QRCode DOOR Project: Access Control Application using QR Code Image

Luiz Antonio Pereira Neves
Dept. of Professional and Technological Education (SEPT)
Federal University of Paraná (UFPR)
Curitiba, Brazil
lapneves@gmail.com

Kevin Santos Martins
Dept. of Professional and Technological Education
Federal University of Paraná (UFPR)
Curitiba, Brazil
kevin238@outlook.com

William Ricardo Santos Lima
Dept. of Professional and Technological Education
Federal University of Paraná (UFPR)
Curitiba, Brazil
william.rslima@gmail.com

Gilson Antonio Giraldi
Dept. of Mathematical and Computational Methods
National Laboratory for Scientific Computing (LNCC)
Petropolis, Brazil
gilson.giraldi@gmail.com

***Abstract*—This research proposes a novel technology for access control, creating a smartphone embedded key making use of two types of cryptography to generate a QR Code image, all this combined with a WebCam attached to an electric lock on a door. This paper introduces an architectural model of the access device, the creation of a QR Code image using encrypted users' data, the encryption process, and the elaboration of the QR Code reading device using a microprocessor Raspberry Pi 2.**

Keywords–QR Code Image; Smart Door; Access Control; Internet of Things; Computer Vision.

I. INTRODUCTION

The evolution of society requires new ways of smart access. This can be seen in home offices, apartment complexes, enterprises, schools, and private events, where identity verification and access validation is necessary. The present paper approaches the use of mobile technologies to figure out this typical problem practically and securely.

One of the main challenges in managing physical access is the administration of the access keys, which can be lost, stolen, or even given away. Also, these keys do not ensure rules about access such as scheduling access, specific locations (sub-locations), or limit of capacity. Given these reasons, the creation of smart doors, which can be managed remotely and have a simple and modern activation interface is justified [1].

As technology use rises, the smartphone has been incorporated into everyday life, hence the loss of a physical key is justified [1]. An embedded virtual key allows creating complex access rules, as encryption and one-time access token. A physical key can be used by anyone who has it or finds it, while a smartphone possesses security engines such as fingerprint lock and/or private passwords, restricting its use to only its owner. Another characteristic is that one single Smartphone can be used to access different profiles, depending on the password used to unlock it [2].

As the digital revolution happens, several daily tasks have become virtual, while the physical access method remained the same. It is more convenient to carry a smartphone holding several access keys than carrying a lot of keys as seen in [1]. It is more inexpensive and practical to manage the access virtually than by using a physical device. This device can control access tries and record user identification, date, and time of admission as well as generates statistics over the accesses.

This paper's objective is to propose and develop access management, web-client aligned with the current mobile and virtual technologies, as for example QR Code, which provides security and ease of use.

Section II describes the proposed approach. Section III presents verification and validation of the system proposed. Finally, conclusions are given in Section IV.

II. PROPOSED METHODOLOGY

This research methodology is built up in four steps of development. The first step is to develop the web-client and the prototype of the mobile layout. The second step consists on the creation of the access device, which involves the QR Code capture definition by the WebCam as seen in [3] and the conceiving of how the captured image is validated and decrypted in the backend, allowing the access by the user. Later on, there will be the conceiving of the concept as well as the system's architecture which incorporates both platforms (hardware and software). The last step is the codification of all system modules, with validation and integration tests with a physical access device.

A. Access Control Device Conception

The access control system (Figure 1) is composed by an electric lock attached to a door and to a Webcam that reads the QR Code from the smartphone's screen. These devices work-integrated through a Raspberry Pi 2 microcontroller.

B. QR Code Image Conception

The QR Code is a two-dimensional barcode (Figure 2) which specifications state is used to encode any set of characters mapped by ISO4 8859-1. Widely diffused in mobile technologies, it is versatile and easy to implement. Besides all this, it is free to use.

