

# Implementation Cryptography and Access Control on IoT-Based Warehouse Inventory Management System

Muhammad Yusuf<sup>1</sup>, Arizal Arizal<sup>2</sup>, Ira Rosianal Hikmah<sup>3</sup>  
Politeknik Siber dan Sandi Negara, Bogor, Indonesia

---

## Article Info

### Article history:

Received August 12, 2022  
Revised September 20, 2022  
Accepted October 20, 2022

### Keywords:

Cryptography  
Database  
Raspberry Pi 3  
Two-Factor Authentication  
Warehouse Inventory Management System

---

## ABSTRACT

Warehousing is a product storage management activity to ensure product availability, so inventory management is needed to oversee the movement of logistics and equipment. Some things need to be considered in the storage process, such as the suitability of the storage location, safe from theft, and safe from physical disturbances. Vulnerabilities can occur when unauthorized users find out information from the database regarding stored goods, so a security mechanism for the warehouse database is needed. In addition, proper identification needs to be made of someone trying to access the database. In this research, a Warehouse Inventory Management System (WIMS) was created by implementing the AES-128 cryptographic algorithm, which was built using ESP32 and Raspberry Pi 3 devices. Time Password (T-OTP). The results show that the built system can overcome inventory problems in conventional warehousing management systems and implement data security using the AES-128 algorithm. The application of two-factor authentication in the form of smartcards and T-OTP shows very good results in testing its accuracy to overcome the vulnerability of unauthorized access to the system database.

Copyright ©2022 MATRIK: Jurnal Manajemen, Teknik Informatika, dan Rekayasa Komputer.  
This is an open access article under the [CC BY-SA](#) license.



---

## Corresponding Author:

Arizal, 081574490480,  
Cryptographic Engineering, Cryptographic Hardware Engineering,  
Politeknik Siber dan Sandi Negara, Bogor, Indonesia,  
Email: [arizal@poltekssn.ac.id](mailto:arizal@poltekssn.ac.id).

---

How to Cite: M. Yusuf, A. Arizal, and I. Hikmah, Implementation Cryptography and Access Control on IoT-Based Warehouse Inventory Management System, MATRIK : Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer, vol. 22, no. 1, pp. 37-50, Nov. 2022.

This is an open access article under the CC BY-NC-SA license (<https://creativecommons.org/licenses/by-nc-sa/4.0/>)

## 1. INTRODUCTION

A warehouse as a place to store goods requires data accuracy for every transaction of goods that exists. Warehousing refers to management activities that involve appropriately storing products on a large scale and ensuring product availability when needed [1]. The Internet of Things (IoT) can show complex relationships between data sources and recipients [2]. Manajemen pergudangan dirancang untuk kepentingan pengolahan kegiatan pergudangan yang akan mempengaruhi seluruh proses produksi. Manajemen pergudangan yang dikelola dengan baik akan dapat meningkatkan efisiensi pengendalian atau penanganan materian di gudang [3]. Therefore, it is necessary to have control, which is a process of monitoring activities for the entry and exit of logistics and equipment from and to the warehouse so that inventory and placement can be known quickly, precisely, accurately, and accountably.

The Internet of Things (IoT) can have a significant impact on the warehousing sector, such as warehouse automation, Enterprise Resource Planning (ERP), and inventory management [4]. Inventory management is a form of tracking the company's stock of goods by monitoring several indicators such as type, weight, quantity, and location [5]. Conventional warehouse inventory management affects efficiency in storing, inventorying, taking, and placing goods [6]. In a previous study, Cidal, et al. [5], explained that there are conventional warehouse inventory methods, namely man lift and basket, which require workers to carry out an inventory process using hydraulic scissors or by tying themselves to RT's Forks and then filling out the paper in their hands according to the information read from the shelves. Both methods pose a serious safety threat when logging goods at a height of 10-15 meters. Another problem arises when workers have to search for items that have been stored manually by relying on their memory because not every worker has a strong memory and generally uses a tool like notes. In another previous study, Tejesh & Neeraja [1] succeeded in building a Warehouse Inventory Management System (WIMS) based on IoT (Internet of Things) using RFID technology as a scanner to obtain product storage data in the form of tag numbers, product descriptions, locations and storage timestamps. be a solution to problems in conventional warehouse inventory. However, in the research of Tejesh & Neeraja, they still have not implemented security in the process of sending data or databases.

