VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

School of Information Technology and Engineering

SECURE E-PERSON HEALTH CARE SYSTEM
USING AES

A project submitted
in fulfillment of the requirements for the degree of
M.Tech. (SE)

By

HARISH D (18MIS0136)

AJAY AADHAV V.V (18MIS0163)

NAVEENKUMAR J (18MIS0395)

Course Instructor

Prof. Sudha M

Assistant Professor Grade 1(SG)

June 2021

UNDERTAKING

This is to declare that the project entitled “SECURE E-PERSON HEALTH CARE SYSTEM” is an original work done by undersigned, in partial fulfillment of the requirements for the degree *M.Tech. (Software Engineering)* at School of Information Technology and Engineering, VIT, Vellore.

All the analysis, design and system development have been accomplished by the undersigned.

HARISH D

AJAY AADHAV VV

NAVEENKUMAR J

ABSTRACT

We live in a world where information is available in our fingertips. This has made life easy not only for the everyday consumer but also for healthcare industry as it means that they are able to get the crucial information that they need within a few seconds rather than waste time manually going through a paper trail. We believe that there is a need for a system that provides timely information while at the same time protecting the confidentiality of the patient. The aim is to design a healthcare database management system that ensures that unauthorized personnel can't access it. It should also enable quick and swift access for the authorized entities. It must also have provisions for the patient to check up on his own data. It should also contain a log book of who and all had accessed the data.

Table of Contents

LIST OF FIGURES

1. Introduction *(Heading 1)*

This chapter comprises background of the project, the reasons for taking it, problems addressed by the project and expected outcomes. A good report starts with an introduction to the title of project. The necessary background information is provided to establish context of the project. The motivation and significance of the project should be highlighted. A crisp problem statement is followed by scope of the project along with any limitation or exclusions. The specific objectives to be achieved should be stated. A roadmap or organization of report concludes the chapter.

1.1 Objectives

- Store the patient data in an encrypted format.
- A dean should be able to add authorized personnel(doctors).
- Doctors and dean should be able to view and add patient records hasle freely.
- Patient should be able to view his own record.
- Dean should be able to see which doctors accessed which patient record.

2. Overview

We are using fernet symmetric key encryption as it combines encryption along with message authentication and a time stamp to ensure that the data is protected and that it has not been modified. Fernet uses 128-bit AES in CBC mode, with HMAC using SHA256 for authentication.

2.1 Literature survey:

Paper Title	Author and Year	Proposed Methodology	Advantages	Limitations/challenges
Assessment of Encryption and Decryption Schemes for Secure Data Transmission in Healthcare Systems[1]	Authors: Kazeem B. Adedeji, Nnamdi I. Nwulu, Clinton Aigbavboa and Saheed L. Gbadamosi Year:2019	Elliptic Curve Cryptography (ECC), Rivest Cipher 4 (RC4), Data Encryption Standard (DES), Advanced Encryption Standard (AES)	❖ Robust security protocols ❖ It's hard to hack AES algorithm	AES uses too simple algebraic structure and Every block is always encrypted in the same way.
Smart Secure System For Human Health Monitoring[2]	Nikhil Nair R, Kiran K A Year:2017	AES algorithm in PHP (Hypertext Pre-processor), Raspberry Pi	❖ multi-tasking capability and low power consumption . ❖ low process time, ❖ high throughput and low latency	Performance limitation due to its hardware.
An Efficient Data Security in Medical Report using Block Chain Technology[3]	Mary Subaja Christo, Anigo Merjora A, Partha Sarathy G, Priyanka C and Raj Kumari M Year: 2019	Quantum Cryptography, for Encryption 1.Authentication, 2. Encryption and	❖ AES is fast algorithm with high security ❖ trustworthiness ❖ Secure Authentication	Patient can't modify the patient's medical History

		3. Data Retrieval using Block Chain technology. AES (Advanced Encryption Standard)	❖ Data Retrieval (only Doctors)	
Hybrid security techniques to protect sensitive data in E-healthcare systems[4]	A. Vithya Vijayalakshmi and Dr. L. Arockiam Year:2018	Proposed security technique is based on data obfuscation and encryption technologies	Seamlessly integrated into the information network using intelligent interface.	In IOT , Data Security is one of the major issues (3 types) Data confidentiality, data integrity and data availability Will be compromised.
An Efficient Lightweight Cryptographic Technique For IoT based E-healthcare System[5]	Ravi Raushan Kumar Chaudhary, Kakali Chatterjee Year:2020	Lightweight Block ciphers Technique(AES, DES AND SIMON)	Secure transmission of data of patients from these IoT devices. Remote Health monitoring will save lot of time.	Wearable Devices may not show accurate data because of various factors like not wearing the smartwatch properly and so on..
Complexity of Cyber Security Architecture for IoT Healthcare Industry: A Comparative Study[6]	Aysha K. Alharam, and Wael El-madany Year: 2017	Complexity for cyber security architecture and its application in IoT healthcare industry	❖ AES Encryption algorithm is a symmetric block cipher algorithm. ❖ AES was known for its efficiency and its fast and	Unable to sync the health and wellness information to our personal devices due to cyber security architecture.

			strong algorithm ❖ S-Box Implementation is used along with AES encription	
Cloud-Based E-Health Systems: Security and Privacy Challenges and Solutions[7]	Mohanad Dawoud D.Turgay Altilar Year:2017	Integration of the e-health systems with the cloud computing systems. Using Wireless Body Area Network(WBAN)	❖ Home server is used to communicate between sensor and other devices ❖ Very good in security and privacy protection	Very hard to implement as it requires lot of resources and cost Maintenance cost is very high
The Effects of Cyber-Security on Healthcare Industry[8]	Aysha K. Alharam, and Wael El-madany Year: 2017	cloud-based framework for secure healthcare application using Wireless Body Area Network (WBAN).AES (Advanced Encryption Standard)	AES is fast algorithm with high security	design complexity
A Survey Paper on Internet of Things based Healthcare System[9]	Ms. Shinde Sayali P , Ms. Phalle Vaibhavi N.	IOT based healthcare	❖ Health care system has minimized complication and complexity with the	Breach of privacy since they didn't use any encryption standards.

	Year: 2017		environment of IOT. ❖ Performance , meticulous and economic benefits.	
PPO-CPQ: A Privacy-Preserving Optimization of Clinical Pathway Query for E-Healthcare Systems[10]	Mingwu Zhang , Yu Chen, and Willy Susilo , Senior Member Year: 2020	Clinical pathway development process using patient data Cloud Servers(CS) is used to outsource the medical data.	❖ Better medical treatment. ❖ Privacy protection.	Privacy an still be compromised, because no Encryption is used.
Simplified AES Algorithm for Healthcare Applications on Internet of Thing[11]	Mariam Khader , Marwah Alian , Raghda Hraiz, Sufyan Almajal.Year:2017	IOT based AES Encryption,	❖ low cost ❖ low power consumption	sensor performance limitation
A Security Approach for Health Care Information Systems[12]	Luliana Chiuchisan , Doru-Gabriel Balan ,Oana Geman , Iulian Chiuchisan , Ionel Gordin Year:2017	Security measures and data Communication security	❖ Ensure information protection ❖ ❖ Very good approach for home based security health care	1. Web portal can still be hacked pretending like Physicians or medical staff or web administrator. So lack of security. 2. Not good for commercial use

Security Management in Health Care Information Systems[13]	Berglind Fjola Smaradottir Year:2017	control access and authentication	High level security mechanism to protect data.	Dean only have access to all modules.
An Empirical Study on the Data Security and Privacy Awareness to Use Health Care Wearable Devices[14]	Chen Yang, Tingting Liu, Lulu Zuo, Zhiyong Hao Year:2019	structural equation analysis[Security knowledge (SK), Security attitude (SAT), Security practice (SP), Security awareness (SAW) and Security conduct (SC)]	Multilayer security	They did analysis onlyon small sample amount size data.
Towards Secure and Smart Healthcare in Smart Cities Using Blockchain [15]	Jinglin Qiu, Xueping Liang, Sachin Shetty Year:2018	Blockchain	❖ Patient privacy ❖ patient confidentiality maintained.	User evaluations should be made both in laboratory and real hospital environments to study the usability of access control solutions and how they impact on clinical work processes.
A HYBRID DATA ACCESS CONTROL USING AES AND RSA FOR ENSURING PRIVACY IN ELECTRONIC HEALTHCARE RECORDS[16]	S. Kanaga Suba Raja, A.Sathya, Dr.L.Priya Year: 2020	RSA	❖ Data owners encrypt their data under the relevant access policies prior to outsourcing the (encrypted) data to a Industrial cloud.	RSA algorithm can be very slow in cases where large data needs to be encrypted by the same computer.

