JCEI'S JAIHIND COLLEGE OF ENGINEERING, KURAN

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

LAB MANUAL

Mini Project
Subject Code: 317536

Prepared By:

Prof.Kale.A.S

JCEI'S JAIHIND COLLEGE OF ENGINEERING, KURAN

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

LAB MANUAL

Third Year Engineering

Semester-VI

Cyber Security

Subject Code: 317536

Class: TE AI&DS

Academic Year 2023-24

Mini Project

Subject Code: 317536

Teaching Scheme	Credit	Examination Scheme
PR: 02 Hours/Week	01	OR: 25 Marks TW: 50 Marks

Guidelines for Instructor's Manual

The instructor‘s manual is to be developed as a reference and hands-on resource. It should include prologue (about University/program/ institute/ department/foreword/ preface), curriculum of the course, conduction and Assessment guidelines, topics under consideration, concept, objectives, outcomes, set of typical applications/assignments/ guidelines, and references.

Guidelines for Student Journal

The laboratory assignments are to be submitted by student in the form of journal. Journal consists of Certificate, table of contents, and handwritten write-up of each assignment (Title, Date of Completion, Objectives, Problem Statement, Software and Hardware requirements, Assessment grade/marks and assessor's sign, Theory- Concept in brief, algorithm, flowchart, test cases, Test Data Set(if applicable), mathematical model (if applicable), conclusion/analysis. Program codes with sample output of all performed assignments are to be submitted as softcopy. As a conscious effort and little contribution towards Green IT and environment awareness, attaching printed papers as part of write-ups and program listing to journal must be avoided. Use of DVD containing students programs maintained by Laboratory In-charge is highly encouraged. For reference one or two journals may be maintained with program prints in the Laboratory.

Guidelines for Laboratory /Term Work Assessment

Continuous assessment of laboratory work should be based on overall performance of Laboratory assignments by a student. Each Laboratory assignment assessment will assign grade/marks based on parameters, such as timely completion, performance, innovation, efficient codes, and punctuality.

Guidelines for Practical Examination

Problem statements must be decided jointly by the internal examiner and external examiner. During practical assessment, maximum weightage should be given to satisfactory implementation of the problem statement. Relevant questions may be asked at the time of evaluation to test the student's understanding of the fundamentals, effective and efficient implementation. This will encourage, transparent evaluation and fair approach, and hence will not create any uncertainty or doubt in the minds of the students. So, adhering to these principles will consummate our team efforts to the promising start of student's academics.

Guidelines for Laboratory Conduction

The instructor is expected to frame the assignments by understanding the prerequisites, technological aspects, utility and recent trends related to the topic. The assignment framing policy need to address the average students and inclusive of an element to attract and promote the intelligent students. Use of open source software is encouraged. Based on the concepts learned. Instructor may also set one assignment or mini-project that is suitable to AI & DS branch beyond the scope of the syllabus.

ASSIGNMENT 1

PROBLEM STATEMENT:

Implementation of S-DES

OBJECTIVE:

1. To understand how encryption takes place using S-DES algorithm which uses 3 different types of keys to encrypted.

PREREQUISITE:

1. Basic of Python Programming

THEORY:

Simplified Data Encryption Standard (S-DES) is equivalent to the DES algorithm. The SDES encryption algorithm produces an 8-bit block of plaintext (example: 10111101) and a 10-bit key as input and makes an 8-bit block of cipher text as output. The S-DES decryption algorithm takes an 8-bit block of cipher text and the same 10-bit key can develop that cipher text as input and makes the initial 8-bit block of plaintext.

These algorithms generate a key and thus encapsulate the message with this key. There are two types of encryptions: asymmetric and symmetric, which are in vogue.

Presentation Layer

The presentation layer in S-DES manages the translation, encryption/decryption, authentication and compression. These are explained below −

Translation

It can transform the complex data structures used by an application string, integers, structures, etc., into a byte flow that can be shared across the network. The message is defined so that communicating devices agree to the structure of the data being transformed. For instance, ASCII or EBCDIC character sets.

Encryption/Decryption

It can handle security and privacy issues. Encryption can scramble the information so that only authorized persons can unscramble the conversation information. Decryption shifts the encryption procedure to interpret the message back into its original form.

There are two types of Encryption which are as follows −

- Asymmetric Encryption − There are two numerically associated keys, such as the name public key and private keys that are created to encrypt and decrypt the message. Asymmetric encryption is considered more secure than symmetric encryption.
- Symmetric Encryption − Symmetric encryption is also defined as conventional or single key Encryption. It is based on a secret key, which both communicating parties share. The sending party encrypts the plain text to cipher text messages using the secret key. The receiving party on receipt of the ciphertext message uses a similar secret key to decrypt it to plain text.

