Security of Infocommunication Systems and Internet of Things

2023 Vol 1, No 1

https://doi.org/10.31861/sisiot2023.1.01011

Received 15 June 2023; revised 20 June 2023; accepted 30 June 2023; published 30 June 2023

Dual Authentication Technique for RFID Access Control Systems with Increased Level of Protection

Andrii Babii and Andrii Samila*

Department of Radio Engineering and Information Security, Yuriy Fedkovych Chernivtsi National University, Chernivtsi, Ukraine

*Corresponding author (E-mail: a.samila@chnu.edu.ua)

ABSTRACT Currently, there are no uniform international standards for devices and systems that use radio frequency identification technology. Using tag alone as an identifier imposes certain restrictions on the level of protection of access control systems, since the unique tag identifier can be easily copied, so two-step authentication is required. The second level of protection can be a digital password entered via the keyboard. The work presents dual authentication technique for access control to the premises. Increasing the level of protection of radio frequency identification systems is achieved by the additional use of a digital password encrypted using a symmetric block encryption algorithm and recorded on the tag. This removes any restrictions on the number of registered users in the system, because passwords and IDs are stored on tags. The laboratory model of the device is based on the NodeMCU-32S Opensource platform with an RFID module that allows connection to most fog and cloud services of the Internet of things. The development is based on the ESP32 controller, which supports the necessary set of commands and hardware for implementing the methodology, in particular, wireless standards and communication protocols, hardware-accelerated encryption. The Cayenne service from MyDevices was chosen as the cloud platform, which made it possible to configure the device using the publish-subscribe messaging protocol and develop a graphical user interface. A log on the Cayenne platform is used to keep track of users, a character display is used to display the current status and settings of the device, and a serial interface is used to debug work algorithms. The software is implemented using the Arduino C programming language.

KEYWORDS automatic identification, RFID tag, IoT security, key, cloud service.

I. INTRODUCTION

Radio Frequency Identification (RFID) is a general term used to characterize systems that wirelessly read the identification number (in the form of a unique serial number) of any object or person containing a corresponding transmit/receive device – radio tag [1-3]. RFID refers to a broad field of automatic identification technologies (Auto-ID), which also includes barcodes, optical readers and some biometric technologies, such as fingerprint and retina scanning, etc [2].

Auto-ID technologies are used to save time and labour spent on manual data entry and improve information accuracy. Some Auto-ID technologies, such as barcode systems, often require human intervention to manually scan and record information. The RFID system functions in such a way that it provides the ability to read and transfer data to a computer system without human intervention in real time [2].

At the beginning of the XXI century, RFID technology began to be actively implemented in practice, for example, Walmart and the US department of the Armed Forces required their suppliers to use RFID to mark the products delivered. It was predicted that the production of RFID systems would soon reach industrial scale and the technology would begin to be used everywhere [3].

By 2010, the rate of development of RFID technology slowed down somewhat, and interest in its use decreased. This can be explained by the appearance of some scientific studies that declare the danger of using radio tags; the difficulty of changing certain technical characteristics of RFID systems; slightly inflated prices for RFID tags.

Over the past ten years, RFID technology has been most in demand in the field of government projects, retail trade, logistics and transportation, accounting for about 60% of revenue. The greatest demand for RFID tags was in the areas of retail trade (27%), security (15,2%) and population documentation (14,4%). Manufacturing, transportation and retail industries are expected to contribute the most revenue to the total RFID market. In addition to the mentioned directions, every year developers and integrators of ready-made solutions bring to the market more and more ideas regarding the use of RFID, which undoubtedly expands the scope of application of this technology [4].

Currently, there are no uniform international standards for RFID technology. The International Organization for Standardization ISO together with the International Electrotechnical Commission IEC have developed a series of RFID standards ISO/IEC 18000 for automatic identification and control of the supply of goods [5]. These standards cover the radio interface protocol and include the seven main radio frequencies used for RFID technology worldwide.

This paper presents the development of a dual authentication technique for RFID access control systems with a higher level of protection and its practical implementation.

II. PROBLEM AND RESEARCH METHODS

The access control and management system is a set of technical and software security tools that allow one to monitor or warn about the movement or unauthorized entry of persons or vehicles into a protected area.

Also, with the help of this system, it is possible to identify persons with the right of access, register the time of their stay on the territory, set different levels of access to the premises, process information and keep statistics on its basis.

The function of an identifier can be performed by special key chains, digital codes, smartphones or cards (contactless or with a magnetic stripe). Biometrics such voice, retina, fingerprint, etc. can also be used for identification.

Using tags alone as an identifier imposes certain restrictions on the level of protection of access control systems, because the unique tag identifier can be easily copied, so two-step authentication is required. The second level of protection will be a digital password which is entered via a keyboard.

The access control system proposed in this work is based on the application of one of the most famous identifiers – the radio frequency tag of the MIFARE trademark, namely the Classic standard [6]. The password can be stored both in the memory of the controller and in the memory of the tag itself. Each storage method has its strong and weak points. The disadvantage of storing the password on the tag is that the mechanism for protecting the memory of the Crypto-1 tag was broken, so it is quite dangerous to store confidential data on it [7].