			❖ The privacy of the data can be preserved with less computational cost.	
Monitoring Health Care System using Internet of Things - An Immaculate Pairing[17]	Veena Tripathi, Faizan Shakeel Year: 2017	Internet of things	provide it for an improved quality of care, leading to better clinical outcomes, Using Internet of Things (IOT), patient conditions are obtained and stored for further analysis and consultation	Lagging of standard compatibility. More opportunities for failure.
A Security Approach for Health Care Information Systems[18]	Iuliana Chiuchisan, Doru-Gabriel Balan , Oana Geman, Iulian Chiuchisan , Ionel Gordin Year:2017	Encryption on electronic health records (EHR), enabling the access via the Internet.	All electronic medical records should be protected through ownership controlled encryption, enabling transmission, access, and secure storage; the maintenance of electronic information should preserve the content authenticity, patient privacy, and data integrity;	Support for multiple protocols makes this type of security vulnerable.

Development of a Security Layer in a Mobile Health System[19]	Karina Lopez-Landa Saul Dom ınguez-Isidro Yesenia Hernandez-Vel ´ azquez ´ and Eduardo Lopez-Dom ´ ´ınguez Year:2018	Symmetric encryption algorithm, Advanced encryption standard	establishes the functional objectives and functionalities that the products of Electronic Clinical File Systems	Data is close to or contained within the point of vulnerability.
Feature based Encryption for Data Privacy and access control for medical application[20]	Mythri G Year:2017	Attitude based encryption, Advanced encryption standard	distributing secured data among multiple user, owner and multiple authority scenarios	Real time application is difficult through using ABE.
Protection of Patient's Privacy and Data Security in E-Health Services[21]	Yi Hong, , Timothy B. Patrick, Rick Gillis Year:2018	Combination of a normal HTTP interaction over an encrypted Secure Sockets Layer (SSL) or Transport Layer Security (TLS) transport mechanism.	Multilayer security	It provides the appropriate cryptographic tools for secure hierarchical access to healthcare data. This ensures that the access of various entities to the healthcare data is accurately and hierarchically controlled.
The Evaluation Report of SHA-256 Crypt Analysis Hash Function[22]	A.Arul Lawrence Selvakumar ,C.Suresh Ganandhas Year:2009	SHA-256 Crypt Analysis Hash Function	❖ it's a one-way function that is deterministic, fast to compute, resistant to pre-image and second-	FORK 256, which is designed to be not only secure but also fast than SHA-256

			preimage attacks, and is collision resistant.	
Design of High-Throughput SHA-256 Hash Function based on FPGA[23]	Shamsiah binti Suhaili,Takahiro Watanabe Year:2017	SHA-256 Hash Function based on FPGA	❖ Better ❖ performance on Arria II GX in terms of throughput. The high ❖ throughput of SHA-256 unfolding design was obtained at a data ❖ transfer speed of 2429.52 Mbps.	There is technically a limit, but it's quite large. The padding scheme used for SHA-256 requires that the size of the input (in bits) be expressed as a 64-bit number. Therefore, the maximum size is (264-1)/8 bytes ~= 2'091'752 terabytes. The power consumption is more and programmers do not have any control on power optimization in FPGA.
FERNET SYMMETRIC ENCRYPTION METHOD to GATHER MQTT E2E SECURE COMMUNICATIONS for IoT DEVICES[24]	EL GAABOURI Ismail , CHAHBOUN Asaad , and RAISSOUNI Naoufal Year:2020	FERNET SYMMETRIC ENCRYPTION	Fernet guarantees that a message encrypted using it cannot be manipulated or read without the key.	complete message contents must be available in memory, making Fernet generally unsuitable for very large files at this time.

Cryptographic Hash Functions: A Review[25]	Rajeev Sobti , G.Geetha Year:2012	Cryptography, Hash function, compression function	It is computationally easy to compute the hash value of any given message. 1. Message integrity: It is not possible to modify the message without modifying the message digest. 2. Collision resistance: It is infeasible to find two distinct messages that generate the same digest	The disadvantages of hash tables include the fact that databases can degrade if they go through a large number of collisions.

2.2Gap Identified

- The system should allow all medical professionals who need access to the document a way to access the medical details of a patient without issue. The system should also record who and all has viewed the system. It should have a mechanism to check if the data has been modified to ensure integrity of the document. The use of a asymmetric encryption is recommended.

- The main security concerns in personal clinical information are authenticating the sender and receiver, to establish audit trails, to ensure integrity of message along with its confidentiality and to make sure no unauthorized modification occurs. It also talks about various tools we can use to achieve this including smart cards, biometric solutions and public key encryptions. Another major issue is the need to have up to date documents. The changing definition of privacy is also a key factor with regard to patients consent.
- The paper stresses the need that the patient must hold control over the decision about which information is shared with whom. It also discusses that the audit must note down who accessed the document along with a timestamp. The systems must also have a backup to ensure that if anything fails, information critical to save a patient's life is still available.
- The paper discusses the need to use digital signatures to verify authenticity. It also supports that there must be an override switch to ensure that during emergencies information is available
- The system must ensure that it protects the user from identity, membership and attribute disclosure. It must also ensure that the bit size is same for each document, so that you cannot link a document to a personal.
- Genetic information should be given higher priority for protection. The level of security should also depend on the type of data.
- The system could be made into a cloud-based system. The access can be role based, each personal is given access to only a part of the electronic health record. Firewalls can be installed to make sure that unauthorized persons are able to implant false information inside the database.

2.3 Entities

Entities in the system:

- Dean
- Doctors
- Patients

Storage type:

The data is stored in encrypted binary files

Identity verification mechanism:

The dean and the doctors will have their passwords stored as hashes to prevent anyone who looks at the doctor’s database to be able to determine their password. Each doctor's’ record will contain his ID and the hash of his password.

The patient identity is verified by his/her knowledge of their patient ID number and birthdate

Health record:

The health record of a patient will contain the following details

- ID number
- Name
- Blood type
- Gender
- Age
- Date of birth

- Height
- Weight
- Allergies
- Medications he/she is on
- Medical conditions he/she has
- Pathological test report
- Phone number
- Emergency phone number
- Remarks
- Details regarding who added the patient to the database and at what time he/she added

Logbook:

A log book will be maintained that records the details of who accessed which patient's record at what time. This is available only to the Dean of medicine.