According to the literature, two-dimensional barcodes are simple and cheap ways to represent commercial data, but it

Copyright (c) IARIA, 2020. ISBN: 978-1-61208-782-5

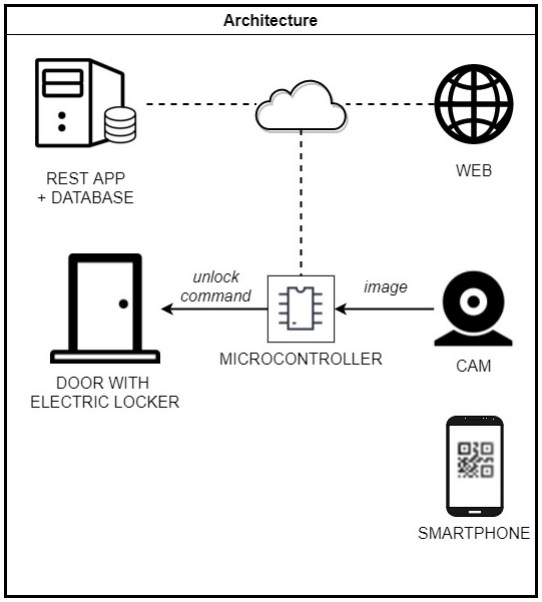

Figure 1. System's architecture proposed.

also improves mobile user experience by reducing the amount of manually input data [4].

The QR Code was developed by Denso Corporation in 1994, and it was acknowledged later as a usable standard. QR Code has been approved as a standard by ISO, JIS e AIM protocols as seen in [5] [6]. This standard has been widely used in a large variety of applications as the manufacture, logistics and sales applications. There are over 40 different QR Code versions, four levels of error correction and its maximum symbol size can encode 7089 numeric data or 4296 alphanumeric data. The QR Code images are captured by the cell phone camera. In most smartphones the images are captured in RGB 24 bit [7], but the QR Code symbol is a set of dark and light pixels, therefore it is needless to deal with color information making its reading and calculation quick [4].

Figure 2. QR Code image.

In this research, the QR Code is used to transmit the access information from the smartphone to the access control system. This information contains the following items: the unique device identification; user's access password; a system generated key; the date and time of the QR Code generation; and a hash [8] MD5 [9] in order to avoid access information corruption. In overview mode, all this information is encrypted using RSA [10] algorithm so that no QR Code reader or other application is able to interpret this information (see Figure 3).

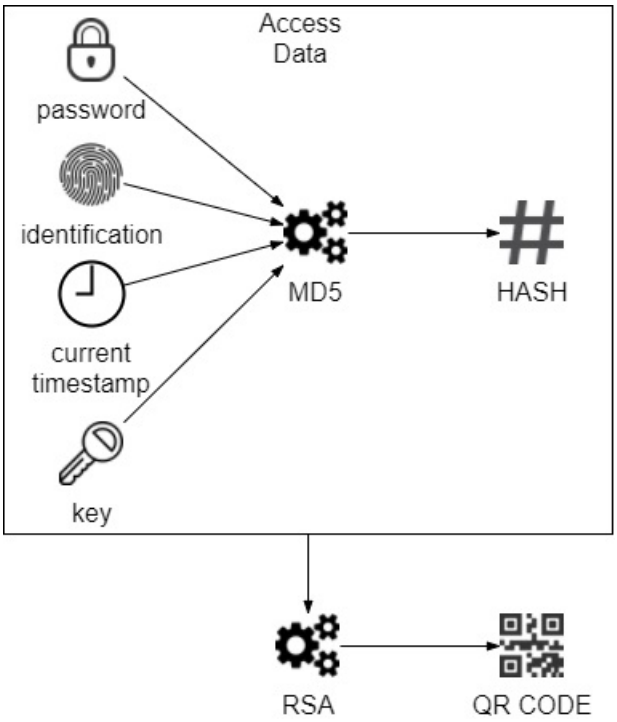

Figure 3. QR Code image generation.