The disadvantage of storing passwords in the controller's memory is that in the event of a technical malfunction or failure of the controller or its memory, all passwords and unique identifiers will have to be entered into the system anew. This is quite difficult, if there were more than a hundred users.

With regard to the shortcomings of the considered solutions, in the proposed technique we will use a combined method of authentication. The tags will store the unique ID and password encrypted with Advanced Encryption Standard (AES) encryption, and the controller will store the AES key [8]. When the tag is raised, the unique identifier is read and immediately encrypted and compared to what is on the tag. If the identifiers match, then the system proceeds to the step of entering the password according to the same principle, namely the password is immediately encrypted after entering and compared to what is written in the tag.

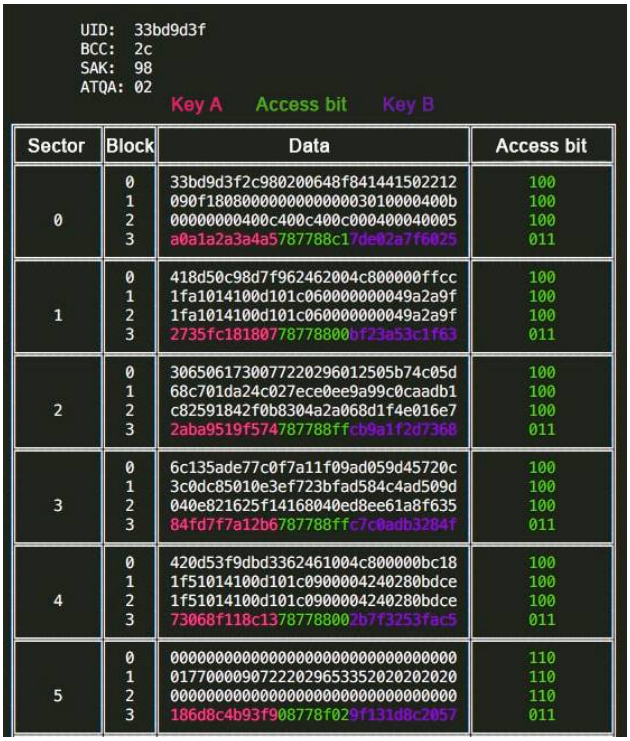
The proposed solution will allow not to limit the number of users by the memory of the controller, and in case of failure of the latter, not to overwrite or add anew each tag to the system.

A. RFID tags. To white data into tags, you need to find out the structure of their memory. But, since the library can only work with a limited list of tags of the MIFARE Classic standard, we will consider only them. The standard supports tags of the following classes:

- Classic mini;
- Classic 1K;
- Classic 4K.

Their main difference is in the amount of memory for storing information, for «mini» it is 320 bytes, for «1K» – 1024 bytes, and for «4K» – 4096 bytes.

Structurally, the memory is divided into blocks with sectors, where 1 sector comprises 4 blocks, and each block consists of 16 bytes (FIG. 1). Also, every fourth (last in the sector) block stores Crypto-1 security keys and access bits. The zero block of the zero sector contains data about the unique card identifier (UID), checksum from the UID (BCC), manufacturer information and tag type (SAK and ATQA). If these data is entered incorrectly, the tag may become unsuitable for further use.


UID: 33bd9d3f
BCC: 2c
SAK: 98
ATQA: 02

Key A Access bit Key B

Sector	Block	Data	Access bit
0	0	33bd9d3f2c980200648f841441502212	100
	1	090f180800000000000003010000400b	100
	2	00000000400c400c400c000400040005	100
	3	a0a1a2a3a4a5787788c17de02a7f6025	011
1	0	418d50c98d7f962462004c800000ffcc	100
	1	1fa1014100d101c060000000049a2a9f	100
	2	1fa1014100d101c060000000049a2a9f	100
	3	2735fc18180778778800bf23a53c1f63	011
2	0	3065061730077220296012505b74c05d	100
	1	68c701da24c027ece0ee9a99c0caadb1	100
	2	c82591842f0b8304a2a068d1f4e016e7	100
	3	2aba9519f574787788ffcb9a1f2d7368	011
3	0	6c135ade77c0f7a11f09ad059d45720c	100
	1	3c0dc85010e3ef723bfad584c4ad509d	100
	2	040e821625f14168040ed8ee61a8f635	100
	3	84fd7f7a12b6787788ffc7c0adb3284f	011
4	0	420d53f9dbd3362461004c800000bc18	100
	1	1f51014100d101c0900004240280bdce	100
	2	1f51014100d101c0900004240280bdce	100
	3	73068f118c1378778800 2b7f3253fac5	011
5	0	00000000000000000000000000000000	110
	1	01770000907222029653352020202020	110
	2	00000000000000000000000000000000	110
	3	186d8c4b93f908778f029f131d8c2057	011

FIG. 1. MIFARE Classic tag memory structure [6].

For example, the MIFARE Classic 1K card has 16 sectors, each of which is divided into four blocks. The memory structure is as follows: 16 bytes (1 Block) × 4 Blocks × 16 Sectors = 1024 bytes. To record the UID, we will use sector number 1, and for the password – sector number 2.