2.4 Flowchart

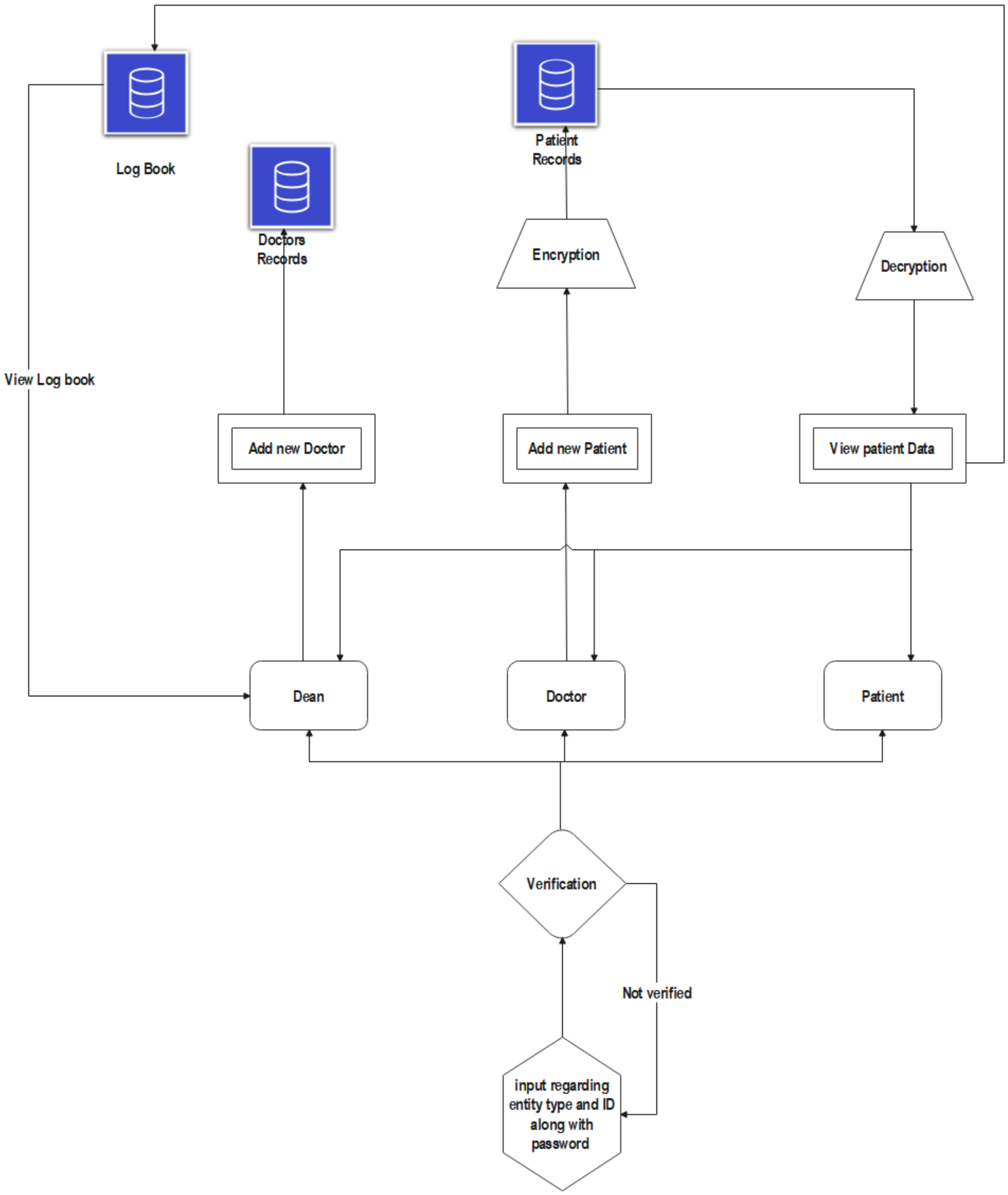


Figure 1 - Flowchart

3. Algorithms and Calculations

3.1 AES

The encryption phase of AES can be broken into three phases: the initial round, the main rounds, and the final round. All of the phases use the same sub-operations in different combinations as follows:

Initial Round

- AddRoundKey

Main Rounds

- SubBytes
- ShiftRows
- MixColumns
- AddRoundKey

Final Round

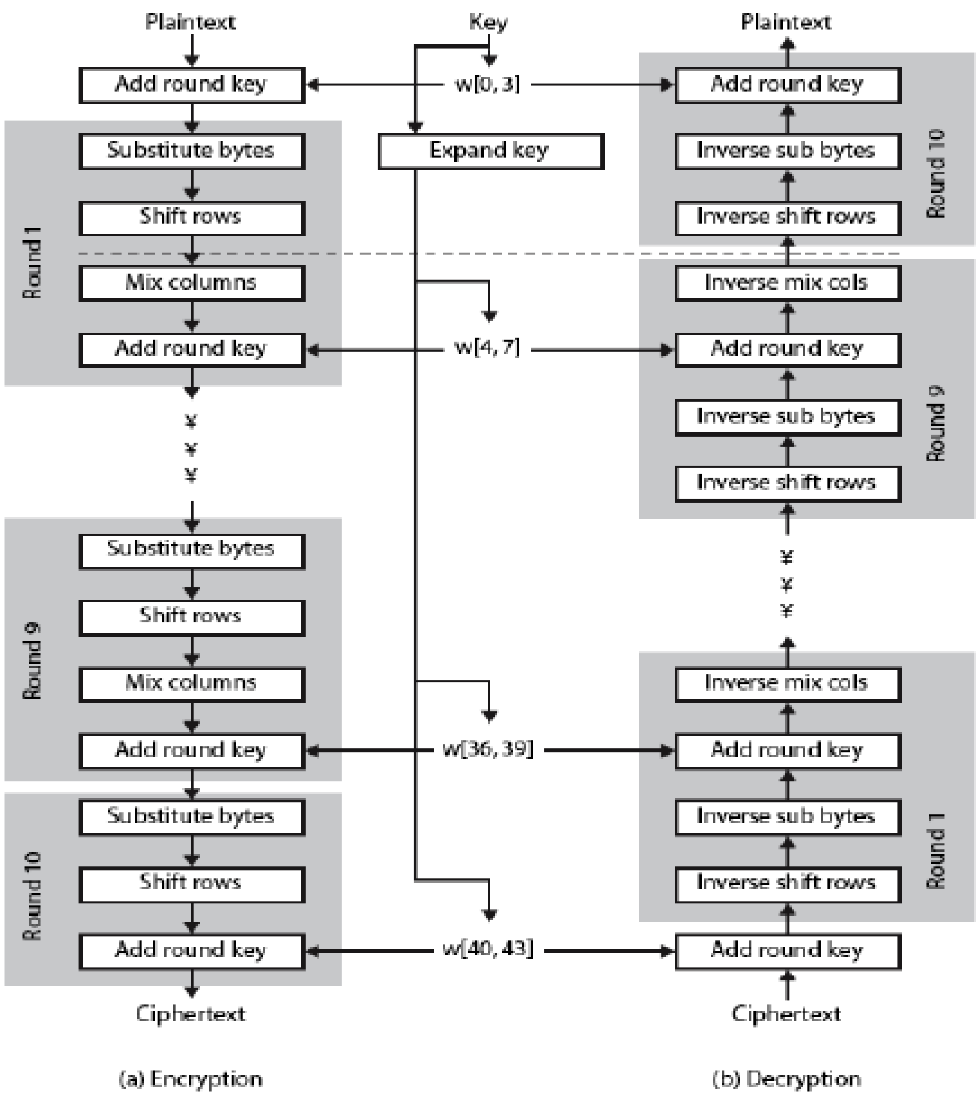
- SubBytes
- ShiftRows
- AddRoundKey

AES is an iterative instead of Feistel cipher. It is based on two common techniques to encrypt and decrypt data knowns as substitution and

permutation network (SPN). SPN is a number of mathematical operations that are carried out in block cipher algorithms [7]. AES has the ability to

deal with 128 bits (16 bytes) as a fixed plain text block size. These 16 bytes are represented in 4x4 matrix and AES operates on a matrix of bytes. In

addition, another crucial feature in AES is number of rounds. The number of rounds is relied on the ength of key. There are three different key sizes are used by AES algorithm to encrypt and decrypt data such as (128, 192 or 256 bits). The key sizes decide to the number of rounds such as AES uses 10 rounds for 128-bit keys, 12 rounds for 192-bit keys and 14 rounds for 256-bit keys.