C. Conceiving the Integrated System

The goal of this session is to define what components are necessary for the proper functioning of the systems, as well as the interaction between them. Five artefacts, at least, were identified in the validation and access liberation process: an access key, a sensor, a client, authentication service and a physical access device. The key is generated by the smartphone which generates the QR Code according to the previously mentioned procedure. The sensor is a WebCam positioned at the door disposed at any user's arm's range. At this point, it is important to consider any physical limitation that the user may have, such as: reduced hight, wheelchair or crutches also considering any possible motor disability in order to manage the QR Code reading. The WebCam is connected to the client and it is responsible for the reading of the QR Code on the mobile. The client is responsible for creating and maintaining this connection with the authentication service and door lock activation. On the authentication service module there are all the access rules and answers to the clients resulting in an allowed access or not.

The physical access device can either be an electric door lock, a vault, a ticket gate, a gate, or any other advice that has been already used as long as it allows electronic activation. According to Figure 4, the authentication process is configured in six steps, which are:

1 The application installed on the user's smartphone is responsible for creating and rendering the QR Code on its screen with the access information,

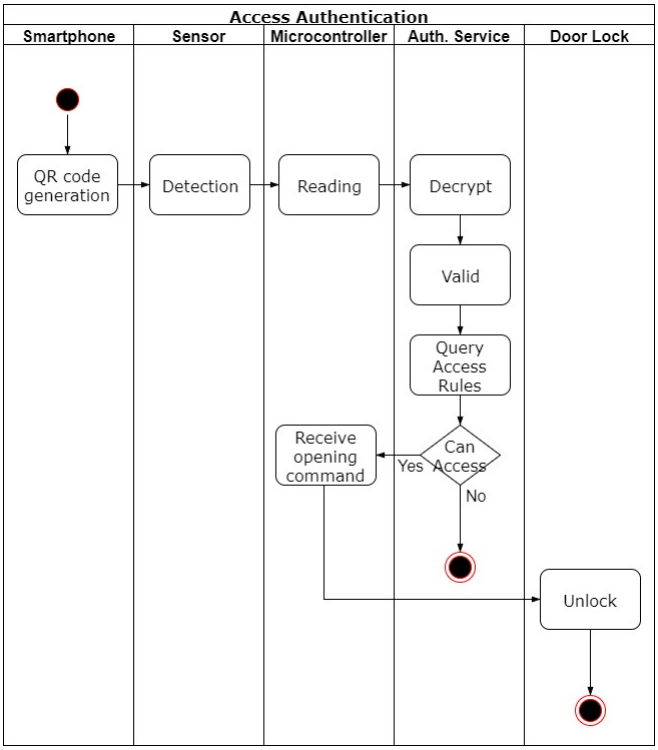

Figure 4. Access authentication flowchart.

2 The user's QR Code is then captured by the camera that transmits the image to the microcontroller.

3 Using image processing techniques on the user's QR Code, the microcontroller identifies the QR Code, reads its containing access information, and sends them to the authentication service.

4 The authentication service decodes the access information, validates the authenticity of the data, and checks within the access rules if the door must be opened or not.

5 In case the service identifies that the door must be opened, it sends the liberation request to the microcontroller attached to the door lock.

6 The microcontroller activates the door lock in order to open it.

III. Verification and Validation

After the conceived system a verification procedure is done in order to answer the question: "Are we building this system correctly?". In software engineering, this question is answered through tests at different levels, and with different techniques. The choice of the procedure was verified and based on its relevance inside the project as a way to soften the risks [11].

Firstly, this procedure verifies if it is viable to generate the QR Code with all the access data in a safe and reliable way. Next, it tests the microcontroller functioning, verifying if it complies with the project requirements, which are: working with a WebCam, QR Code reading, activation of electric devices, and establishing a connection with authentication services. And lastly, it runs the integration tests in order to validate the system functioning in its core parts.

Most importantly, it creates a simple server that implements login authentication and the WebSocket [12]. Then, it is built on a client on the camera's microcontroller. The QR Code is generated. And lastly, the integrated system is tested.

