PROJECT REPORT

SECURE RFID USING DATA ENCRYPTION AND COMPRESSION ALGORITHM

MICHAEL VALENTINO
17.K1.0007

Faculty of Computer Science
Soegijapranata Catholic University
2021

APPROVAL AND RATIFICATION PAG

HALAMAN PENGESAHAN

Judul Tugas Akhir:	:	Secure RFID Using Data Encryption and Compression Algorithm
Diajukan oleh	:	Michael Valentino I K
NIM	:	17.K1.0007
Tanggal disetujui	:	09 Juli 2021
Telah setujui oleh		
Pembimbing	:	R. Setiawan Aji Nugroho S.T., MCompIT., Ph.D
Penguji 1	:	R. Setiawan Aji Nugroho S.T., MCompIT., Ph.D
Penguji 2	:	Hironimus Leong S.Kom., M.Kom.
Penguji 3	:	Rosita Herawati S.T., M.I.T.
Penguji 4	:	Y.b. Dwi Setianto
Penguji 5	:	Yonathan Purbo Santosa S.Kom., M.Sc
Penguji 6	:	Yulianto Tejo Putranto S.T., M.T.
Ketua Program Studi	:	Rosita Herawati S.T., M.I.T.
Dekan	:	R. Setiawan Aji Nugroho S.T., MCompIT., Ph.D

Halaman ini merupakan halaman yang sah dan dapat diverifikasi melalui alamat di bawah ini.

sintak.unika.ac.id/skripsi/verifikasi/?id=17.K1.0007

STATEMENT OF ORIGINALITY

I, the undersigned:

Name : MICHAEL VALENTINO I K

ID : 17.K1.0007

Certify that this project was made by myself and not copy or plagiarize from other people, except that in writing expressed to the other article. If it is proven that this project was plagiarizing or copy the other, I am ready to accept a sanction.

Semarang, July, 9, 2021

MICHAEL VALENTINO

17.K1.0007

APPROVAL PAGE FOR PUBLICATION OF
SCIENTIFIC PAPERS FOR ACADEMIC INTEREST

The undersigned below:

Name	: Michael Valentino I K
Undergraduate Program	: TECHNICAL INFORMATION
Faculty	: COMPUTER SCIENCE
Type of Work	: SKRIPSI

Approved to give Non-Exclusive Royalty Free Right to Soegijapranata Catholic University Semarang for scientific work entitled **"SECURE RFID USING DATA ENCRYPTION AND COMPRESSION ALGORITHM"** along with the existing tools (if needed). With this Non-Exclusive Royalty Free Right to Soegijapranata Catholic University has the right to store, transfer data / format, manage in the form of a database, maintain and publish this final project as long as I keep my name as a writer / creator and as a Copyright owner.

This statement I made in truth

Semarang, July, 9, 2021

Sincerely

MICHAEL VALENTINO

ACKNOWLEDGMENT

First of all thank you to Jesus Christ for his blessing, that I have received a myriad of support, advice, and assistance throughout working on my final project. I would like to thank my supervisor Robertus Setiawan Aji Nugroho, Ph.D. for formulating this topic. Also, thank you to Arcelina Sukiatmo in whose final project works influence me when writing this documentation.

The final project is a requirement to take the Bachelor of Computer Science exam in the Informatics Engineering Study Program at Soegijapranata Catholic University Semarang.

In the preparation and making of this final project, I got support and encouragement from people around me, I would give my special thanks to :

1. My family always support and pray for me while in the making of this final project.
2. My Friends, Steven Raymond and Ivan Albern who helped me work on this document.
3. And other people that I could not mention the name one by one who also giving support and prays for me when making this final project.

Semarang, July, 9, 2021

Sincerely

MICHAEL VALENTINO

ABSTRACT

RFID cards are commonly used nowadays as a security measure to unlock housing doors, such as apartments or hotels. It works in pairs with an RFID reader to receive the data sent through low-power radio waves from the card. The problem is that the data inside the RFID card could easily read by an RFID reader and possibly read by unauthorized people. The RFID card data security could be improved using encryption and compression algorithm.

This study will compare the encryption using RSA (Rivest-Shamir-Adleman) and AES (Advanced Encryption Standard) algorithms implementation combined with the compression algorithm. Furthermore, time and memory usage is measured in different scenarios of implementation on encryption and compression, which one should put at the first stage than the other. Thus, implement the best solution for encryption and compression in the Arduino environment.