Basically, in the warehousing mechanism, several things need to be considered in the storage process, such as storing goods in the warehouse according to the plan, safe from theft, and safe from physical disturbance. Vulnerabilities can occur when unauthorized users find out information about stored goods, causing a threat to the goods. This, of course, can be easily found through the warehouse database that stores detailed information about the goods stored, so a security mechanism is needed to implement access control to the warehouse database to prevent theft and physical disturbances to the goods in the warehouse.

In general, databases contain data with various levels of urgency, and the data is shared with various users with different privileges, so it needs to be managed and protected because any changes to the database can affect its integrity [7]. Data security is crucial and is the main concern of various communication system providers. In the world of information, there is a lot of important and confidential data that should not be known by the public [8]. In order for the data stored in the database to have high security, the data must be secured. Data can be secured through data transmission and data storage in the database. Data security can be applied in the warehousing sector that stores items with high selling value, high urgency, or secrets, such as weapons warehouses, combat vehicle warehouses, password equipment warehouses, and chemical laboratory warehouses by implementing a secure Warehouse Inventory Management System (WIMS). Data security can be done using cryptographic techniques to maintain the confidentiality and authenticity of data so as to improve the security aspects of an information. One of the cryptographic algorithms that can be used for data security is the Advanced Encryption Standard (AES) [8].

The AES algorithm is a symmetrical block cipher algorithm with a particular structure to perform the encryption and decryption on sensitive data, which is widely applied to hardware and software. This is because it is complicated for hackers to get plaintext data that has been encrypted using the AES algorithm. Until now, there has been no evidence of someone successfully hacking data encrypted using the AES algorithm [9]. The AES algorithm is a good choice for use in applications with high confidentiality and integrity due to its high cryptographic strength [10]. Proper identification also needs to be done, in addition to securing the stored data to verify whether people trying to access a facility should be allowed or not [11]. Identification and authentication can be verified using access control using a smart card (smartcard), matching anatomical attributes (biometric system), displaying photo ID for security guards, or entering passwords or PINs [? ]. The AES algorithm is a cryptographic algorithm that can encrypt and decrypt data with varying key lengths, one of which is 128 bits long [12].

In this research, the development of research [1] regarding creating a Warehouse Inventory Management System (WIMS) will be carried out with the development of implementing the AES-128 cryptographic algorithm on scanned data sent from the ESP32 module to the Raspberry Pi 3 central server. High frequency, low cost, built-in sensors, integrated Wi-Fi and Bluetooth, speed up to 150Mbps, 10-pin ADC, power management module, low noise amplifier receiver, security, filtering, easy to install, and more [13]. This research also uses an access control scheme on the database in the form of using a smartcard with the addition of authentication in the form of One Time Password (OTP) with the type of Time Based-One Time Password (T-OTP), which will be implemented in the development of this research. T-OTP is a one-password algorithm that uses the present time as unique [14]. That is, the

T-OTP code on the system uses a time-based method so that the user must pay attention to the time settings on his device. If the time setting is not the same as the time on the server, it is inevitable that every time you generate the T-OTP code, it will not be able to be used to enter the system [15]. Smartcard in this research is also used as authentication because it has the advantage of portability, effectiveness, and low computing [16]. Smartcard-based authentication containing passwords is a convenient and effective user authentication mechanism. This smartcard technology has been widely used for authentication applications such as host login, online banking, and e-health services [16]. Even so, there are still vulnerabilities in using smartcards, namely smartcard PIN sniffing; when a smartcard is lost or accidentally taken by an unauthorized person, that person can use it to create fake login messages and then impersonate the user to bypass the server [16, 17]. The problem of vulnerability on the smartcard can be overcome by using OTP authentication techniques to ensure security when the smartcard is lost or stolen [16, 17].

## 2. RESEARCH METHOD

This prototype was built with the desired function, namely being able to carry out an inventory process of goods with high urgency stored in the warehouse, with some product storage data indexes in the form of item ID, item name, location, and storage time stamp. The warehouse database can only be accessed by people who have been authenticated from the results of access restrictions using Two-Factor Authentication in the form of smartcards and T-OTP.

The T-OTP code will be sent to the previously registered user's telegram account via the telegram bot, where the user's telegram id is stored in the system database, which is also the user's identity. In implementing T-OTP, web applications and telegram bots are used. The T-OTP library is implemented in the source code created to build a web application that can send T-OTP code in the form of messages to the user's telegram account. The use of smartcards and T-OTP will be the answer to the need for authentication (authentication) of the system.

In this prototype, the AES-128 algorithm is also implemented for the data sent to the database, which results from scanning RFID tags on goods that contain information about goods stored in the warehouse. The implementation of the AES-128 algorithm is to answer the needs of the integrity (integrity) of the data sent. An overview of this research system can be seen in Figure 1.