(a) Encryption

(b) Decryption

Figure 2 – AES Algorithm

AES algorithm uses a particular structure to encrypt data to provide the best security. To do that it relies on a number of rounds and inside each round comprise of four sub-process. Each round consists of the following four steps to encrypt 128-bit block.

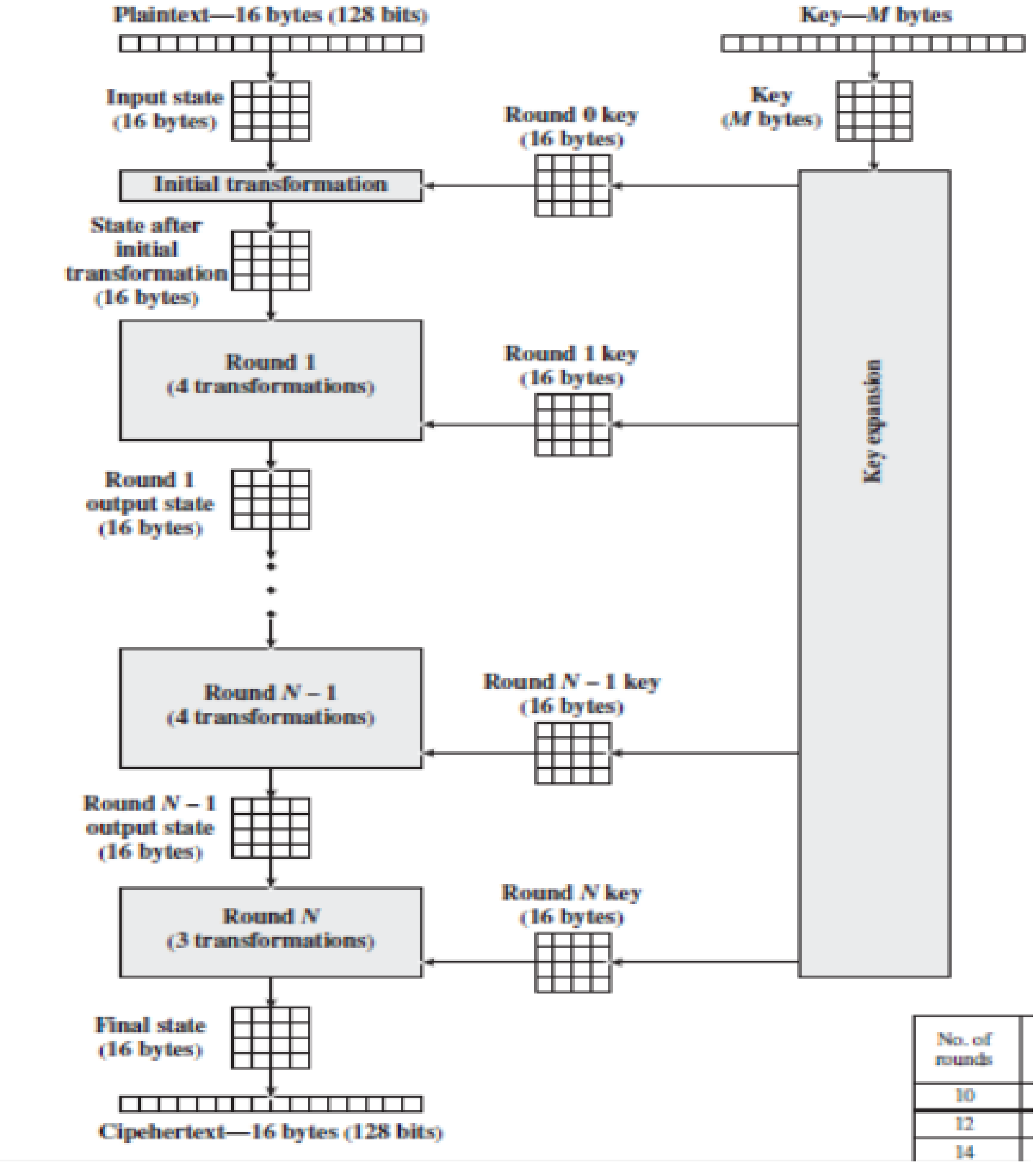

Plaintext—16 bytes (128 bits)
Key—M bytes
Input state (16 bytes)
Key (M bytes)
Round 0 key (16 bytes)
Initial transformation
State after initial transformation (16 bytes)
Round 1 (4 transformations)
Round 1 key (16 bytes)
Key expansion
Round 1 output state (16 bytes)
Round N – 1 (4 transformations)
Round N – 1 key (16 bytes)
Round N – 1 output state (16 bytes)
Round N (3 transformations)
Round N key (16 bytes)
Final state (16 bytes)
Cipehertext—16 bytes (128 bits)
No. of rounds
10
12
14

Figure 3 – AES Encryption

The decryption processes of an AES is similar to the encryption process in the reverse order and both sender and receiver have the same key to encrypt and decrypt data. The last round of a decryption stage consists of three stages such as InvShiftRows, InvSubBytes, and AddRoundKey.

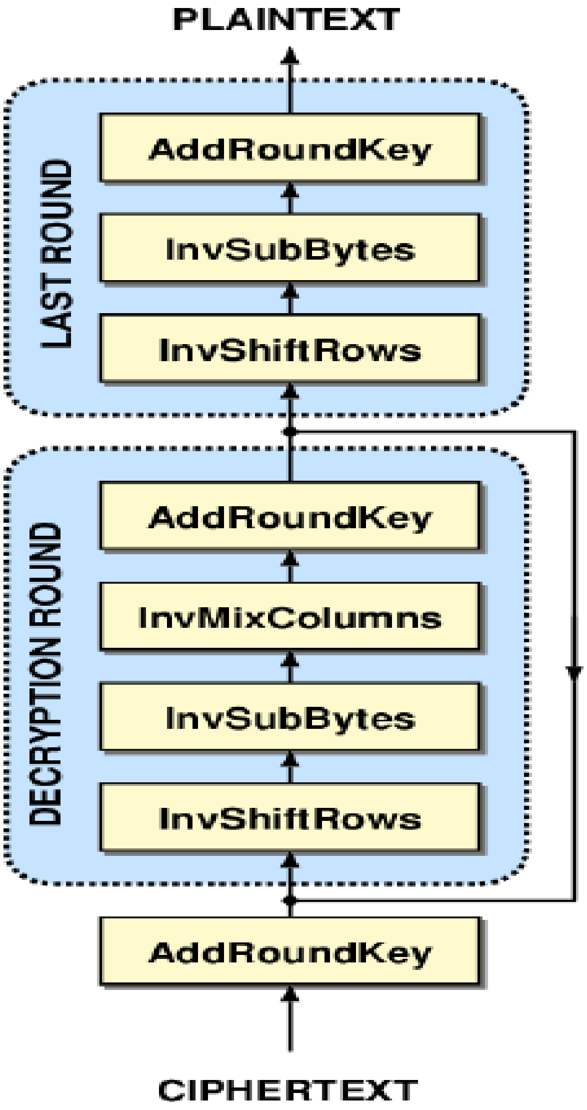


Figure 4 – AES Decryption

3.2 SHA 216

The SHA-256 compression function operates on a 512-bit message block and a 256- bit intermediate hash value. It is essentially a 256-bit block cipher algorithm which encrypts the intermediate hash value using the message block as k.

