the binary mode regarding to the ECC (error-correction) levels [1]. For example, the two best combinations for the binary mode would be the ECC Level Q or H and the reasons therefore are explained below.

Version	Modules	ECC Level	Binary
1	21 x 21	L	17
		M	14
		Q	11
		H	7

Fig. 3. Maximum capacity of the QR-code

The main reason for using these combinations is the user friendly password length, while still allowing for a high error correction. There are some advantages especially when using the binary mode. In contrast to the alphanumeric mode, which does not support lower case letters, the required signs for generating a secure password like lower and upper case letters, numbers and special characters are all available in the binary mode. In the next step, with the aid of the $(2, 2)$ VSS scheme, two shares S_1 and S_2 are generated (see Section 2.2), so that either layer by itself conveys no information, but when the layers are combined, the secret image is revealed. An example of the two shares can be seen below in Figure 4.

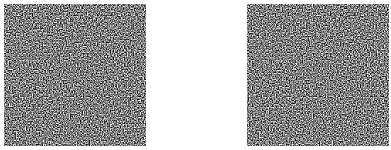

Fig. 4. Share 1 and share 2

To make the shares assignable there a PKI (public key infrastructure) [7] with a unique RSA key-pair (K_1, K_2) for each voter is needed. A PKI is an infrastructure that allows to verify, which public key belongs to whom. In our approach, K_1 serves as the private key while K_2 is the public key. The public keys have to be certified by a trusted third party. Such an infrastructure for e-government purposes is already provided in some countries, like the Federal Ministry of Labour, Social Affairs and Consumer Protection in Austria.

The Communication and Storing Process. After the setup, the share S_2 and the public key K_2 of each voter are stored in the database, while the share/transparency S_1 and the private key K_1 are stored on a CD and sent separately via post to eligible users to enable them to participate in the election. The letter is sent as an official mail, that means that only the addressed voter is allowed to receive the share S_1 and the key K_1. They are sent separately in order to prevent efficient interception. If the key or the share gets lost while sending them through the post, the voter can report this to the appropriate authorities and get them resent.

Authentication Process. During the election, the voter has to input the username and the private key K_1 from the CD on the voting platform. While the user uploads his share, the system's function is to fetch the correct share S_2 from the database and then stack the shares to reveal the secret QR-code (as this can be seen in Figure 5).

Now the user can scan the QR-code with any QR-code reader to reveal the secret password. Afterwards the user enters the correct password, he is authenticated and finally is allowed to vote.

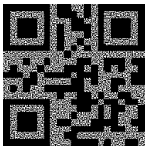

Fig. 5. Stacked shares

4 Evaluation

In this section we evaluate the usability of our scheme, as well as security related considerations.

4.1 Usability

To achieve as many advantages as possible for voters, our approach emphasizes on the usability. In comparison to scheme [10] which concentrates on stacking the printed shares manually by putting one share on a transparent sheet in front of the screen, our scheme offers an automatic stacking process on the system for revealing the secret image. The procedure of a simultaneously perfect aligning and scanning of the stacked shares is difficult and normally many attempts are necessary. In this point our approach reaches a very good performance in our test case. This test case mainly includes the part of the voter and his interaction with the voting platform, i.e. the duration and effort during the scanning part. Due to the automatic stacking and aligning of the shares by the voting system the voter can save a lot of time, because the stacked image gets recognized in a few seconds by the reader. Moreover, even in the case of distortions or destructed parts, QR-codes can be easily scanned due to the high error correction. Through the use of our scheme people with physical limitations or diseases like tremor also have the possibility to vote without the help of others. Another advantage is that the voter needs no background information about visual cryptography and the structure of QR-codes, thus the proposed scheme is accessible to a broad audience. The only required knowledge the voter needs to have is how to use a QR-code scanner on a smartphone or tablet. Regarding to the QR-code scanning, the voter can decide between all available readers, no specific scanner is needed in this approach.

4.2 Attacker Model

In this section some possible attacks are shown and the impacts on the authentication scheme are explained. This includes a description of all parts and entities that are assumed as trusted parties.