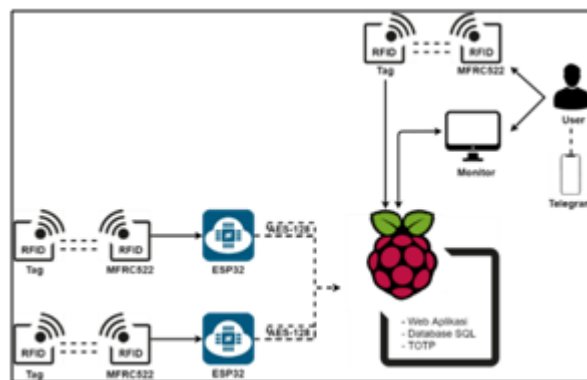


Figure 1. System Overview

The stages of making the Warehouse Inventory Management System (WIMS) prototype refer to the System Development Life Cycle (SDLC), which uses a waterfall development approach. The following are the steps carried out in this research which consist of the planning, analysis, design, and implementation phases. [18]:

### 1. Planning

This is the first step to raising the selected problem. It is necessary to know the background of the researcher conducting this research and the reasons for choosing the problem to be studied. At this stage, the steps of the research work are arranged to be able to overcome the issues selected. Planning in the research will use a waterfall development approach so that this research began to be compiled after the problems that were happening in the conventional Warehouse Inventory Management System (WIMS).

### 2. Analysis

This stage begins with analyzing the needs of devices and components and then identifying functional and non-functional requirements to build the system. Analysis of device and component requirements is carried out by studying literature on

the main reference journal articles for the development of the Warehouse Inventory Management System (WIMS) as well as journal articles discussing the use of smartcards, AES-128 algorithms, and T-OTP type OTP as the basis for research. Tables 1 and 2 are tables of functional requirements and non-functional requirements in this research.

Table 1. Functional Requirements

Functional Requirements	Explanation
Can carry out the process of scanning goods	Scan using RFID
Can save scanned data for incoming goods	Save the scanned data on the database server
Can reduce item data for outgoing goods	Delete the scanned data on the database server when reading the same item ID.
Can encrypt data sent to the database server	The data sent to the database server is encrypted using a cryptographic algorithm
There is a login mechanism using a smartcard and T-OTP on the web application	Authenticate the web application to the user
There is a logout mechanism	The user stops using the web application or exits
There is a user registration mechanism	User registers on the web application
The system can authorize the user to access the database	Accessible database containing stored item data

Table 2. Non-Functional Requirements

Non-Functional Requirements
Communication between ESP32 and Raspberry Pi uses Wi-Fi
The transmitted data is encrypted using the AES-128 algorithm
The database used is MySQL
The OTP code sent is the type T-OTP
The system is built using ESP32, RFID MFRC522, and Raspberry Pi 3 Model B . devices

### 3. Design

At the design stage, the system design is built and describes how the processes occur in the system are carried out. At this stage, it is determined how the system will work, what devices support the system's operation, how the user interface will be, and all supporting needs in the system. The output of this stage is a prototype Warehouse Inventory Management System (WIMS), which is a simple simulation device using Raspberry Pi 3 Model B, RFID, and ESP32 that implements access control in the form of smartcards and T-OTP type OTP as user authentication and database security. This system database will be displayed on the web application so authenticated users can access it.

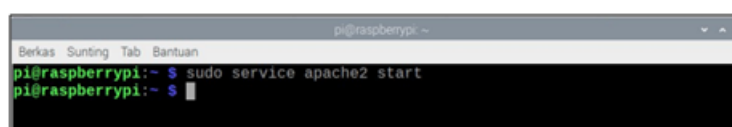
### 4. Implementation

It is a stage divided into two parts: system development and system testing. The system is built using the Python programming language for Raspberry and the Arduino and PHP programming languages to build web applications. The system testing section in this research aims to determine the system's performance and feasibility. Some of the tests carried out are vector testing, unit testing, integration testing, and system testing. Vector testing is carried out to test the suitability of the cryptographic algorithm used, whether it has been properly implemented, and whether it provides the expected output or not. Unit testing aims to test all functions of each most negligible unit of the system, whether it has been running correctly or not, before being integrated with other units. Integration testing is a test to ensure that the results of the integration of each unit in the system can run well and form a system as expected. System testing is carried out after all units in the system have been integrated and aims to ensure whether the entire system can work according to the requirements that have been set or not.