Computation of the hash of a message begins by preparing the message:

1. Pad the message in the usual way: Suppose the length of the message M, in bits, is `. Append the bit \1" to the end of the message, and then k zero bits, where k is the smallest non-negative solution to the equation `+1+k 448 mod 512. To this append the 64-bit block which is equal to the number ` written in binary. For example, the (8-bit ASCII) message \abc" has length 8 3 = 24 so it is padded with a one, then 448 (24 + 1) = 423 zero bits, and then its length to become the 512-bit padded message. Length of the padded message should now be a multiple of 512 bits

2. Parse the message into N 512-bit blocks M(1); M(2) ;:::;M(N) . The rest 32 bits of message block i are denoted M(i) 0 , the next 32 bits are M(i) 1 , and so on up to M(i) 15 . We use the big-endian convention throughout, so within each 32-bit word, the left-most bit is stored in the most significant bit position

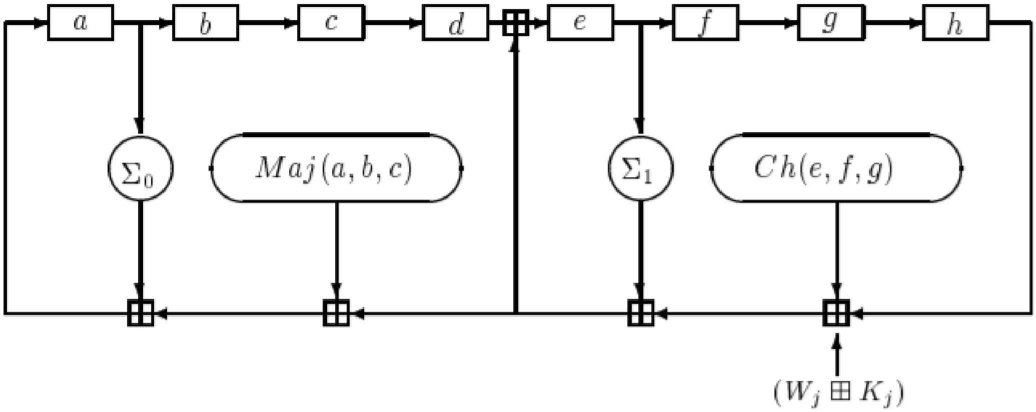


Figure 5 – SHA256 Compression Function

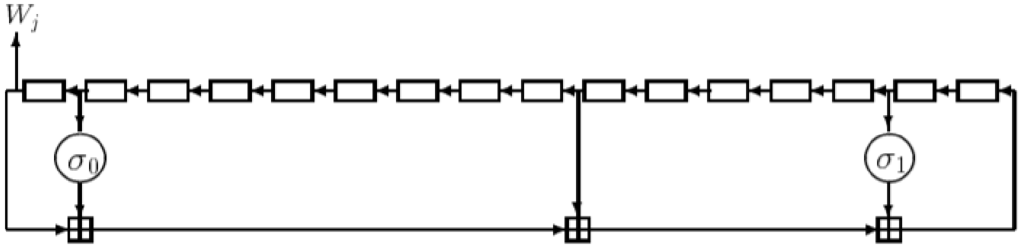


Figure 6 – SHA256 Message Schedule

3.3 HASHLIB

Python hashlib hashing function takes variable length of bytes and converts it into a fixed length sequence. This is a one-way function. That means, you hash a message, you get a fixed length sequence. But you cannot get the original message from those fixed length sequence.

Python secure hash values are used in storing password in encrypted form. So even the application owner won't have access to user password, passwords are matched when user enters the password again and hash value is calculated and compared with the stored value.

4. Conclusion

We were able to create a program that could protect a patient's privacy while at the same time ensuring the information is available to the required personnel when required. It is vulnerable if the source code is exposed and modified.

Improvements

- Cloud based server development
- Mechanism to provide data for research without revealing patient's personal data
- Giving dean the power to re-assign a patient to a new doctor

5. Team Members Contribution

Register Number	Name	Contribution / Role in this Project
18MIS0136	HARISH D	Modules design, algorithms implementation and cryptographic functions in python
18MIS0163	AJAY AADHAV VV	Dean module and remaining system construction and documentation
18MIS0395	NAVEENKUMAR J	Patient and doctor module construction and documentation

References

1. An Efficient Lightweight Cryptographic Technique For IoT based E-healthcare System Ravi Raushan Kumar Chaudhary and Kakali Chatterjee ,Computer Science & Engineering National Institute of Technology Patna, Year:2020[5]
2. Complexity of Cyber Security Architecture for IoT Healthcare Industry: A Comparative Study Aysha K. Alharam, and Wael El-madany Computer Engineering department University of Bahrain, Year:2017[6]
3. A Security Approach for Health Care Information Systems Iuliana Chiuchisan , Doru-Gabriel Balan , Oana Geman , Iulian Chiuchisan , Ionel Gordin , Computers, Electronics and Automation Department, Stefan cel Mare University of Suceava, Suceava, Romania, Year:2017[12]
4. Cloud-Based E-Health Systems: Security and Privacy Challenges and Solutions Mohanad Dawoud and D.Turgay Altilar, Computer Engineering Department Istanbul Technical University Istanbul, TURKEY [7]

5.An Efficient Data Security in Medical Report using Block Chain Technology, Authors: Mary Subaja Christo, Anigo Merjora A, Partha Sarathy G, Priyanka C and Raj Kumari M , Year: 2019[3]

6.PPO-CPQ: A Privacy-Preserving Optimization of Clinical Pathway Query for E-Healthcare Systems Mingwu Zhang , Yu Chen, and Willy Susilo , Senior Member, IEEE, Year:2020[10]

7.Hybrid security techniques to protect sensitive data in E-healthcare systems A. Vithya Vijayalakshmi and Dr. L. Arockiam, Year: 2019.[4]

8)The Effects of Cyber-Security on Healthcare Industry . Publisher: IEEE , Authors: Aysha K. Alharam, and Wael El-madany. Year: 2017[8]

9)Simplified AES Algorithm for Healthcare Applications on Internet of Thing, 2017 8th International Conference on Information Technology (ICIT). Authors: Mariam Khader , Marwah Alian , Raghda Hraiz, Sufyan Almajali[11]

10)Smart Secure System For Human Health Monitoring. International Conference on Intelligent Computing and Control Systems ICICCS 2017. Authors: Nikhil Nair R, Kiran K [2]

11)Security Management in Health Care Information Systems, A literature review. 2017 International Conference on Computational Science and Computational Intelligence. Authors: Berglind Fjola Smaradottir[13]

12)An Empirical Study on the Data Security and Privacy Awareness to Use Health Care Wearable Devices. Publisher: IEEE Authors: Chen Yang, Tingting Liu, Lulu Zuo, Zhiyong Hao. Year:2019[14]

13)A Survey Paper on Internet of Things based Healthcare System. Authors: Ms. Shinde Sayali P. , Ms. Phalle Vaibhavi N. Year:2017 [9]

14)Assessment of Encryption and Decryption Schemes for Secure Data Transmission in Healthcare Systems. Authors: Kazeem B. Adedeji, Nnamdi I. Nwulu, Clinton Aigbavboa and Saheed L. Gbadamosi[1]

15) Towards Secure and Smart Healthcare in Smart Cities Using Blockchain Jinglin Qiu, Xueping Liang, Sachin Shetty

Year:2018[15]

16)A HYBRID DATA ACCESS CONTROL USING AES AND RSA FOR ENSURING PRIVACY IN ELECTRONIC HEALTHCARE RECORDS S. Kanaga Suba Raja, A.Sathya, Dr.L.Priya ,Year: 2020[16]

17)Monitoring Health Care System using Internet of Things - An Immaculate Pairing Veena Tripathi,Faizan Shakeel, Year: 2017[17]

18)A Security Approach for Health Care Information Systems Iuliana Chiuchisan, Doru-Gabriel Balan , Oana Geman, Iulian Chiuchisan , Ionel Gordin ,Year: 2017[18].