Trusted parties. A party is defined as *trusted* when all included parts and members in this party work trustworthy and without any malicious intents. Figure 6

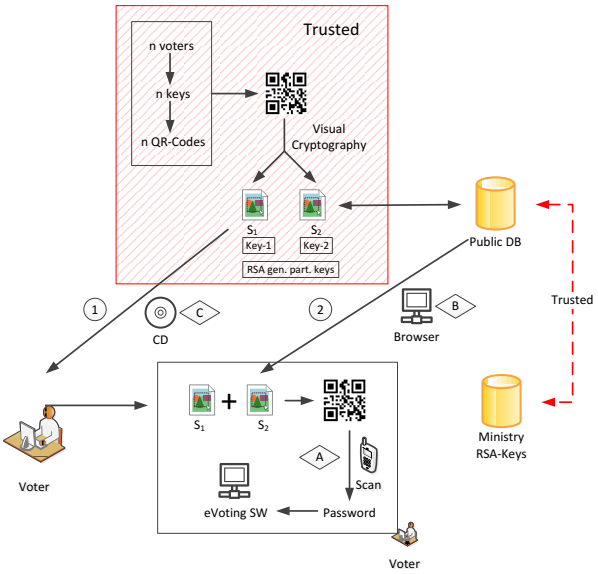

Fig. 6. Attacker model

outlines the two trusted parties: The first one includes the steps during the initialization process in the approach (see Section 3), especially during the initial password generation part, an important assumption is that the passwords and the PRNG-files are not manipulated by the authorities. The same applies for the share and for the RSA-key generation as well as for the trusted third party who certifies them. The second trusted party contains the assignment between the public keys and the shares in the database. The main reason why we assume that this part should be trusted, is to avoid manipulations in the allocation procedure. When this is not the case there is a possibility that the voter receives a wrong share and is not able to reveal the correct password and therefore the authentication process will fail. Furthermore the link between the two trusted parties must also meet the assumptions mentioned above. All this preconditions are perfectly reasonable in an environment where the state is considered trustworthy.

Attack vectors. There exist two major attack vectors against this scheme: (i) Intercepting the shares sent by the post, (ii) Manipulating the shares sent by the browser and (iii) Manipulating the scanning software and sending the password to the attacker. Manipulations and interference on other channels, like e.g. the network layer or a malicious database administrator, are out of the scope of this paper, since these vectors are not specific to our approach and need to be solved by any e-voting solution. Actually, these attack vectors can be translated into two different attacker models: An attacker trying to infiltrate the distribution process of the shares, and an attacker directly targeting the voter.

Resulting Security Requirements. In case of interception (attack vectors (i) and (ii)), our schema is secure in case only one share was derived by the attacker, i.e. only one of the suppliers in the two factor authentication is corrupted. Thus, it is absolutely necessary that the distribution of the two shares is done by completely independent means, as outlined in the approach and that there is no organizational overlap where a single attacker could intercept both shares. Especially the generation process needs to be trusted, as outlined earlier in this Section. As for the attacker that is directly targeting the voter, this could be achieved by either distributing a malicious scanner, or finding security holes in existing, popular scanners and using them through exploits. The intercepted password could then be sent to the attacker. Still, since the hardware the scanner is running on is under the control of the voter, this attack can be avoided: First, our approach does not require the voter to use any kind of special or predefined scanning software, any QR-code scanner can be used, which would require the attacker to either break a large amount of different scanners, or to persuade the voter to use his own. Furthermore, any application wanting to send data needs the respective rights on current smartphone systems. Thus, the users need to be educated by the voting agency, in order not to use any scanning software that requires any network access. Furthermore, this attack vector could be mitigated by supplying a scanner that is under the control of the (trusted) voting agency that does not require any transmission rights. Without network access, the only thing a malicious scanner could do is corrupting the result of the scan. Since our approach supports any scanner and thus does not rely on a specific software, this can be easily mitigated.

Security Limitations. The main security limitation of this approach, aside the need for trusted parties, lies in an attacker who is able to intercept both shares S_1 and S_2. Still, since the underlying VSS is very flexible, the schema could easily be extended to a k-factor authentication scheme, where the only limitation for k lies in the practical usability. Another share could e.g. via MMS from the telecom provider, or several, different, shares could be sent per postal service.

5 Related Work

Authentication, anonymity, confidentiality and non-repudiation are four of the main principles in e-voting schemes. In our scheme we concentrate on authentication because it is an essential part in a voting procedure. We refer to [10], where an authentication scheme for remote voting with visual cryptography is proposed: The voter receives one share from the election office and the second one is shown on the screen. Thus, he has to put the printed transparency in front of the screen image and align it to reveal the secret image. To generate the shares, a unique symmetric key and n random symmetric keys for n voters are needed. With the aid of a RNG (random number generator) the shares are generated. The used seeds are the results of the encryption of n random symmetric keys with the unique key. Our proposed scheme explained in Section 3 differs

from the remote voting scheme in some aspects. First of all, we make use of the visual cryptography method given in [9], therefore we do not need predefined keys for generating the shares. In contrast to our scheme, there is no mapping between the voters and the transparencies ([10]). In our scheme, the shares are generated explicitly and one is stored in a database for each voter. With the aid of a PKI we make the shares assignable. If the shares would not be assignable, a man in the middle could fetch the voting packages including the transparencies and vote an arbitrary number of times. In the latter case, the votes are not retractable and hence the voting result can be distorted.