## 3. RESULT AND ANALYSIS

### 3.1. Server Initiation Process

As shown in Figure 2, the server initiation process is the initial stage for running the system. The purpose of the initiation process is to turn on the local server, namely localhost, so that the system on the device can connect to the database.



```

pi@raspberrypi: ~
Berkas  Sunting  Tab  Bantuan
pi@raspberrypi:~$ sudo service apache2 start
pi@raspberrypi:~$

```

Figure 2. Localhost Server Initiation Process

### 3.2. Implementation of Source Code MFRC522 on ESP32

At this stage of implementing the MFRC522 source code on the ESP32, there are two stages, namely the implementation of the source code write data to write data on the RFID tag and the implementation of the source code read data for scanning the RFID tag. The implementation of this source code uses the Arduino IDE application. The execution of this source code aims to write data in the form of item ID and item name to be stored in the database. Writing data on RFID is done in block 1 and block 2. Block 1 will store the item ID, while block 2 will store the item name. When writing data on the smartcard, those entered in block 1 and block 2 must be 16 characters long. This is an adjustment to the length of characters that can be encrypted using the AES-128 algorithm, which is 16 characters long so that if the data entered in the smartcard is less than 16 characters, padding (\*) will be added behind the last character so that the total character becomes 16 characters as shown in Figure 3.

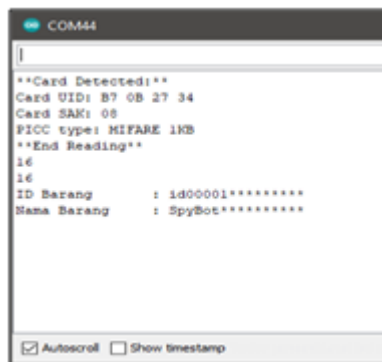


Figure 3. Read Data Smartcard

During the process of writing data on the smartcard, the data entered in blocks 1 and 2 in the smartcard must each be 16 characters long. This is an adjustment to the length of characters that can be encrypted using the AES-128 algorithm, which is 16 characters long, so if the data entered in a smartcard is less than 16 characters, padding (\*) will be added after the last character entered so that the total character is 16 characters as shown in Figure 3.

### 3.3. Implementation of Source Code MFRC522 on Raspberry Pi

Implementing the MFRC522 source code on the Raspberry Pi is carried out for scanning the UID on the smartcard for the authentication process using the programming application available on the Raspberry Pi, namely Thonny. The implementation of the MFRC522 source code on the Raspberry Pi can be seen in Figure 4.

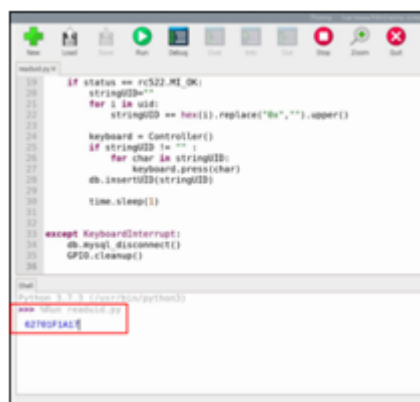


Figure 4. MFRC522 Implementation on Raspberry Pi

### 3.4. System Development

In the development phase, the Warehouse Inventory Management System (WIMS) is made by integrating all the tools that make up the WIMS prototype. The display of the WIMS prototype builder device can be seen in Figure 5.



Figure 5. Warehouse Inventory Management System Prototype

#### 1. Item Tag RFID Scan

This RFID tag scanning process aims to identify incoming goods that will be stored in the warehouse. The results of scanning the RFID tag in the form of item ID and item name will be sent to the database for storage. The RFID tag scanning process is carried out using the MFRC522 and ESP32 devices, then ESP32 will send the data to the system database. The goods tag RFID scanning device is shown in Figure 6.



Figure 6. Item Tag RFID Scan Process

#### 2. Shipping RFID Goods Tags

The process of sending RFID tag data occurs after the MFRC522 successfully reads the data stored in the RFID tag. The data that has been read will be sent to the system database, as shown in Figure 7. This item's RFID tag data is delivered by an item's RFID tag scanner consisting of an ESP32 module and an MFRC522 RFID module. In this prototype, there are two RFID tag goods scanner devices, each of which will be identified as an RFID goods tag scanner in room 1 and room 2.

```

COMM4
*****
**Card Detected:**
Card UID: 10 4B C7 A3
Card SAK: 08
PICC type: MIFARE 1KB
**End Reading**
ID Barang : id00005*****
21ad7ba907aed0eb50ab74f08d3ee5de
Nama Barang : Rheinmetall*****
ff5282e02e41d9b6878a334ca3619de3
connecting to 192.168.137.234
Requesting URL: . . .
Pendaftaran Berhasil.
Menutup koneksi...
Selamat Datang
 Autocroll  Show timestamp