Authentication

It can test the antecedents of the remote party being the real party instead of an impostor. It represents that the message is received from an authentic person, not from an impostor. A digital signature is one of the multiple authentication methods that use the public key encryption method.

Algorithm:

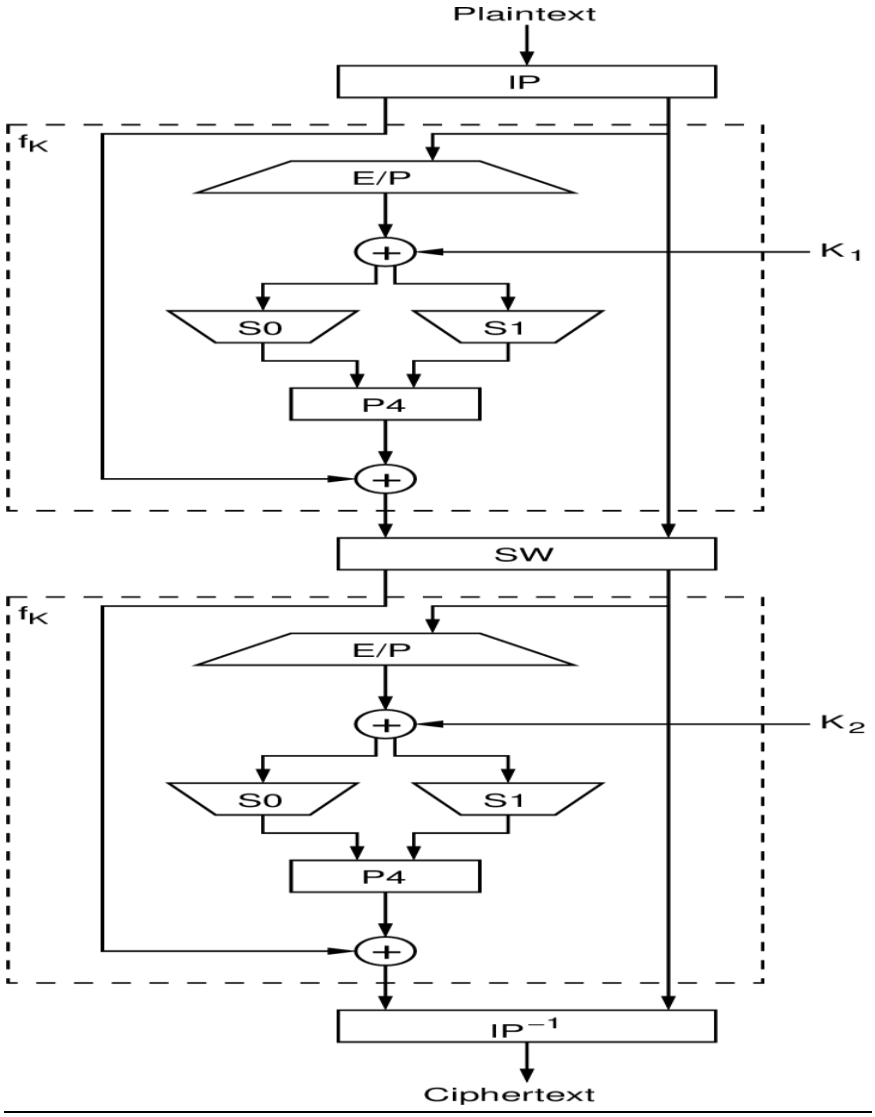
DES encrypts 64-bit blocks using a 56-bit key and produces 64-bit cipher text through a series of steps.

S-DES or Simplified Data Encryption Standard is a simplified version of DES algorithm which is a block cipher that inputs 8-bit plaintext or ciphertext and uses 10-bit key for encryption and decryption. S-DES was designed for educational purposes only, to help students learn about modern cryptanalytic techniques. SDES has similar properties and structure as DES, but has been simplified to make it much easier to perform encryption and decryption by hand with pencil and paper.

The Encryption Processing of plaintext proceeds in 3 phases:-

1. First, the plaintext passes through an initial permutation (IP) that rearranges the bits to produce permutted output.
2. The permutted output is then passed through 16 rounds of both Permutation and Substitution functions. The left and right halves of output are swapped to produce the preoutput.
3. Finally, preoutput is passed through a permutation (IP-1) that is inverse of initial permutation function, to produce ciphertext.