A. QR Code Image Generation

Concerning the verification of the QR Code generation it is used an unit test technique [11] validating the system input and output. We seek to find if it is possible to generate the QR Code inside the predefined security parameters:

1 Containing the basic authentication data, composed by device identification and password;

2 Containing date and time of QR Code generation;

3 Containing hash-based data to validate the information authenticity;

4 Encrypting all the data before the QR Code generation.

The access data encoding process is shown in Figure 5. The test input is an 8 digit identification code and a 10 digit password. This means that with the same pattern, it is possible to generate E+18 different combinations. The date and time of generation have minute precision, which means that it is possible to create a maximum tolerance between the QR Code generation and the authentication of up to one minute. After the input data is encoded, the MD5 algorithm is applied on the generation of a hash. Then, the first two bytes of the hash are concatenated at the end of the previously encoded data. This step is fundamental to the authentication service to verify if the access data hasn't been changed.

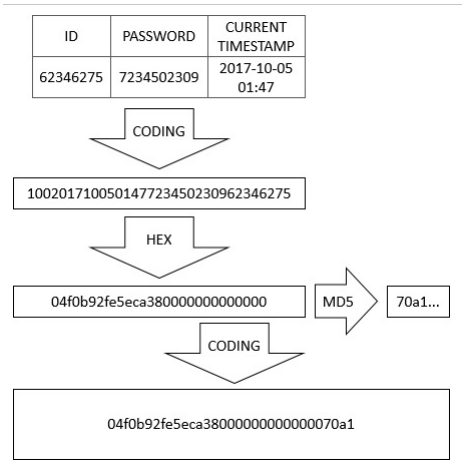

Figure 5. Access authentication flowchart.

Lastly, the data is encrypted with the RSA algorithm and encoded in base64 (Figure 6). The encryption is necessary so that no other QR Code is able to interpret its containing information. The base64 encoding is necessary because the RSA output is a set of bytes and the QR Code is generated from a text, as shown on Figure 7.

Thus, the output is a QR Code image (Figure 7) that can be easily read by the access control device and owns all the needed security characteristics of the project.

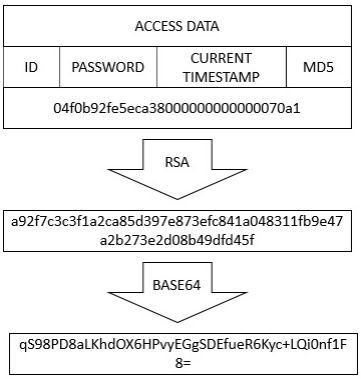

Figure 6. Access data encryption.

Figure 7. QR Code image generation.

B. Hardware to Access Control

The hardware engineering process can be seen in three phases: planning and specification; project and prototyping; production, distribution, and service field [11]. The first two phases are executed in this project.

The hardware requirement analysis is made in order to specify the functionality, interface, and performance requirements, for all the hardware components. The component validated in this step of the project is the access control device.

In prototyping, it is assembled a device to meet the requirements (Figure 8). It is possible to identify the following components:

- CC Power cable, 5 Volts and 2 Amperes with micro USB 2.0 type B connector (see Figure 8-1);
- Ethernet cable with RJ-45 connector, connected to the same network as the authentication service (see Figure 8-2);
- Microcontroller Raspberry Pi 2, with 900MHz processor, 2GB of RAM, e 8GB of ROM (see Figure 8-3);
- LEDs to indicate the system status of the door lock simulation system on a breadboard (see Figure 8-4) and
- Digital camera with 5 Megapixels resolution connected via USB (see Figure 8-5).

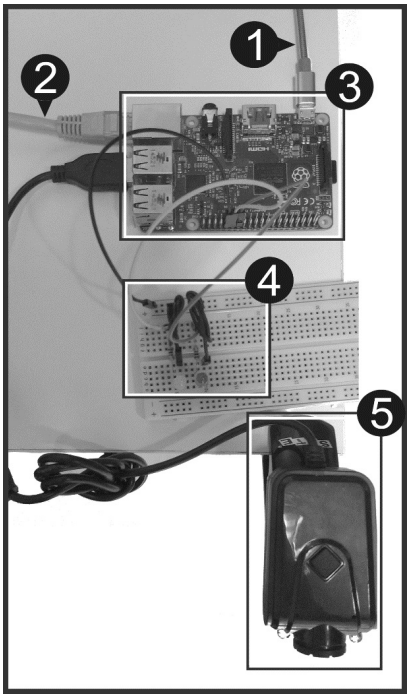

Figure 8. Assembled Device for testing.