```

Figure 7. The Process of Sending Items RFID Tag Data

### 3. Item Tag RFID Encryption

This encryption process occurs right before the process of sending the scanned data. Data that the RFID tag has successfully read will be secured using the AES-128 encryption algorithm, with a fixed key and IV (Initial Value) length of 16 bytes. The ESP32 device carries out this data encryption process on the goods RFID tag scanner device. The encrypted data is 16 characters long, so if, during scanning, it is found that the ID and name of the item stored in the RFID are less than 16 characters, the system will automatically pad the character (\*) behind the last letter until the ID and/or name of the item is 16 characters. Furthermore, the data encryption process is carried out using the AES-128 algorithm. The method of data encryption for RFID tags of goods in this prototype is shown in Figure 8.

```

COMM4
*****
**Card Detected:**
Card UID: 10 4B C7 A3
Card SAK: 08
PICC type: MIFARE 1KB
**End Reading**
ID Barang : id00005*****
21ad7ba907aed0eb50ab74f08d3ee5de
Nama Barang : Rheinmetall*****
ff5282e02e41d9b6878a334ca3619de3
connecting to 192.168.137.234
Requesting URL: . . .
Pendaftaran Berhasil.
Menutup koneksi...
Selamat Datang
 Autocroll  Show timestamp

```

Figure 8. Item Tag RFID Data Encryption Process

### 4. Item Tag RFID Decryption

The process of decrypting the encrypted data occurs after the data transmission process. The data sent will be decrypted using the AES-128 encryption algorithm with the same key and IV length as the data encryption process, which is 16 bytes. This data decryption process is carried out on the server before the data is stored in the database using the PHP programming language. Figure 9 shows the results of the RFID tag decryption of goods received on the Raspberry Pi server. The process of documenting the results of data decryption is carried out separately from the system so that the results of data decryption before being stored in the database can be documented.

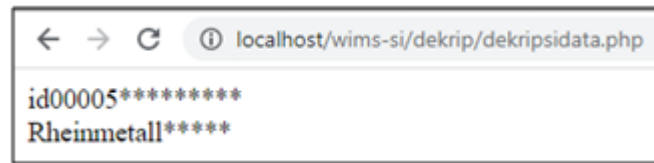


Figure 9. The Process of Decrypting Data for RFID Items

### 5. Storage of RFID Items Data in Database

The process of storing RFID tag data for goods occurs after the server successfully decrypts the scanned data, so that plaintext data is generated in the form of item ID and item name. The database used is MySQL using the PHP programming language. This database can be accessed using the localhost server. Figure 10 shows the data storage of RFID tags of goods. In the process of storing the RFID tag for the goods, goods data are stored in the form of the ID of the goods, the name of the goods, the time of storage, and the location of the goods being stored, which are distinguished based on the data received from the delivery by the RFID tagging device for the goods.

	no	id	barang	lokasi	timestamp
<input type="checkbox"/>	67	id00005*****	Rheinmetall*****	ruang2	2021-04-12 14:01:38
<input type="checkbox"/>	68	id00003*****	QCRYH*****	ruang1	2021-04-12 14:01:42

Figure 10. Item Tag RFID Data Storage

### 6. User Smartcard UID Scan

Scanning the UID user smartcard occurs on a device that is different from the device that reads item data, where the MFRC522 device, which acts as the UID scanner module, will be directly connected to the Raspberry Pi 3 device as a central server as shown in Figure 11. This UID scan is the first authentication of users who will access the warehouse database.



Figure 11. User Smartcard UID Scan



## 7. Sending T-OTP code

The process of sending this T-OTP code occurs when the user has been successfully authenticated based on the UID of his smartcard. If the UID read is the same as the UID stored in the database, the user passes the second authentication, namely T-OTP and Two-Factor Authentication. This T-OTP code will be sent via the WIMS-TOTP telegram bot to the registered user's telegram chat ID as shown in Figure 12, so that only users who have subscribed to the WIMS-TOTP telegram bot and whose telegram chat ID is already stored in the database, can receive T-OTP code sent.