The key is passed through a permutation function. Then a subkey is produced for each 16 rounds by combination of left circular swift and a permutation. The permutation function is the same for every round, but a different subkey is produced because of the repeated shifts of key bits.



CONCLUSION:

After completion of this assignment students are able to understand how encryption took place using S-DES algorithm by using 3 different types of keys to encrypt.

ORAL QUESTION:

1. What is the key length of S-DES algorithm?
2. What is the difference between a block cipher and a stream cipher?
3. What is some common application of S-DES?
4. What happens when you use a weak key with DES?

ASSIGNMENT 2

PROBLEM STATEMENT:
Implementation of S-AES

OBJECTIVE:

1. To understand how to Apply Advanced Encryption Standard Algorithm to encryption of given data.

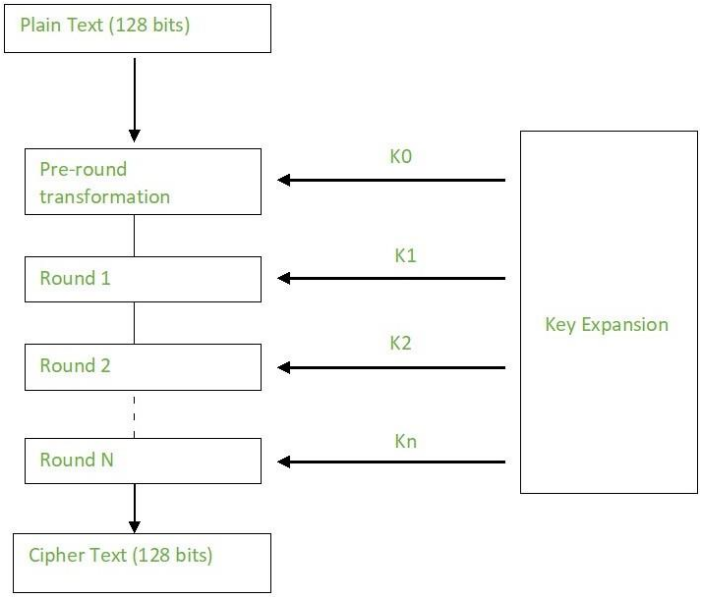
PREREQUISITE: -

1. Basic of Python Programming
2. Concept of Advanced Encryption Standard

THEORY:
AES performs operations on bytes of data rather than in bits. Since the block size is 128 bits, the cipher processes 128 bits (or 16 bytes) of the input data at a time.
The number of rounds depends on the key length as follows:

- 128 bit key – 10 rounds
- 192 bit key – 12 rounds
- 256 bit key – 14 rounds

Creation of Round keys:
A Key Schedule algorithm is used to calculate all the round keys from the key. So the initial key is used to create many different round keys which will be used in the corresponding round of the encryption.



Algorithm:

Encryption:

AES considers each block as a 16 byte (4 byte x 4 byte = 128) grid in a column major arrangement.

[b0 | b4 | b8 | b12 |
| b1 | b5 | b9 | b13 |
| b2 | b6 | b10| b14 |
| b3 | b7 | b11| b15]

Each round comprises of 4 steps:

- SubBytes
- ShiftRows
- MixColumns
- Add Round Key

The last round doesn't have the MixColumns round.

The SubBytes does the substitution and ShiftRows and MixColumns performs the permutation in the algorithm.

SubBytes :

This step implements the substitution.

In this step each byte is substituted by another byte. Its performed using a lookup table also called the S-box. This substitution is done in a way that a byte is never substituted by itself and also not substituted by another byte which is a compliment of the current byte. The result of this step is a 16 byte (4 x 4) matrix like before.

The next two steps implement the permutation.

ShiftRows:

This step is just as it sounds. Each row is shifted a particular number of times.

- The first row is not shifted
- The second row is shifted once to the left.
- The third row is shifted twice to the left.
- The fourth row is shifted thrice to the left.

(A left circular shift is performed.)

[b0 | b1 | b2 | b3] [b0 | b1 | b2 | b3]
| b4 | b5 | b6 | b7 | -> | b5 | b6 | b7 | b4 |
| b8 | b9 | b10 | b11 | | b10 | b11 | b8 | b9 |
[b12 | b13 | b14 | b15] [b15 | b12 | b13 | b14]

MixColumns:

This step is basically a matrix multiplication. Each column is multiplied with a specific matrix and thus the position of each byte in the column is changed as a result.
This step is skipped in the last round.