B. Advanced Encryption Standard. Consider the encryption algorithm which we will later use for the practical implementation of the technique. AES, also known as the Rijndael algorithm, is a symmetric block cipher that can process blocks of data using cipher keys of 128, 192 and 256 Bits [8]. The number of rounds and consecutive executions of algorithm operations depends on the size of the key. AES is adopted as an encryption standard by the US government. Currently, it is one of the most common symmetric encryption algorithms, and AES acceleration support is implemented both in x86 architecture processors (Intel, AMD), and in ARM and RISC-V architecture processors. This type of encryption is also supported by many microcontrollers at the hardware level, in which case all procedures will be

performed quite quickly and imperceptibly for the system user.

AES has some features:

- Strict adherence to the key length of 128, 192, 256 bit (16, 24, 32 bytes);
- Data is divided into blocks of 128 bits for encryption, if there is not enough data in the block, it is automatically supplemented until it is equal to 16 bytes;
- The output cipher is a multiple of 16.

Regarding the UID card identifier, usually its length in MIFARE Classic tags varies between 4 bytes and 7 bytes, so there are no special restrictions, because it will be less than 16 bytes in any case.

III. PRACTICAL IMPLEMENTATION

A. Hardware. For the hardware implementation of the proposed technique and its integration with Internet of Things (IoT) cloud services, a block-diagram shown in FIG. 2 is proposed. Consider the connection of each module and their interaction.

The basis of the laboratory model is opensource IoT platform NodeMCU-32S from Ai-Thinker, based on the ESP32 microcontroller. The platform integrates all the necessary set of peripheral devices – from capacitive touch sensors, Hall sensors, I2C, UART, SPI interfaces to Wi-Fi and Bluetooth.

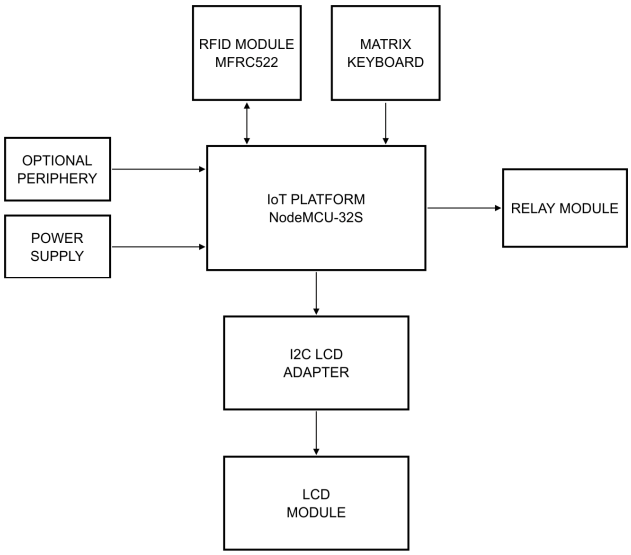


FIG. 2. Block-diagram of hardware implementation.

Opensource IoT platform (FIG. 3) is created for rapid development, testing or debugging of devices, and includes the following functional elements [9]:

- USB-UART converter based on CP2102 (or CH340 in more modern versions) for a convenient communication with personal computer;
- linear voltage stabilizer AMS1117, which regulates the input supply voltage (from the pin Vin, or from USB) to the required 3.3 V;
- micro-USB;
- two LEDs for indicating operation and data transmission through the serial port;
- input-output ports connected to the corresponding pins of the microcontroller.

The ESP32 microcontroller is designed to be scalable and adaptable, so there are many modules based on it with different configurations. It is in this board that a 2-core processor is used, with a clock frequency from 80 MHz to 240 MHz. The microcontroller sleep current consumption is less than 5 μA, making it suitable for electronics powered by a standalone power supply. The microcontroller has cryptographic hardware acceleration of encryption of popular algorithms, in particular the AES algorithm, which we used in the project [8, 9].

The RFID module is based on the NXP MFRC522 contactless reader/writer chip that operates in the HF band. We used this module to read or write MIFARE RFRID tags.

The internal transmitter, which together with the receiver module is part of the MFRC522 analogue interface, can control the read/write antenna designed for communication with ISO/IEC 14443 A/MIFARE cards and transponders without additional active circuits.

The receiver provides a reliable and efficient implementation of methods of demodulation and decoding of signals from ISO/IEC 14443 A/MIFARE compatible cards and transponders (FIG. 4). The digital module handles full framing and error detection (parity and CRC). Contactless UART is used for protocol negotiation. The FIFO buffer provides bidirectional data transfer between the contactless UART and the host. The reader is connected via the SPI interface.

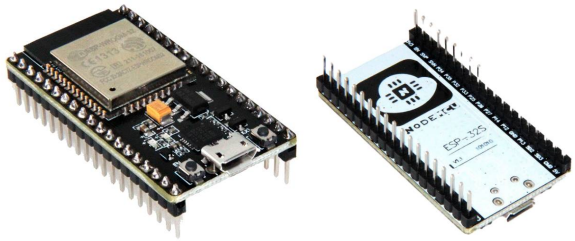


FIG. 3. Opensource IoT platform NodeMCU-32S.

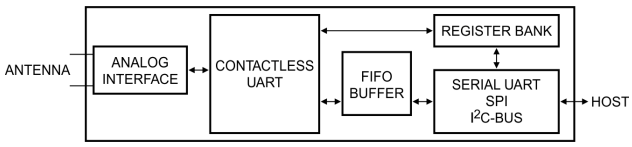


FIG. 4. Simplified block-diagram of the MFRC522.