Figure 12. Sending T-OTP Code

## 8. Web Application Development

This web application functions as a user interface to access the Warehouse database. On the web application, the user must successfully pass Two-Factor Authentication, namely smartcard and T-OTP to access the warehouse database containing data on goods stored in the warehouse. The process of making this web application uses the PHP programming language, which is also connected to the MySQL database. In this web application, the admin and user each have different privileges in accessing the database. This is done to limit user access rights. Figure 13 shows the admin/user login page that will open the Warehouse database. This is the first page that the admin/user will encounter when accessing the web application, where the admin/user must pass the first authentication by using the UID on the smartcard. After the user is successfully authenticated using the UID on the smartcard, this second authentication requires the admin/user to enter the T-OTP code that has been received on his telegram account. The T-OTP code is sent to the telegram account when the admin/user UID is authenticated.



Figure 13. Login Page

Figure 14 is an inventory data page which is the main page of this web application. On this inventory data page, the admin/user can see specific data about the goods stored in the warehouse. The specific data displayed is each item's item ID, name, location, and storage timestamp. In addition, there is also a delete button that helps delete the data of the item.

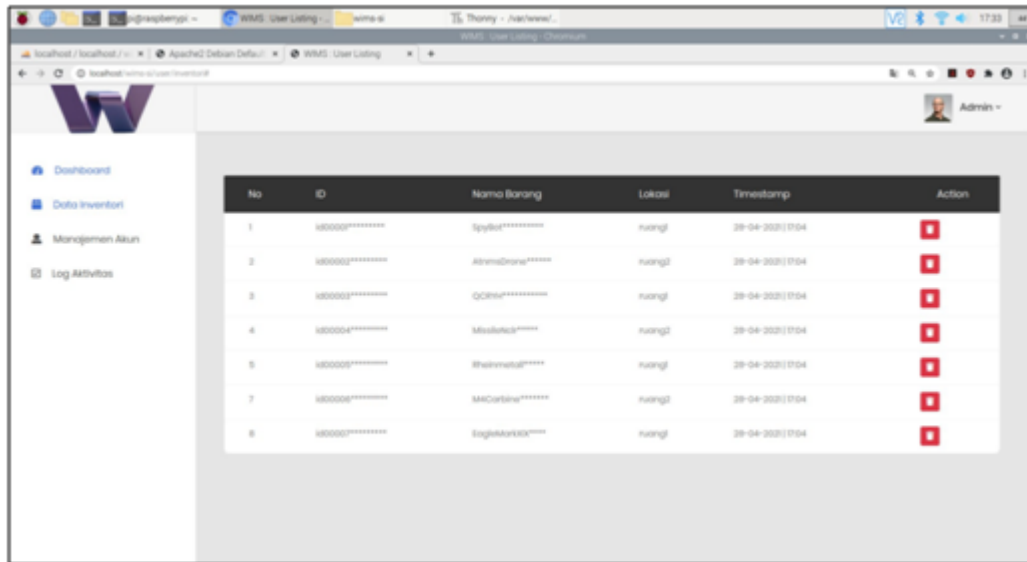


Figure 14. Inventory Data Page

## 9. Testing

The purpose of testing is to determine and ensure the suitability of the performance of the system that has been made. The tests carried out include test vectors, unit testing, integration testing, and system testing, whose test results can be seen in Table 3. Based on Table 3, the Two-Factor Authentication accuracy test was conducted to determine the accuracy of the authentication process using smartcards and T-OTP. This research also tested the accuracy of the implementation of Two-Factor Authentication, with the login scenario as follows:

- Scenario 1: User uses UID on the unregistered smartcard.
- Scenario 2: The user uses the UID on the registered smartcard, and the T-OTP code is incorrect.
- Scenario 3: The user uses the UID on the registered smartcard, and the T-OTP code is correct with a time range of 30 seconds.
- Scenario 4: The user uses the UID on the registered smartcard, and the T-OTP code is correct with a time range of  $\geq$  30 seconds.

Table 3. Test Results

Testing	Scenario Testing	Test Results
Test Vector AES-128	Proving the suitability of the results of implementing the AES-128 cryptographic algorithm	Appropriate/Fulfilled
Unit Test	Knowing whether each unit of the system built has met the needs and requirements	Appropriate/Fulfilled
Integration Testing	Ensure the relationship of every part of the system can work properly	Appropriate/Fulfilled
System Test	Knowing whether the prototype that has been built has been able to meet the system requirements following what has been determined both in terms of functional requirements and non-functional requirements	Appropriate/Fulfilled
Two-Factor Authentication Accuracy	Knowing the accuracy of the authentication process using smartcards and T-OTP	Appropriate/Fulfilled

The results of the authentication accuracy test were carried out ten times for each login scenario. The results of the authentication accuracy test are excellent because all the outputs from the scenario are in accordance with the expected conditions. In addition, an encrypted data transmission test was carried out, and the results can be seen in Figure 15. The test was carried out using Wireshark tools to prove that the implementation of the AES-128 algorithm was successful and the data was properly encrypted.