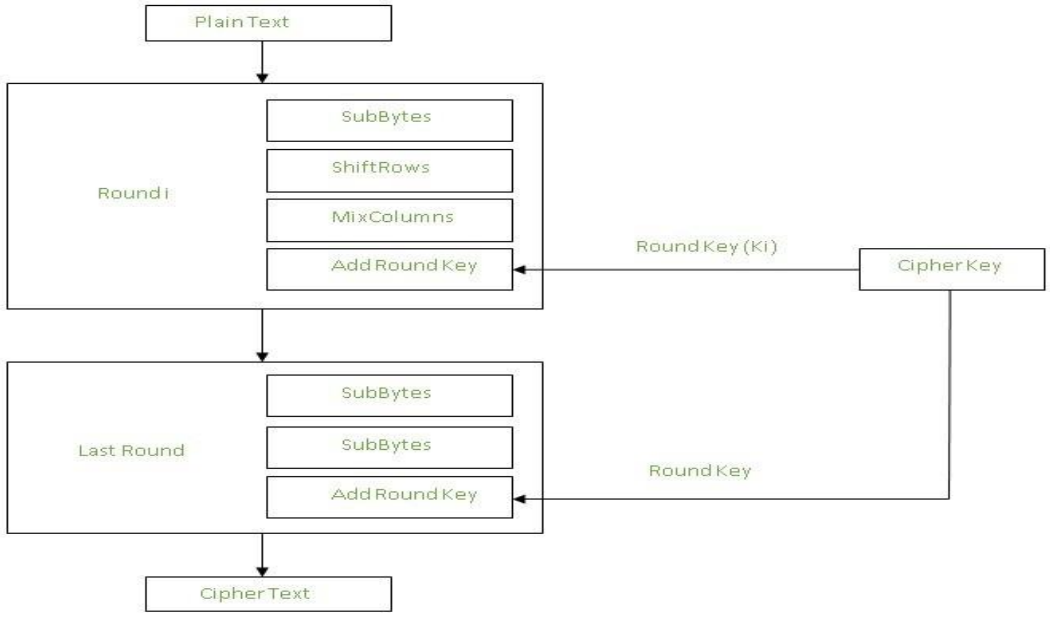
$$\begin{bmatrix} c0 \\ c1 \\ c2 \\ c3 \end{bmatrix} = \begin{bmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{bmatrix} \begin{bmatrix} b0 \\ b1 \\ b2 \\ b3 \end{bmatrix}$$

Add Round Keys:
Now the resultant output of the previous stage is XOR-ed with the corresponding round key. Here, the 16 bytes is not considered as a grid but just as 128 bits of data.
After all these rounds 128 bits of encrypted data is given back as output. This process is repeated until all the data to be encrypted undergoes this process.

Decryption:
The stages in the rounds can be easily undone as these stages have an opposite to it which when performed reverts the changes. Each 128 blocks goes through the 10,12 or 14 rounds depending on the key size.
The stages of each round in decryption is as follows :

- Add round key
- Inverse MixColumns
- ShiftRows
- Inverse SubByte

The decryption process is the encryption process done in reverse so i will explain the steps with notable differences.



Inverse MixColumns:

This step is similar to the MixColumns step in encryption, but differs in the matrix used to carry out the operation.

$$\begin{bmatrix} b0 \\ b1 \\ b2 \\ b3 \end{bmatrix} = \begin{bmatrix} 14 & 11 & 13 & 9 \\ 9 & 14 & 11 & 13 \\ 13 & 9 & 14 & 11 \\ 11 & 13 & 9 & 14 \end{bmatrix} \begin{bmatrix} c0 \\ c1 \\ c2 \\ c3 \end{bmatrix}$$

Inverse SubBytes :

Inverse S-box is used as a lookup table and using which the bytes are substituted during decryption.

APPLICATIONS:

AES is widely used for encryption in various applications and industries due to its strong security, efficiency, and versatility. Some common applications of AES encryption include: File, Database, and Standalone Encryption: AES is most often used to encrypt data at rest.

CONCLUSION:

After completion of this assignment students are able to implement code for Advanced Encryption Standard Algorithm for given data and find the encrypted data of the given data.

ORAL QUESTION:

1. Describe the process of AES key generation.
2. What are the main stages of the AES algorithm?
3. How does key expansion work in AES?
4. What differences exist between the three AES variants, AES-128, AES-192, and AES-256?