We see that the RFID module exchanges data bilaterally, as a device for reading and writing information. But most often it will perform a direct reading function, because the writing function is needed only when registering a tag.

A simplified electrical schematic diagram of the device is shown in FIG. 5. A liquid crystal display based on the HD44780 controller is used to display information about the device operating modes and settings. The I2C LCD adapter serves as an interface between the controller and the liquid crystal display. The 3×4 matrix keyboard is used to enter the user's password during authentication and to set the operating modes. The Tongling JQC-3FF-S-Z relay module with built-in optical decoupling and LED status indication allows one to control external actuators, for instance, to turn on/off the servomechanisms of door locks.

B. Software. Let us consider the toolkit for software implementation of algorithms of the proposed method.

Espressif IoT Development Framework (ESP-IDF) is the official IoT development framework for the SoC ESP32, ESP32-S, ESP32-C and ESP32-H series of SoCs [10]. It is a self-contained set of development tools for writing applications on these platforms, using programming languages such as C and C++. Currently, ESP-IDF serves as a base for millions of devices and enables the creation of a variety of networked products, ranging from simple smart lamps and toys to large household appliances and industrial devices.

ESP-IDF supports many software components, including RTOS, peripheral device drivers, network stack, various protocol implementations and templates for typical software use cases. These components help developers focus on the overall concept of project implementation, for which the vast majority of the necessary building blocks required for typical applications are available. The open source developer tools are freely available, and the Eclipse and VSCode IDEs are officially supported for ease of use. However, in order to use this platform, the developer must have and advanced level in C and C++ programming languages, and projects also often require additional configurations.

In the program code itself, the relevant libraries are connected using the #include "library name" directive characteristic of C/C++ languages. Some third-party and integrated libraries were used to seep up the development: Keypad.h – for using matrix keyboards with Arduino; LiquidCrystal_I2C.h is an analogue of the LiquidCrystal library built into the Arduino IDE and has similar functionality, which allows you to connect liquid crystal displays based on the Hitachi HD44780 controller and similar to it; MFRC522.h – for managing the RFID module, which supports communication with MIFARE Classic tags, and allows reading or changing the UID of the tag or the information written on it; Preferences.h used as a replacement for the Arduino EEPROM library for arduino-esp32; SPI.h is part of every Arduino platform (avr, megaavr, mbed, samd, sam, arc32) and enables data exchange with SPI devices; Wire.h – enables data exchange with I2C/TWI devices; mbedtls/aes.h – for ESP32 family controllers, which allows you to use data encryption or decryption capabilities, as well as generate an encryption key.

The Cayenne MQTT ESP library provides functions to easily connect to the myDevices Cayenne IoT cloud service [11]. This library is designed to work with ESP8266 and ESP32 Wi-Fi modules, enabling data to be sent and received from the Cayenne. For use in the project, we made some changes in the modules of this library.

In the "CayenneMQTTWiFiClient.h" module, after line 41, add the line:

```
int num_try = 0;
```

and also find line 59, and replace it with the following:

```
while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    num_try++;
    if (num_try > 10)
    return; }
```

In the "CayenneArduinoMQTTClient.h" module, find line 57, and remove the "do" construction from it, but leave everything in curly brackets. Also delete the following lines:

```
while (error != MQTT_SUCCESS);
CAYENNE_LOG("Connected");
```

The proposed code modifications allow the device to work even without an Internet connection, as well as resume work if it disappears.