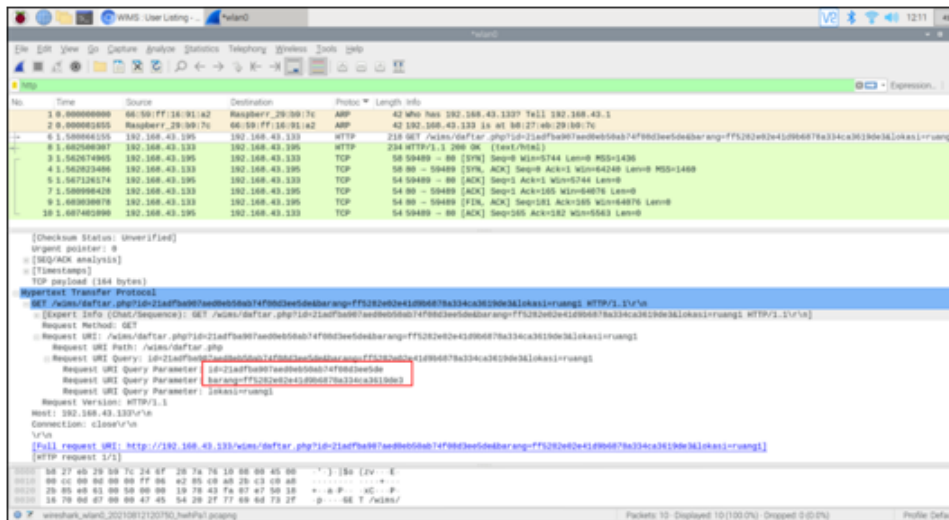


Figure 15. Captured Data Transfer Results on Wireshark Tools

**4. CONCLUSION**

This IoT-based Warehouse Inventory Management System (WIMS) prototype has overcome problems in the conventional warehouse inventory system, namely in the form of an automatic inventory of incoming and outgoing goods. This is due to the success of making modern WIMS by implementing IoT so that the problems of conventional warehouse inventory systems that require workers to process inventory of goods both to be stored and removed from the warehouse manually can be resolved with this system built. To get the WIMS prototype in this research, several stages were carried out, namely server initiation, implementation of MFRC533 on ESP32, implementation of MFRC522 on Raspberry Pi, WIMS development, scanning, sending encryption, and decryption of RFID tags of goods, storing of data on RFID tags of goods in a database, scanning UID User Smartcard, sending T-OTP codes, and creating web applications. In addition to building a WIMS prototype, several tests were carried out to ensure the suitability of system performance. The test results show that the WIMS prototype that has been built is appropriate and fulfilling. Tests were also conducted to determine the accuracy of Two-Factor Authentication. The results show that the level of authentication accuracy is very good. Suggestions that can be given for further research are the addition of features to be able to find out item data that shows specifically the location of the item’s position in each room and the development of features limiting access to the database, by allowing users to only access data in certain rooms as well as adding authentication to the add function. and reduce data on goods in the warehouse.

**5. ACKNOWLEDGEMENTS**

The authors would like to thank Politeknik Siber dan Sandi Negara which has provided infrastructure support during the research and helped fund the publication of this research.

**6. DECLARATIONS**

**AUTHOR CONTRIBUTION**

The ideas, designs, and experimental designs were carried out by the First Author and Second Author, implementation and treatment of tests carried out by the First Author and Second Author, data collection and data analysis were carried out by the first and third authors, script writing by the second and third authors, revision and finalization of the manuscript was carried out by all authors

#### FUNDING STATEMENT

This research is supported by the Center for Research and Community Service of the State Cyber and Crypto Polytechnic in the form of the cost of publishing an accredited national journal through the PPM Budget from the National Cyber and Crypto Agency.