After the prototype assembly, the function of each device part is verified through software executions. The camera functioning and QR Code reading competence are checked by Zbar Barcode Reader [13] as seen in Figure 9, it highlights the captured QR Code image inside a light green rectangular area. The configuration and network connectivity with the authentication service is done by a shell Telnet command that verifies not only the network layer but also the application layer.

And finally, a logical port microcontroller verification is done by an embedded programming that connects LEDs on microcontroller ports and executes a written script in Python language [14] using RPi, GPIO library, as seen in Figures 10 and 11. As a result, it is possible to see that the hardware requirements are met by the assembled device.

C. Integration Test

The integration tests aim to verify if the individual system modules work properly when running combined [11]. It also tests if the interfaces that interconnect the components work with an error level below allowed. So, all the components created in the previous sections are used in this test. The QR code generated is rendered on the smartphone screen in front of the device's camera.

The access data is then captured and sent to the server who decodes and validates if the access is allowed and returns an authorization message to the door lock, or in this experiment case, activating the LED. All these tests data are obtained through the logs shown in the screen connected to the microcontroller.

It is possible to see all microcontrollers processing through the logs sent to the client screen in Figure 10. Firstly, the device logins the server sending user data and password. Afterward, it

Figure 9. The camera capture the QR Code image.

```
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)
```

Figure 10. Python script to verify the logic port.

```
GPIO.setup(18, GPIO.OUT)
print("LED on")
GPIO.output(18, GPIO.HIGH)
time.sleep(1)
print("LED off")
GPIO.output(18, GPIO.LOW)
```

Figure 11. Python script python to verify the logic port.

```
qrcdoor-embedded/src $ python3 main.py
Connecting to server by websocket
Creating session
URL: "http://192.168.1.3:8070/login"
PAYLOAD: "{'password': 'user1', 'username'
Sending data to server
Server's response
URL: "http://192.168.1.3:8070/login"
RESPONSE: "<Response [200]>"
Cheking session
URL: "http://192.168.1.3:8070"
RESPONSE: "<Response [200]>"
Session valid
COOKIES: "{'SESSION': '41341f27-4c60-41db-
URL: "ws://192.168.1.3:8070/open/websocket
COOKIE: "SESSION=41341f27-4c60-41db-9019-3
WS connection has been opened
Sending QR code:
QR-Code:Bzhshsjsjjdhhddjdjhdhdjdjxj
A message has been received:
Hello, QR-Code:Bzhshsjsjjdhhddjdjhdhdjdjxj
```

Figure 12. View from Microcontroller logs.

opens a WebSocket connection to exchange messages referring to the door lock (Figure 12). As soon as it detects and extracts the access data from the QR Code, the system sends information to the server. And lastly, after the server decrypt and validates the data, it notifies the client that the door lock must be activated. Finally, it is certified that the component integration has been well succeeded.

Considering the problems encountered during the development of this project, there were many implementation difficulties in programming the communication of the door microcontroller with the server. And this was solved, with the improvement and refinements of the embedded system algorithms, in a stable direction.

This project was established quickly, and it is part of a larger project, entitled Smart Door [15], which has as relevance the proposed web system that manages doors and users, the use of RFID technology to access the door with use of key tags and also the use of Android technology resources to establish a communication to the door. Finally, the combination of all these technologies is not yet available on the marketplace, making out of this project a very relevant one.

IV. Conclusion

As the access to mobile technologies is becoming more and more common, and the regular locks need physical keys (which can be lost or stolen), digital locks offer more safety and simplicity. In this paper, we propose using simpler, cheaper, and consolidated technologies, creating virtual embedded keys in smartphones. Using this gadget that is already inserted in daily life to perform one or more tasks, replacing physical keys. The Raspberry Pi 2 has entry for one single camera, or via USB, making this project's assembly more accessible.