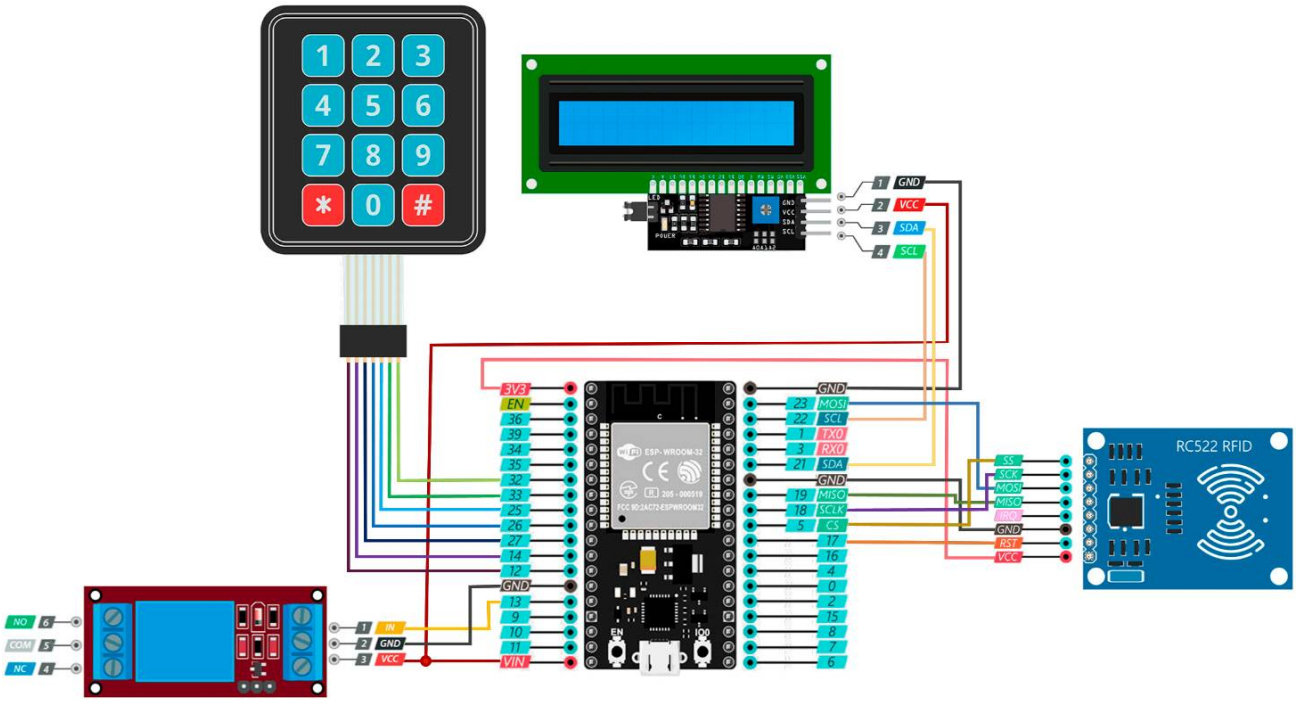


FIG. 5. Simplified schematic diagram of the device.

IV. EXPERIMENT

A. Operating principle. When the device management program is loaded, a basic check of the entire system takes place, including the health of the reader module and the stability of the connection to the Wi-Fi network. Next, the user settings are loaded. After complete readiness for work, the display shows the screen waiting for the tag, which displays the current time in the case when the lock is closed, in the opposite case, a countdown is displayed, which shows how much more time the lock will be open.

To register a tag, you need to enter the "debug" menu item and select a further action. To enter the menu, press the "#" button on the keyboard and enter the master password. The following buttons are used to navigate the menu:

"4" – to return to the previous menu item;
"6" – to go to the next menu item;
"5" or "*" to confirm the selection;
"#" – to exit the menu.

The name and functions of each menu item are listed in TABLE 1. The settings "Serial debug", "Invert gate", "Invert sensor", "Set unlock time", "Set master key" are stored in the device memory and are automatically restored when the device is turned on again. When registering a tag, first enter a password that will be written into the tag's memory, after that you need to attach the tag to the reader and the tag registration procedure will end. When one or more tags are recorded, it is enough to attach them to the reader, following which the password entry screen (FIG. 6) will appear, and if the correct password is entered, the lock will be opened.

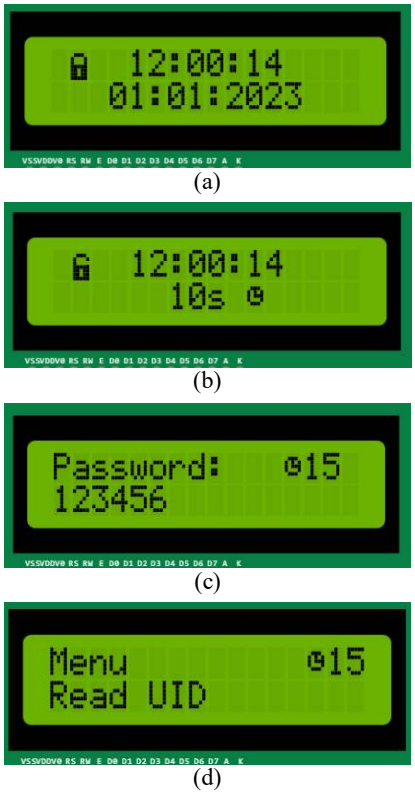


FIG. 6. Graphical user interface: (a) standby screen "closed", (b) standby screen "open", (c) password input screen, (d) menu view.

TABLE 1. Menu layout description.

Designation	Action
Exit	Exit the menu
Read UID	Read UID tags
Register new tag	Register new tag
Clear tag	Clear tag
System status	System status (displays the presence of Wi-Fi network and the status of the reader)
Set unlock time	Set opening time
Serial debug	Enable/disable the display of service information in the Serial-port
Invert gate	Invert the output
Invert sensor	Invert sensor/button/peripheral input
Set master key	Set a new master key
Reboot	Reboot a device
Reset settings	Reset to default settings