#### COMPETING INTEREST

We have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

#### REFERENCES

- [1] B. S. S. Tejesh and S. Neeraja, "Warehouse Inventory Management System using IoT and Open Source Framework," *Alexandria engineering journal*, vol. 57, no. 4, pp. 3817–3823, 2018.
- [2] A. Mude and L. B. F. Mando, "Implementasi Keamanan Rumah Cerdas Menggunakan Internet of Things dan Biometric System," *Matrik: Jurnal Manajemen, Teknik Informatika, dan Rekayasa Komputer*, vol. 21, no. 1, pp. 179–188, 2021.
- [3] R. de Assis and J. Sagawa, "Assessment of the Implementation of a Warehouse Management System in a Multinational Company of Industrial Gears and Drives," *Gestão & Produção*, vol. 25, no. 2, pp. 370–383, 2018.
- [4] C. K. N. Guptha, M. G. Bhaskar, and V. Meghasree, "Design of IoT Architecture for Order Picking in A Typical Warehouse," in *2018 3rd International Conference on Computational Systems and Information Technology for Sustainable Solutions (CSITSS)*. IEEE, 2018, pp. 50–53.
- [5] G. M. Cidal, Y. A. Cimbek, G. Karahan, Ö. E. Böler, Ö. Özkardesler, and H. Üvet, "A Study on The Development of Semi Automated Warehouse Stock Counting System," in *2019 6th International Conference on Electrical and Electronics Engineering (ICEEE)*. IEEE, 2019, pp. 323–326.
- [6] S. Xu, H. Yu, Y. Yang, and Q. Tan, "A Portable Warehouse Management Terminal Based on Internet of Things," in *2019 IEEE 2nd International Conference on Computer and Communication Engineering Technology (CCET)*. IEEE, 2019, pp. 168–171.
- [7] A. A. Shastri and P. N. Chatur, "Efficient and Effective Security Model for Database Specially Designed to Avoid Internal Threats," in *2015 International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials (ICSTM)*. IEEE, 2015, pp. 165–167.
- [8] R. Andriani, S. E. Wijayanti, and F. W. Wibowo, "Comparision Of AES 128, 192 And 256 Bit Algorithm For Encryption And Description File," in *2018 3rd International Conference on Information Technology, Information System and Electrical Engineering (ICITISEE)*. IEEE, 2018, pp. 120–124.
- [9] A. M. Abdullah, "Advanced Encryption Standard (AES) Algorithm to Encrypt and Decrypt Data," *Cryptography and Network Security*, vol. 16, pp. 1–11, 2017.
- [10] P. Patil, P. Narayankar, D. G. Narayan, and S. M. Meena, "A Comprehensive Evaluation of Cryptographic Algorithms: DES, 3DES, AES, RSA and Blowfish," *Procedia Computer Science*, vol. 78, pp. 617–624, 2016.
- [11] A. Apriani, H. Zakiyudin, and K. Marzuki, "Penerapan Algoritma Cosine Similarity dan Pembobotan TF-IDF System Penerimaan Mahasiswa Baru pada Kampus Swasta," *Jurnal Bumigora Information Technology (BITe)*, vol. 3, no. 1, pp. 19–27, 2021.
- [12] A. Prameshwari and N. P. Sastra, "Implementasi Algoritma Advanced Encryption Standard (AES) 128 untuk Enkripsi dan Dekripsi File Dokumen," *Eksplora Informatika*, vol. 8, no. 2, pp. 52–58, 2018.
- [13] T. S. Rao, P. Pranay, S. Narayana, Y. Reddy, Sunil, and P. Kaur, "ESP32 Based Implementation of Water Quality and Quantity Regulating System," *Atlantis Hilight in Computer Sciences*, vol. 4, pp. 122–129, 2021.
- [14] L. Lumburovska, J. Dobрева, S. Andonov, H. M. Trpcheska, and V. Dimitrova, "Comparative Analysis of HOTP and TOTP Authentication Algorithms. Which one to choose?" *International Scientific Journal "Security & Future"*, vol. V, no. 4, pp. 131–136, 2021.

- 
- [15] I. Permana, M. Hardjianto, and K. A. Baihaqi, "Securing the Website Login System with the SHA256 Generating Method and Time-based One-Time Password (TOTP)," *Systematics*, vol. 2, no. 2, pp. 65–71, 2020.
- [16] B. Vaidya, J. H. Park, S.-S. Yeo, and J. J. P. C. Rodrigues, "Robust One-time Password Authentication Scheme Using Smart Card for Home Network Environment," *Computer Communications*, vol. 34, no. 3, pp. 326–336, 2011.
- [17] H. Rezaeighaleh, R. Laurens, and C. C. Zou, "Secure Smart Card Signing with Time-based Digital Signature," in *2018 International Conference on Computing, Networking and Communications (ICNC)*. IEEE, 2018, pp. 182–187.
- [18] B. H. Wixom and R. M. Roth, *System Analysis and Design*, 4th ed. Wiley Publishing, 2008.