19)Development of a Security Layer in a Mobile Health System Karina Lopez-Landa Saul Dom Inguez-Isidro Yesenia Hernandez-Vel ´ azquez ´ and Eduardo Lopez-Dom ´ ´ınguez Year: 2018[19]

20)Feature based Encryption for Data Privacy and access control for medical application Mythri G: Year:2017[20]

21)Protection of Patient's Privacy and Data Security in E-Health Services Yi Hong, , Timothy B. Patrick, Rick Gillis, Year:2018[21]

22)FERNET SYMMETRIC ENCRYPTION METHOD to GATHER MQTT E2E SECURE COMMUNICATIONS for IoT DEVICES,Ismail el GaabouriIsmail el GaabouriChahboun AsaadChahboun Asaad,November 2020[24]

23)Cryptographic Hash Functions: A Review. Rajeev SobtiRajeev SobtiGeetha GanesanGeetha Ganesan[25]

24)Design of High-Throughput SHA-256 Hash Function based on FPGA .Shamsiah binti Suhaili,Takahiro Watanabe[23]

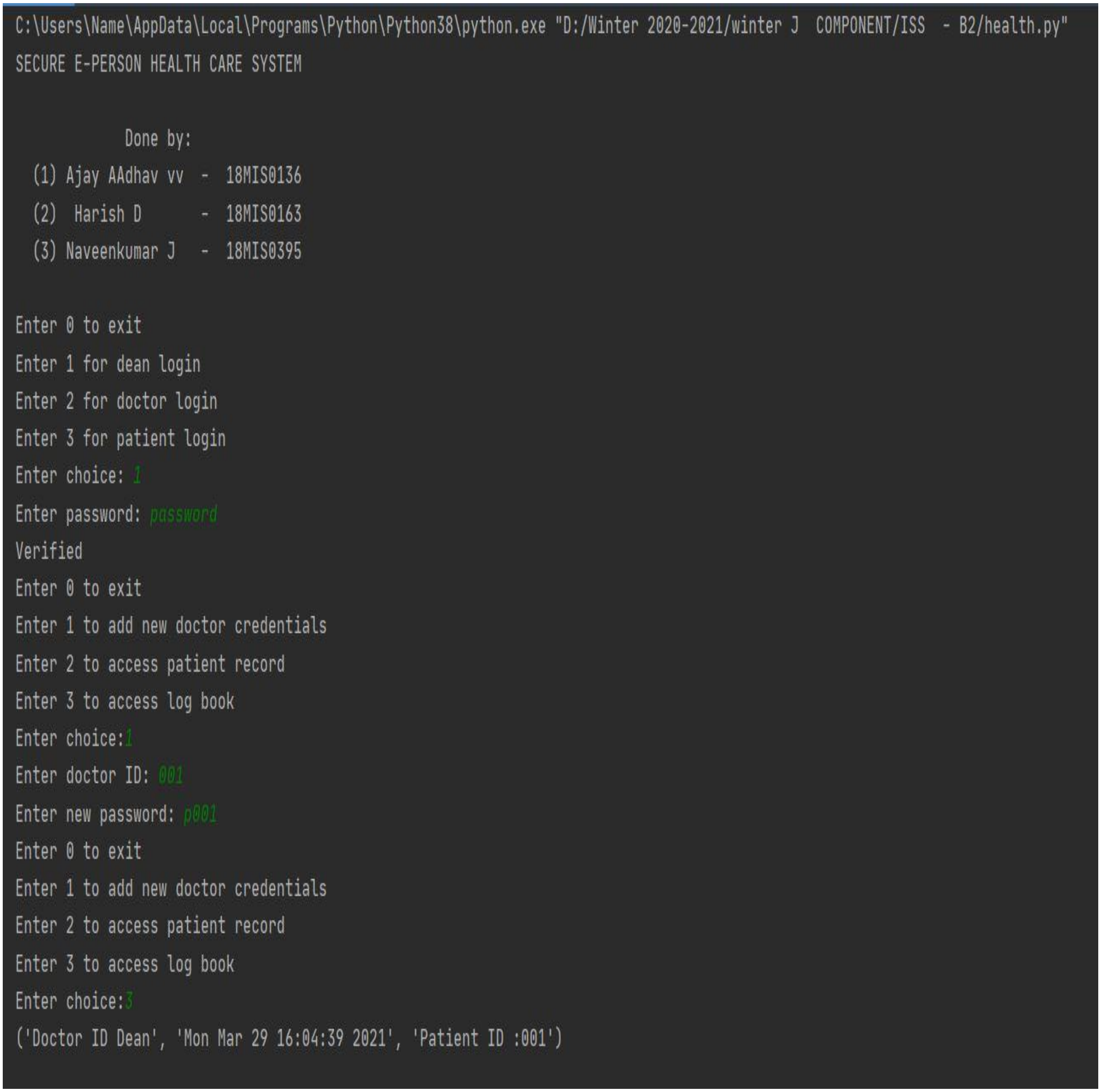
25)The Evaluation Report of SHA-256 Crypt Analysis Hash Function. A.Arul Lawrence Selvakumar ,C.Suresh Ganandhas
Year:2009 [22]

Code:

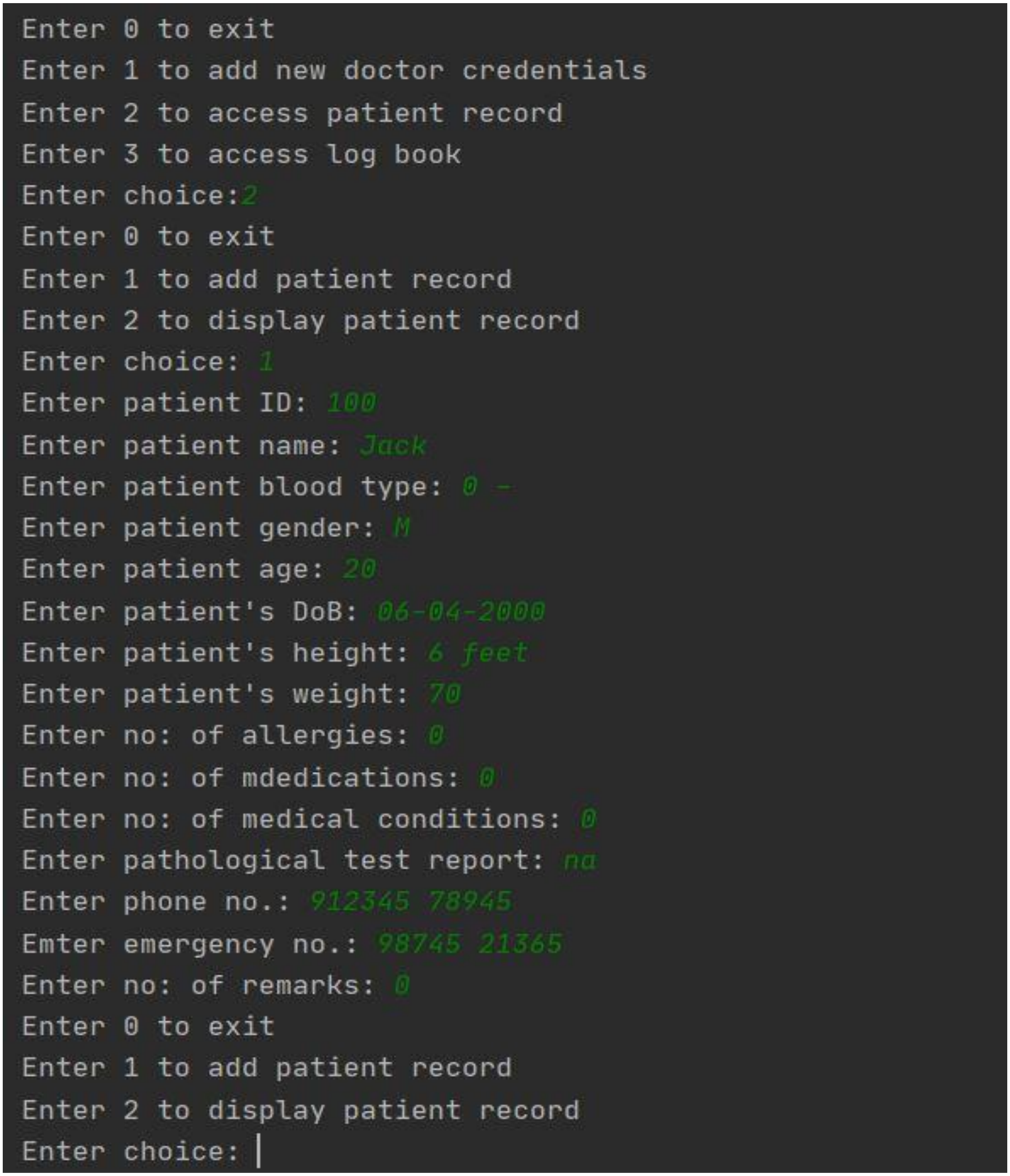
```
import os,pickle,hashlib
import time
from cryptography.fernet import Fernet as aes
print("SECURE E-PERSON HEALTH CARE SYSTEM")
print()
print("            Done by: ")
print("  (1) Ajay AAdhav vv  -  18MIS0136")
print("  (2)  Harish D       -  18MIS0163")
print("  (3) Naveenkumar J   -  18MIS0395")
print()

def pat(T):
 choice=1
 while(choice!=0):
   print("Enter 0 to exit")
   print("Enter 1 to add patient record")
   print("Enter 2 to display patient record")
   choice=int(input("Enter choice: "))
   if choice==1:
      p1=medRecord()
      p1.insRecord()
      f=open("Patient records.bin","ab")
      pickle.dump(p1,f)
      f.close()
   elif choice==2:
      pid=input("Enter patient ID: ")
      f=open("Patient records.bin","rb")
      try:
         while True:
                p1=pickle.load(f)
                if cipher.decrypt(p1.pid).decode()==pid:
                    break
        except EOFError:
            print("Patient doesn't exist")
        else:
            p1.printRec()
            g=open("logbook.bin","ab")
            q=("Doctor ID " +str(T),str(time.asctime(time.localtime(time.time()))),"Patient ID :" + str(pid))
            pickle.dump(q,g)
            g.close()
fernet_key=b'rrm-9Rx_5eeVLJQRehibrO_AwjazFV_mEb7RrzcHans='
cipher=aes(fernet_key)
#patient
class medRecord:
    def __init__(self):
        self.pid=""
        self.name=""
        self.btype=""
        self.gender=""
        self.age=0
        self.dob=""
        self.height=0
        self.weight=0
        self.allergies=[]
        self.medications=[]
        self.conditions=[]
        self.pTestRep=""
        self.phone=""
        self.emerno=""
        self.remarks=[]
        self.time=""
    def insRecord(self):
        self.pid=cipher.encrypt(input("Enter patient ID: ").encode())
        self.name=cipher.encrypt(input("Enter patient name: ").encode())
        self.btype=cipher.encrypt(input("Enter patient blood type: ").encode())
        self.gender=cipher.encrypt(input("Enter patient gender: ").encode())
        self.age=cipher.encrypt(input("Enter patient age: ").encode())
        self.dob=cipher.encrypt(input("Enter patient's DoB: ").encode())
        self.height=cipher.encrypt(input("Enter patient's height: ").encode())
        self.weight=cipher.encrypt(input("Enter patient's weight: ").encode())
        n=int(input("Enter no: of allergies: "))
        for i in range(0,n):
            self.allergies+=[cipher.encrypt(input("Enter allergy: ").encode())]
        n=int(input("Enter no: of mdedications: "))
        for i in range(0,n):
            self.medications+=[cipher.encrypt(input("Enter medication: ").encode())]
        n=int(input("Enter no: of medical conditions: "))
        for i in range(0,n):
            self.conditions+=[cipher.encrypt(input("Enter medical condition: ").encode())]
        self.pTestRep=cipher.encrypt(input("Enter pathological test report: ").encode())
        self.phone=cipher.encrypt(input("Enter phone no.: ").encode())
        self.emerno=cipher.encrypt(input("Emter emergency no.: ").encode())
        n=int(input("Enter no: of remarks: "))
        for i in range(0,n):
            self.remarks+=[cipher.encrypt(input("Enter remarks: ").encode())]
        self.time=cipher.encrypt(str(time.asctime(time.localtime(time.time()))).encode())
    def printRec(self):
        print("\n")
        print("Patient ID: ",cipher.decrypt(self.pid).decode())
        print("Patient name: ",cipher.decrypt(self.name).decode())
        print("Patient blood type: ",cipher.decrypt(self.btype).decode())
        print("Patient gender: ",cipher.decrypt(self.gender).decode())
        print("Patient age: ",cipher.decrypt(self.age).decode())
        print("Patient's DoB: ",cipher.decrypt(self.dob).decode())
        print("Patient height: ",cipher.decrypt(self.height).decode())
        print("Patient weight",cipher.decrypt(self.weight).decode())
        print("Patient allergies:")
        for i in self.allergies:
            print("\t-",cipher.decrypt(i).decode())
        print("Patient medications:")
        for i in self.medications:
            print("\t-",cipher.decrypt(i).decode())
        print("Patient medical conditions:")
        for i in self.conditions:
            print("\t-",cipher.decrypt(i).decode())
        print("Pathological test report: ",cipher.decrypt(self.pTestRep).decode())
        print("Patient phone no.: ",cipher.decrypt(self.phone).decode())
        print("Patient emergency no.: ",cipher.decrypt(self.emerno).decode())
        print("Remarks:")
        for i in self.remarks:
            print("\t-",cipher.decrypt(i).decode())
        print("Patient since : " ,cipher.decrypt(self.time).decode())
        print("\n")

#Doctor class
class doc:
    def _init_(self):
        self.did=""
        self.hash=""
    def insrec(self):
        self.did=input("Enter doctor ID: ")
        P1=input("Enter new password: ")
        self.hash=((hashlib.sha256(P1.encode())).hexdigest())
#Dean hash
```

```
DH="5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8"
c0=1
while c0!=0:
    print("Enter 0 to exit")
    print("Enter 1 for dean login")
    print("Enter 2 for doctor login")
    print("Enter 3 for patient login")
    c0=int(input("Enter choice: "))
    #Dean
    if c0==1:
        P=input("Enter password: ")
        deanh=((hashlib.sha256(P.encode())).hexdigest())
        #Dean basic
        if deanh==DH:
            print("Verified")
            c1=1
            while c1!=0:
                print("Enter 0 to exit")
                print("Enter 1 to add new doctor credentials")
                print("Enter 2 to access patient record")
                print("Enter 3 to access log book")
                c1=int(input("Enter choice:"))
                if c1==2:
                    pat("Dean")
                elif c1==3:
                    w=open("logbook.bin","rb")
                    try:
                        while True:
                            e=pickle.load(w)
                            print(e)
                    except EOFError:
                        w.close()
                        print()
                elif c1==1:
                    d1=doc()
                    d1.insrec()

                    f=open("Doctor records.bin","ab")
                    pickle.dump(d1,f)
                    f.close()
        else:
            print("Access denied")
    #Doctor
    if c0==2:
        did=input("Enter doctor ID: ")
        f=open("Doctor records.bin","rb")
        dip=input("Enter password :")
        dih=((hashlib.sha256(dip.encode())).hexdigest())
        try:
            while True:
                p1=pickle.load(f)
                if p1.did==did:
                    if p1.hash==dih:
                        print("Verified")
                        pat(str(did))
                    else:
                        print("Access denied")
                    break


        except EOFError:
            print("Doctor does not exist")
    #Patient
    if c0==3:
        pid=input("Enter patient ID: ")
        pdob=input("Enter date of birth : ")
        f=open("Patient records.bin","rb")
        try:
            while True:
                p1=pickle.load(f)
                if cipher.decrypt(p1.pid).decode()==pid:
                    if cipher.decrypt(p1.dob).decode()==pdob:
                        p1.printRec()
                        break
                    else:
                        print("Access denied")
                        break
        except EOFError:
            print("Patient doesn't exist")
    print("\n")
```