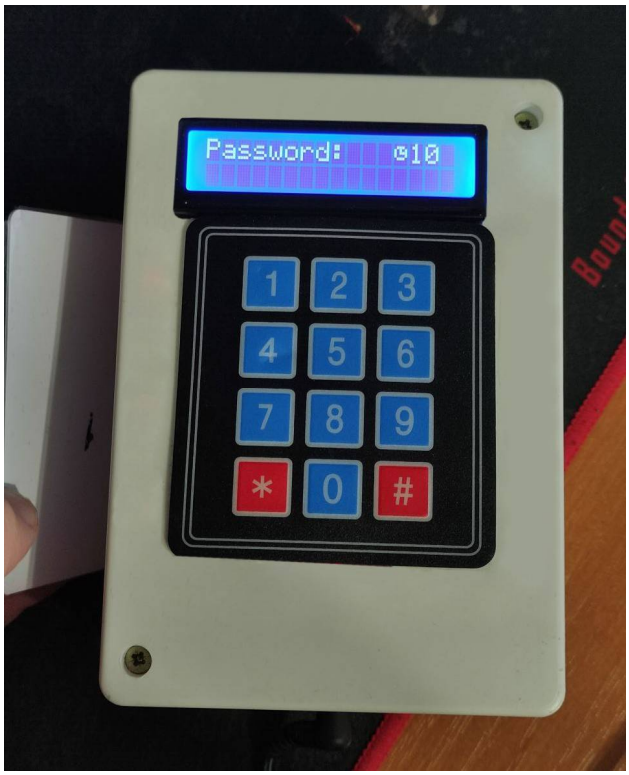


FIG. 7. Photo of the prototype of the main RFID module of the access control system with an increased level of protection.

B. Integration with IoT cloud service. To connect the proposed device to the Internet, the Cayenne IoT cloud service platform from myDevices is chosen, an IoT prototyping tool that supports hundreds of devices, from the most famous development boards to professional remote sensors, incl [11].

To connect physical devices of "things" with IoT cloud services, one can use HTTP and MQTT data transfer protocols, or specialized solutions, such as AWS IoT Core from Amazon. For our purposes, the MQTT protocol was used [12]. There is also a version of the MQTT-SN (MQTT for Sensor Networks) protocol, formerly known as MQTT-S, which is designed for use with wireless devices that do not support TCP/IP protocols, such as ZigBee.

The main features of the MQTT protocol:

- asynchronous protocol;
- compact messages;
- work in conditions of unstable communication on the data transmission line;
- support for several levels of quality of service (QoS);
- easy integration of new devices.

The exchange of messages in the MQTT protocol is carried out between a client, which can be a publisher or subscriber, and a broker of messages (for example, Mosquitto MQTT) [12].

The MQTT devices use certain types of messages to interact with the broker, the main ones being:

- Connect – establish a connection with the broker;
- Disconnect – disconnect from the broker;
- Publish – publish data to a topic on the broker;
- Subscribe – subscribe to a topic on the broker;
- Unsubscribe – unsubscribe from the topic.

To start working with Cayenne IoT, you need to register or log in using your personal login and password on the mydevices.com website. After authorization, we will be able to add the device to the system using the MQTT protocol, for which purpose the MQTT Username, MQTT Password, Client ID obtained on the website must be specified in the program code (FIG. 8). After connecting the device to the Internet, we will get to the control panel on the website, which allows adding control elements (widgets) (FIG. 9). Some elements will be added automatically, they can be removed or configured for use in the project.

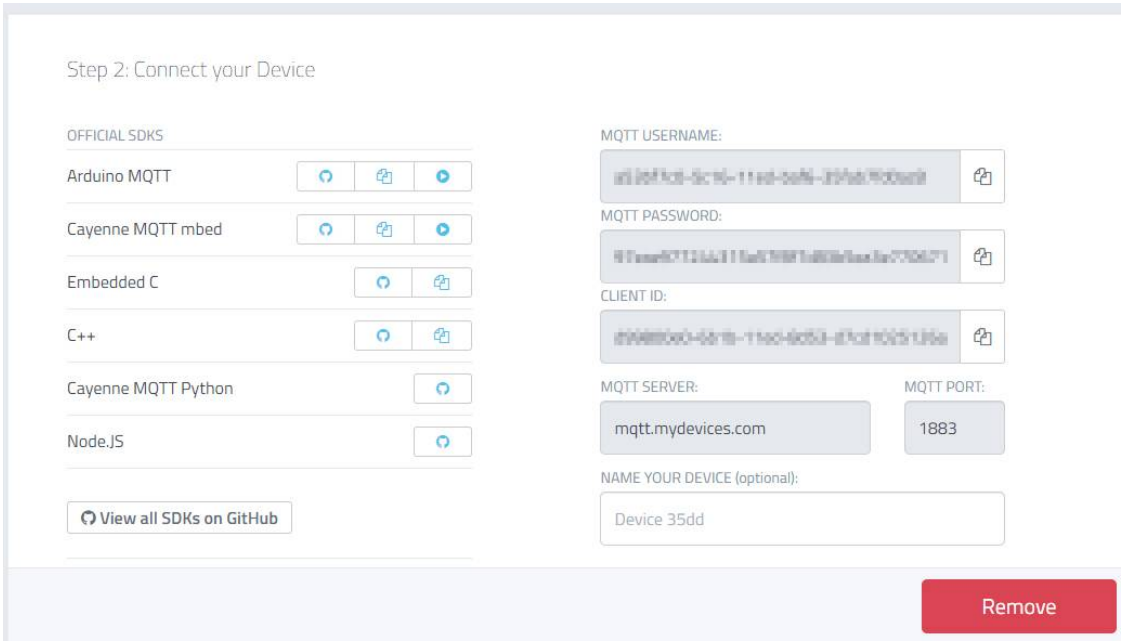


FIG. 8. MQTT device connection panel in myDevices Cayenne.