The QRCDoor project consists of the creation of a web client and a mobile application, conceiving the door access device, which involves the definition and capture of the QR Code image by the mobile camera. The elaboration of the system architecture, integrating all proposed platforms were validated. Finally, we have the coding of all system modules, with validation tests, and integrating with the physical access control device. Thus, all goals of this research have been reached.

Controlling access is a present-day problem. The access control available depends on a human direct intervention, where the user is registered in a system and receives a physical card that grants him the access, this card can be easily lost or stolen. The QRCDoor innovates it by making the control digital and embedding the access keys and cards in a smartphone. Making it possible to create complex access rules to places with specific date and time and also allows an easy identification and access key control. Therefore, another relevance for this research is that the proposed system makes daily life more practical for home and professional environments.

In the future, with the new QRCDoor modules, the intention is to make everyday life, even simpler, by also making access to private events digital, eliminating the use of traditional keys and cards.

References

[1] Y. Kao et al., "Physical Access Control Based on QR Code", in Proceedings of International Conference on Cyber-Enabled Distributed Computing and knowledge Discovery, pp. 285-288, Oct. 2011.

Copyright (c) IARIA, 2020. ISBN: 978-1-61208-782-5

[2] M. Ilyas and S. A. Ahson, ”Smartphones”. Intl. Engineering Consortium, IEC Publications, ISBN: 978-1-931695-50-3, 409 pages, 2006.

[3] Y. Pritch et al., ”Webcam synopsis: Peeking around the world.” in Proceedings of IEEE 11th International Conference on Computer Vision 2007, ICCV 2007, pp. 1-8.,IEEE, 2007.

[4] Y. Liu and J. Yang and M. Liu, ”Recognition of QR Code with mobile phones.” in Proceedings of IEEE 2008 Chinese Control and Decision Conference, CCDC 2008, pp. 203-206, IEEE, 2008.

[5] ”The Japanese Industrial Standards Committee (JISC)”, URL:https://www.jisc.go.jp/eng/ [accessed:2018-08-09].

[6] ”OnBarcode”, URL: http://www.onbarcode.com/qr_code/#gascii [accessed: 2017-10-01].

[7] S. Süsstrunk and R. Buckley and S. Swen, ”Standard RGB Color spaces” in Proceedings of Seventh Color Imaging Conference, IS&T/SID, Scottsdale Ariz., pp. 127–134, 1999.

[8] P. Fouque and G. Leurent and P. Q. Nguyen, ”Full Key-Recovery Attacks on HMAC/NMAC-MD4 and NMAC-MD5”, In: Menezes, A. (ed.) CRYPTO 2007. LNCS, vol. 4622, pp. 13–30. Springer, 2007.

[9] R. L. Rivest, ”The MD5 Message-Digest Algorithm”, Internet informational RFC 1321, April 1992.

[10] B. Kaliski, ”RSA Encryption Version 1.5”, DOI:https://doi.org/10.17487/RFC2313, RFC Editor, United States, 1998.

[11] R. S. Pressman, ”Software Engineering: A Practitioner’s Approach”, ISBN: 0-07-285318-2, McGraw Hill, 2005.

[12] V. Pimentel and G. B. Nickerson, ”Communicating and Displaying Real-Time Data with WebSocket” in Proceedings of IEEE Internet Computing, vol. 16, no. 4, pp. 45-53, 2012.

[13] ”ZBar bar code reader”, URL: http://zbar.sourceforge.net/[accessed: 2017-10-05].

[14] M. Lutz, ”Learning Python:Powerful Object-Oriented Programming”, 5th Edition, ISBN: 978-1-449-35573-9, O’Reilly, 2013.

[15] k. S. Martins and W. R. S. Lima, ”Smart door: gerenciamento e acesso remoto de portas [versão 2]”, Monografia (Graduação), Universidade Federal do Paraná, UFPR, Setor de Educação Profissional e Tecnológica, Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas, 2019.

Copyright (c) IARIA, 2020. ISBN: 978-1-61208-782-5