This research concluded that the encryption time of AES is significantly faster than RSA. the test result has shown that AES, on average, is 145768 microseconds faster than RSA in encryption without compression. Also, when tested with compression, AES is 134193.45 microseconds faster than RSA. The AES is better suited for the Arduino UNO system because of the lower computation cost. Also, RSA needs a larger key size to be as safe as AES with a smaller key size. Therefore, AES is better overall than RSA when tested for this project.

Keyword: rfid, arduino, rsa, aes, run-length encoding, encryption, compression

TABLE OF CONTENTS

COVER **i**

CHAPTER 1 INTRODUCTION 12

1.1. Background 12

1.2. Problem Formulation 13

1.3. Scope 13

1.4. Objective 13

CHAPTER 2 LITERATURE STUDY 14

CHAPTER 3 RESEARCH METHODOLOGY 18

3.1. Rivest-Adleman-Shamir (RSA) Algorithms 18

3.1.1. Key Generation Algorithm : 18

3.1.2. Encryption : 18

3.1.3. Decryption : 19

3.2. Advanced Encryption Standard (AES) Algorithms 19

3.2.1. Adding Round Key 20

3.2.2. Substitutes Bytes 20

3.2.3. Shift Rows 22

3.2.4. Mix Columns 22

3.2.6. AES Encryption and Decryption Flow Diagram 23

3.3. Huffman Encoding 24

3.4. RSA and AES Implementation on Arduino 27

3.5. Modified Huffman Implementation on Arduino 28

3.6. Compare RSA and AES Encryption without Compression Speeds 28

3.7. Compare the RSA and AES Encryption with Compression Speeds 28

3.8. Testing 28
CHAPTER 4 ANALYSIS AND DESIGN 29
4.1. Analysis 29
4.1.1. RSA, AES, and Modified Huffman Compression in Arduino 29
4.1.2. MFRC522 RFID Reader and Tag 30
4.1.3. Encryption and Compression Speed 31
4.2. Desain 32
4.2.1. Data Encryption and Decryption in Arduino 32
4.2.2. Modified Huffman Compression in Arduino 34
4.2.3. Complete Arduino Module Scheme 35
CHAPTER 5 IMPLEMENTATION AND TESTING 37
5.1. Implementation 37
5.1.1. Code for Compression (Modified Huffman) 37
5.1.2. Code for RSA Encryption 39
5.1.3. Code for AES Encryption 41
5.1.4. Code for Random Compression 41
5.2. Testing 42
5.2.1. RSA Encryption Testing without Compression 42
5.2.2. AES Encryption Testing without Compression 42
5.2.3. Encryption Time without Compression 43
5.2.4. Decryption Time without Compression 45
5.2.5. Combining Encryption with Simple Compression Algorithms 46
5.2.6. Comparison of RSA and AES with Compression Method 48
5.2.7. Compression Effectiveness 50

CHAPTER 6 CONCLUSION 53

References 1

APPENDIX 2

LIST OF FIGURE

Figure 3.1 Adding Round Key 9

Figure 3.2 S-box Table 10

Figure 3.3 Substitution Bytes 21

Figure 3.4 Shift Rows 11

Figure 3.5 Mix Columns 12

Figure 3.6 Mix Column Multiplication 12

Figure 3.6 AES Encryption And Decryption Flowchart 13

Figure 3.7 Binary Tree of 'AAAABBBCCCCCD' 15

Figure 4.1 MFR522 Pin Configuration and Description 19

Figure 4.2 MIFARE RFID Tag Memory Construction 20

Figure 4.3 Flowchart of RSA Program 21

Figure 4.4 Flowchart of AES Program 22

Figure 4.5 Flowchart of Modified Huffman Program 23

Figure 4.6 Complete Arduino Module Scheme 24

Figure 5.1 Compression Test 1 Using Paragraph Input 35

Figure 5.2 Compression Test 2 Using Paragraph Input 36

Figure 5.3 Simple Diagram of Run Length Encoding 36

LIST OF TABLE

Table 5.1. RSA Encryption Testing Comparison 31

Table 5.2. AES Encryption Testing Comparison 32

Table 5.3. RSA and AES Encryption Time Comparison 32

Table 5.4. RSA and AES Decryption Time Comparison 34

Table 5.5. AES and RSA with Compression Comparison 37