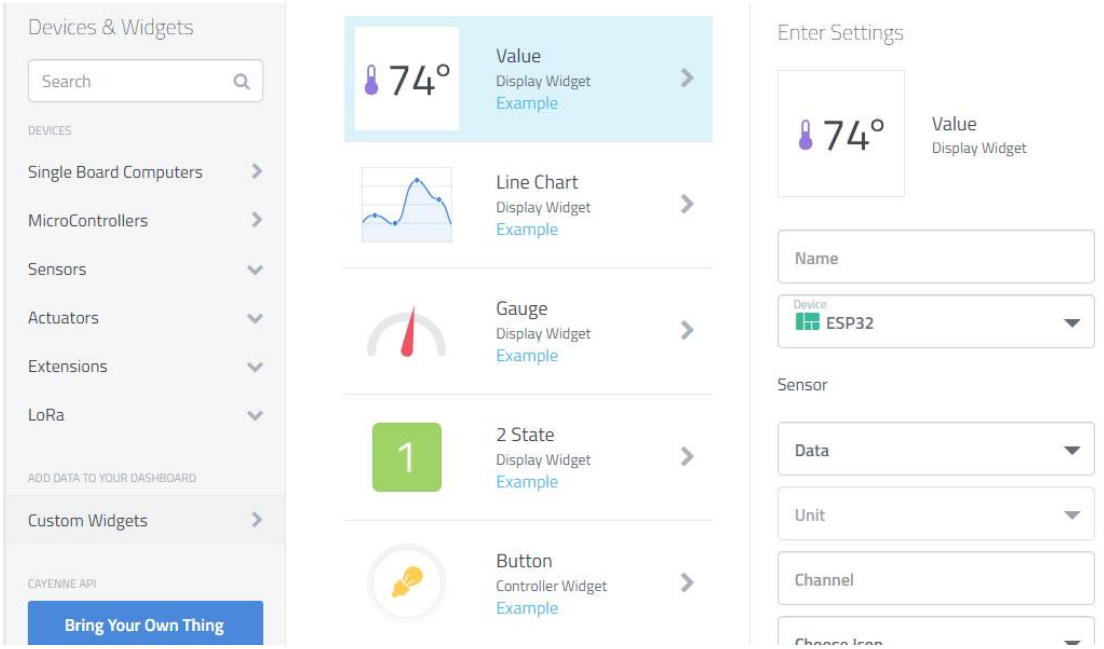


FIG. 9. The item selection panel in myDevices Cayenne.

When adding an element, you must specify the name of the element, the channel number and the type of data the element will work with. Information about the channels we used in the development is shown in TABLE 2. Channel 0 does not display information directly, it serves as a buffer to send the user ID to the server for display in the log.

TABLE 2. Data channels for exchange.

Channel	Data type	Description
0		The channel in which events are received in the log
1	Digital (0/1)	Lock status indication
2	Digital (0/1)	Sending a command to open the lock
3	Digital (0/1)	Sending a command to close the lock
10	Analog	Display current opening timer
15	Analog	Display/set current maximum opening time

The location of the elements of the graphical user interface (FIG. 10) can be in any order, but it is worth highlighting the main blocks:

1 – informs about the state of the lock;

2 – lock state control;

3 – setting the opening time.

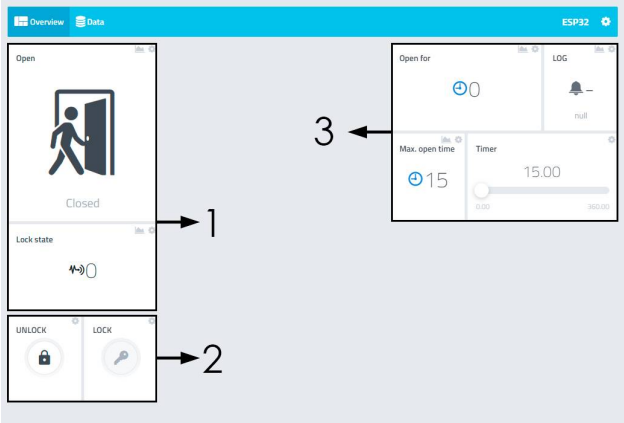


FIG. 10. Device control panel in myDevices Cayenne.

In the "Data" tab of myDevices Cayenne, there is a log that allows one to track the means of opening the lock and the time when it happened. The Timestamp column of the "Data" tab shows the time, and the Values column shows the tool.

V. CONCLUSION

The dual authentication technique proposed in the work for RFID access control systems with an increased level of protection has confirmed its effectiveness.

In the course of the research, the features of authentication and the possibility of increasing the level of RFID protection with the additional use of a digital password, which is encrypted using the AES algorithm and recorded on the tag. This removes any restrictions on the number of registered users in the system, since passwords and UID are stored on the tag.

The research resulted in practical implementation of the hardware and software tools of the laboratory model of access control device with increased level of protection. The feature of the development is its full integration with most IoT cloud services that support the MQTT standard. In particular, the device passed a successful test for compatibility with the Cayenne service from MyDevices.

AUTHOR CONTRIBUTIONS

A.B., A.S. – conceptualization, methodology; A.B., A.S. – investigation; A.B., A.S. – writing (original draft preparation), A.S. – writing (review and editing).

COMPETING INTERESTS

The authors declare no competing interests.

REFERENCES

[1] L. Bin, Z. Rong, L. Sifeng, "RFID system and its perspective analysis with KERGM(1,1) model," *Journal of Computers,* vol. 3, pp. 9-15, July 2008.