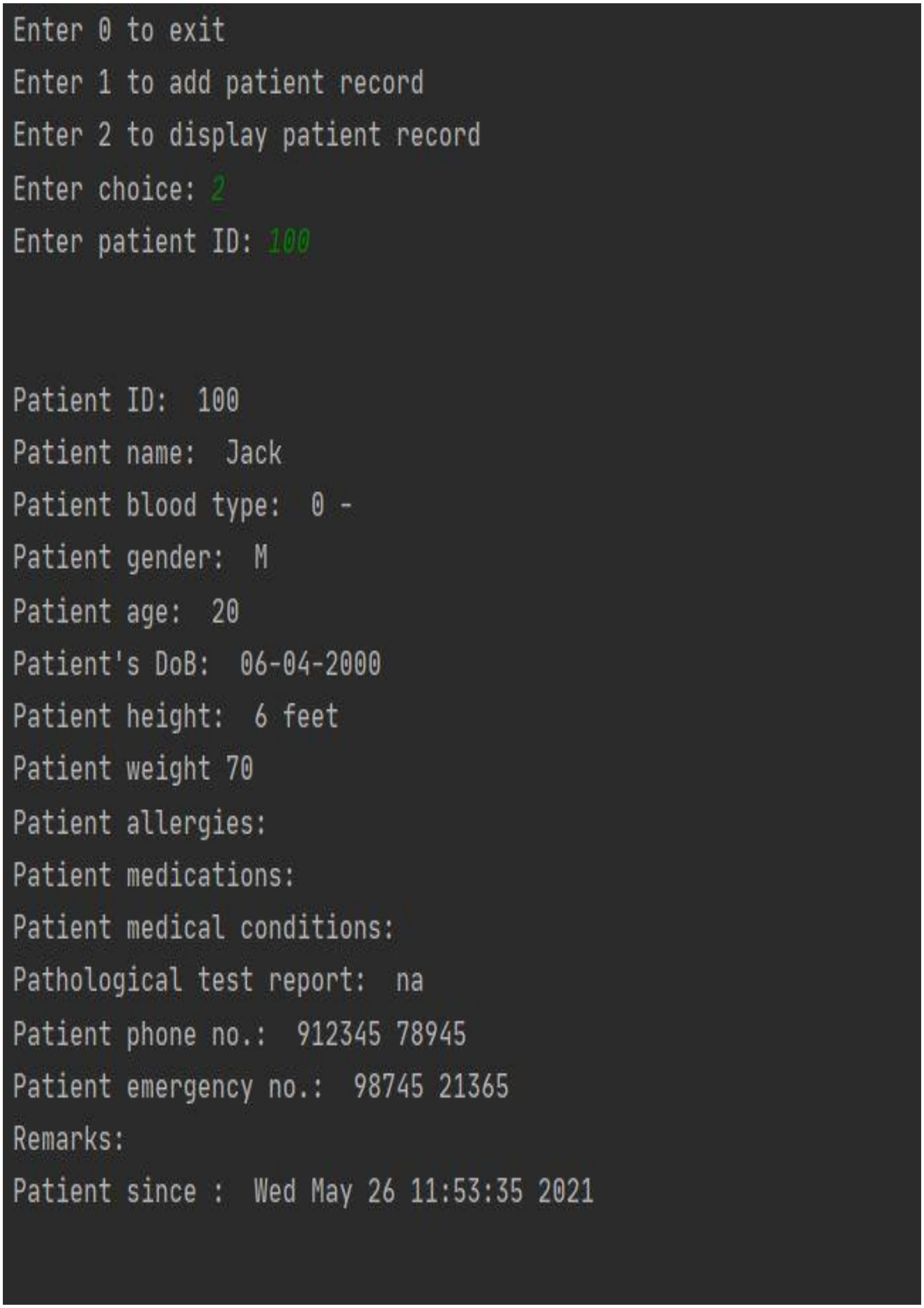
OUTPUT:



```
C:\Users\Name\AppData\Local\Programs\Python\Python38\python.exe "D:/Winter 2020-2021/winter J  COMPONENT/ISS  - B2/health.py"
SECURE E-PERSON HEALTH CARE SYSTEM


             Done by:
  (1) Ajay AAdhav vv  -  18MIS0136
  (2)  Harish D       -  18MIS0163
  (3) Naveenkumar J   -  18MIS0395

Enter 0 to exit
Enter 1 for dean login
Enter 2 for doctor login
Enter 3 for patient login
Enter choice: 1
Enter password: password
Verified
Enter 0 to exit
Enter 1 to add new doctor credentials
Enter 2 to access patient record
Enter 3 to access log book
Enter choice:1
Enter doctor ID: 001
Enter new password: p001
Enter 0 to exit
Enter 1 to add new doctor credentials
Enter 2 to access patient record
Enter 3 to access log book
Enter choice:3
('Doctor ID Dean', 'Mon Mar 29 16:04:39 2021', 'Patient ID :001')
```



```
Enter 0 to exit
Enter 1 to add new doctor credentials
Enter 2 to access patient record
Enter 3 to access log book
Enter choice:2
Enter 0 to exit
Enter 1 to add patient record
Enter 2 to display patient record
Enter choice: 1
Enter patient ID: 100
Enter patient name: Jack
Enter patient blood type: 0 -
Enter patient gender: M
Enter patient age: 20
Enter patient's DoB: 06-04-2000
Enter patient's height: 6 feet
Enter patient's weight: 70
Enter no: of allergies: 0
Enter no: of mdedications: 0
Enter no: of medical conditions: 0
Enter pathological test report: na
Enter phone no.: 912345 78945
Emter emergency no.: 98745 21365
Enter no: of remarks: 0
Enter 0 to exit
Enter 1 to add patient record
Enter 2 to display patient record
Enter choice: |
```



```
Enter 0 to exit
Enter 1 to add patient record
Enter 2 to display patient record
Enter choice: 2
Enter patient ID: 100


Patient ID:  100
Patient name:  Jack
Patient blood type:  0 -
Patient gender:  M
Patient age:  20
Patient's DoB:  06-04-2000
Patient height:  6 feet
Patient weight 70
Patient allergies:
Patient medications:
Patient medical conditions:
Pathological test report:  na
Patient phone no.:  912345 78945
Patient emergency no.:  98745 21365
Remarks:
Patient since :  Wed May 26 11:53:35 2021
```