The main purpose of [2] was to introduce an online authentication scheme with visual cryptography in online money transaction systems. In this approach, the users get a special hardware device from their bank with a numbered set of transparencies stored on it. To transact money, the second associated transparency is shown on the screen. By scanning the transparency with the special device the transparencies are stacked and the secret TAN for authentication is revealed. In Germany a similar authentication scheme is already implemented in reality.

Regarding Australia, there also exists a voting scheme using QR-codes [3], which should be ready in November 2014 for the next Victorian State election. During the voting process, the voter has to scan the code to reveal the candidates names.

Finally, in [6] an e-voting scheme combined with secret sharing is proposed. The voting ballot is encrypted and afterwards the private key is divided into n shares. To enable the decryption of the vote, n authorities have to stack their shares to reveal the private key.

6 Conclusions and Future Work

In this paper, we discuss one of the most important security principles in e-voting schemes and therefore we propose a new efficient authentication scheme for e-voting. It is based on QR-codes and visual cryptography schemes. We mainly concentrate on the usability and consider some security aspects. To consider human factors we use QR-codes, which enables a robust scanning process. In combination with visual cryptography and a public key infrastructure we also address some security issues.

The proposed authentication method for an e-voting scheme, raises also some research questions. For example, there are possibilities to extend the authentication scheme to a complete e-voting scheme, including vote casting and tallying [5]. Therefore, other principles such as receipt-freeness, non-repudiation and confidentiality also can be considered. Moreover, watermarking could be introduced to protect shares from cheating attacks [8]. Using digital watermarking provides double security of image shares. Finally, though we have used the $(2, 2)$-threshold scheme there is also the possibility to use more shares or extended schemes [4], [11] in an appropriate use-case scenario.

Acknowledgements. This work has been supported by the Austrian Research Promotion Agency (FFG) under the Austrian COMET Program. In addition, the work of the third author was carried out during the tenure of an ERCIM "Alain Bensoussan" Fellowship Programme. The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 246016.

References

1. Iso/iec 18004:2000 qr code bar code symbology specification
2. Borchert, B., Reinhardt, K.: Applications of Visual Cryptography. In: Visual Cryptography and secret image sharing, pp. 329–350. CRC Press Taylor & Francis (2012)
3. Burton, C., Culnane, C., Heather, J., Peacock, T., Ryan, P.Y.A., Schneider, S., Srinivasan, S., Teague, V., Wen, R., Xia, Z.: Using pret a voter in victorian state elections. In: Proceedings of the 2012 International Conference on Electronic Voting Technology/Workshop on Trustworthy Elections, EVT/WOTE 2012, p. 1. USENIX Association, Berkeley (2012)
4. Cimato, S., Prsico, R.D., Santis, A.D.: Visual Cryptography for Color Images. In: Visual Cryptography and secret image sharing, pp. 32–56. CRC Press Taylor & Francis (2012)
5. Cramer, R., Franklin, M., Schoenmakers, B., Yung, M.: Multi-authority secret-ballot elections with linear work. In: Maurer, U.M. (ed.) EUROCRYPT 1996. LNCS, vol. 1070, pp. 72–83. Springer, Heidelberg (1996)
6. Cramer, R., Gennaro, R., Schoenmakers, B.: A secure and optimally efficient multi-authority election scheme. In: Fumy, W. (ed.) EUROCRYPT 1997. LNCS, vol. 1233, pp. 103–118. Springer, Heidelberg (1997)
7. Ferguson, N., Schneier, B.: Practical Cryptography. Wiley Publishing, Inc. (2003)
8. Jagdeep Verma, V.K.: A visual cryptographic technique to secure image shares. International Journal of Engineering Research and Applications 2, 1121–1125 (2012)
9. Naor, M., Shamir, A.: Visual Cryptography. In: De Santis, A. (ed.) EUROCRYPT 1994. LNCS, vol. 950, pp. 1–12. Springer, Heidelberg (1995)
10. Paul, N., Evans, D., Rubin, A., Wallach, D.: Authentication for remote voting. In: Workshop on Human-Interaction and Security Systems (2003)
11. Shimizu, T., Isami, M., Terada, K., Ohyama, W., Kimura, F.: Color recognition by extended color space method for 64-color 2-d barcode. In: Proceedings of the IAPR Conference on Machine Vision Applications (IAPR MVA 2011), pp. 259–262 (2011)