[2] D. C. Wyld, "24-karat protection: RFID and retail jewelry marketing," *International Journal of UbiComp,* vol. 1, pp. 1-14, Jan 2010.

[3] C. Munoz-Ausecha, J. Ruiz-Rosero, G. Ramirez-Gonzalez, "RFID applications and security review," *Computation,* vol. 9, pp. 69 – 21, Jun 2021.

[4] G. M. Bianco, E. Raso, L. Fiore, V. Mazzaracchio, L. Bracciale, F. Arduini, et al., "UHF RFID and NFC point-of-care – architecture, security, and implementation," *IEEE Journal of Radio Frequency Identification,* doi: 10.1109/JRFID.2023.3268422.

[5] International Organization for Standardization. (2023, Jun. 26). *Automatic identification and data capture techniques Including RFID, OCR, bar coding, etc.* [Online]. Available: https://www.iso.org/ics/35.040.50/x/

[6] B. O. Kose, H. Uluoz, V. Coskun, "Secure design on MIFARE Classic cards for ensuring contactless payment and control services," *Advances in Cyber-Physical Systems,* vol. 7, pp. 22-28, 2022.

[7] N. T. Courtois, K. Nohl, S. O'Neil. (2008). *Algebraic attacks on the Crypto-1 stream cipher in MIFARE Classic and oyster cards* [Online]. Available: https://eprint.iacr.org/2008/166

[8] J. Nechvatal, et al., "Report on the development of the Advanced Encryption Standard (AES)," National Institute of Standards and Technology, U.S. Department of Commerce, Gaithersburg, Sci Rep. 106, 2001.

[9] Shenzhen Ai-Thinker Technology Co., "Nodemcu 32s," datasheet, 2019.

[10] Espressif Systems. (2023, Jun. 26). *ESP-IDF Programming Guide* [Online]. Available: https://docs.espressif.com/projects/esp-idf/en/latest/esp32/

[11] GitHub Inc. (2019, Feb. 15). *Cayenne MQTT ESP Library* [Online]. Available: https://github.com/myDevicesIoT/Cayenne-MQTT-ESP

[12] F. Azzedin, T. Alhazmi, "Secure data distribution architecture in IoT using MQTT," *Applied Sciences,* vol. 13, pp. 2515 – 13, Feb 2023.

Andrii Babii

Received the BSc in Radiotechnics and Telecommmunication systems in 2021 and the MSc degree in Cybersecurity in 2022 from Yuriy Fedkovych Chernivtsi National University.

Andrii Samila

Yuriy Fedkovych Chernivtsi National University. D.Sc. (Engineering), Full Professor, Vice Rector for Scientific Research. Research interests: IoT, Microelectronics & Electronic Packaging, Signal Processing, Computer Hardware Design, Robotics, High Energy & Nuclear Physics. Author of nearly 200 publications in this research area.

Методика подвійної автентифікації для RFID систем управління доступом з підвищеним рівнем захисту

Андрій Бабій, Андрій Саміла*

Кафедра радіотехніки та інформаційної безпеки, Чернівецький національний університет імені Юрія Федьковича, Чернівці, Україна

*Автор-кореспондент (Електронна адреса: a.samila@chnu.edu.ua)

АНОТАЦІЯ Наразі не існує єдиних міжнародних стандартів для пристроїв і систем, що працюють за технологією радіочастотної ідентифікації. Використання лише одних міток в якості ідентифікатора накладає певні обмеження на рівень захисту систем управління доступу, тому що унікальний ідентифікатор мітки можна з легкістю скопіювати, тому потрібна двоетапна автентифікація. Другим рівнем захисту може бути цифровий пароль, який вводиться за допомогою клавіатури. В роботі представлено методику подвійної автентифікації для систем управління доступом до приміщення. Підвищення рівня захисту систем радіочастотної ідентифікації досягнуто шляхом додаткового використання цифрового паролю, що зашифровується за допомогою симетричного алгоритму блочного шифрування та записується на мітку. Це знімає будь-які обмеження на кількість зареєстрованих користувачів в системі, адже паролі та ідентифікатори зберігаються на мітках. Лабораторний макет пристрою реалізовано на основі Opensource платформи NodeMCU-32S з модулем радіочастотної ідентифікації, яка уможливлює підключення до більшості туманних та хмарних сервісів Інтернету речей. Основою розробки є контролер ESP32, який підтримує необхідний набір команд та апаратних засобів для реалізації методики, зокрема – бездротові стандарти і протоколи зв'язку, апаратне прискорення шифрування. У якості хмарної платформи обрано сервіс Cayenne від MyDevices, який дозволив налаштувати пристрій за допомогою протоколу обміну повідомленнями за принципом видавець-підписник та розробити графічний інтерфейс користувача. Для ведення обліку користувачів використовується журнал на платформі Cayenne, символьний дисплей використовується для відображення поточного стану та налаштувань пристрою, послідовний інтерфейс використовується для відлагодження алгоритмів роботи. Програмні засоби реалізовані з використанням мови програмування Arduino C.

КЛЮЧОВІ СЛОВА автоматична ідентифікація, RFID-мітка, безпека IoT, ключ, хмарний сервіс.