ASSIGNMENT 3

PROBLEM STATEMENT:

Implementation of Diffie-Hellman key exchange

OBJECTIVE:

1. To analyze and demonstrate knowledge of Diffie-Hellman key exchange.

PREREQUISITE: -

1. Basic of Computer Networking and Python

THEORY:

The Diffie-Hellman algorithm is being used to establish a shared secret that can be used for secret communications while exchanging data over a public network using the elliptic curve to generate points and get the secret key using the parameters.

For the sake of simplicity and practical implementation of the algorithm, we will consider only 4 variables, one prime P and G (a primitive root of P) and two private values a and b.

P and G are both publicly available numbers. Users (say Alice and Bob) pick private values a and b and they generate a key and exchange it publicly. The opposite person receives the key and that generates a secret key, after which they have the same secret key to encrypt.

Step-by-Step explanation is as follows:

Alice	Bob
Public Keys available = P, G	Public Keys available = P, G
Private Key Selected = a	Private Key Selected = b
Key generated =	Key generated =

Alice	Bob
Exchange of generated keys takes place	
Key received = y	key received = x
Generated Secret Key =	Generated Secret Key =
Algebraically, it can be shown that	
Users now have a symmetric secret key to encrypt	

Algorithm:

Step 1: Alice and Bob get public numbers P = 23, G = 9

Step 2: Alice selected a private key a = 4 and

Bob selected a private key b = 3

Step 3: Alice and Bob compute public values

Alice: x =(9^4 mod 23) = (6561 mod 23) = 6

Bob: y = (9^3 mod 23) = (729 mod 23) = 16

Step 4: Alice and Bob exchange public numbers

Step 5: Alice receives public key y =16 and

Bob receives public key x = 6

Step 6: Alice and Bob compute symmetric keys

Alice: ka = y^a mod p = 65536 mod 23 = 9

Bob: kb = x^b mod p = 216 mod 23 = 9

Step 7: 9 is the shared secret.

CONCLUSION:

After completion of this assignment students are able to understand the Diffie-Hellman key Exchange

ORAL QUESTION:

1. Explain “Diffie-Hellmen key exchange algorithm with suitable example
2. What is Man in the middle attack?
3. How to Preventing a Man-in Middle Attack?

ASSIGNMENT 4

PROBLEM STATEMENT:

Implementation of RSA

OBJECTIVE OF THE ASSIGNMENT:

1. To understand how RSA enables public key encryption and is widely used to secure sensitive data.

PREREQUISITE:

1. Basics of Python

Theory:

RSA algorithm is an asymmetric cryptography algorithm. Asymmetric actually means that it works on two different keys i.e. Public Key and Private Key. As the name describes that the Public Key is given to everyone and the Private key is kept private.

An example of asymmetric cryptography:

1. A client (for example browser) sends its public key to the server and requests some data.
2. The server encrypts the data using the client's public key and sends the encrypted data.
3. The client receives this data and decrypts it.

Since this is asymmetric, nobody else except the browser can decrypt the data even if a third party has the public key of the browser.

The idea! The idea of RSA is based on the fact that it is difficult to factorize a large integer. The public key consists of two numbers where one number is a multiplication of two large prime numbers. And private key is also derived from the same two prime numbers. So if somebody can factorize the large number, the private key is compromised. Therefore encryption strength totally lies on the key size and if we double or triple the key size, the strength of encryption increases exponentially. RSA keys can be typically 1024 or 2048 bits long, but experts believe that 1024-bit keys could be broken in the near future. But till now it seems to be an infeasibl

Algorithm:

Select two prime no's. Suppose P = 53 and Q = 59.

Now First part of the Public key : n = P*Q = 3127.

We also need a small exponent say e:

But e Must be

An integer.

Not be a factor of Φ(n).

1 < e < Φ(n) [Φ(n) is discussed below],

Let us now consider it to be equal to 3.

Our Public Key is made of n and e

Generating Private Key:

We need to calculate Φ(n) :

Such that Φ(n) = (P-1)(Q-1)

so, Φ(n) = 3016

Now calculate Private Key, d :

d = (k*Φ(n) + 1) / e for some integer k

For k = 2, value of d is 2011.

ow we are ready with our – Public Key (n = 3127 and e = 3) and Private Key(d = 2011)

Now we will encrypt “HI”:

Convert letters to numbers : H = 8 and I = 9

Thus Encrypted Data c = (89e)mod n

Thus our Encrypted Data comes out to be 1394

Now we will decrypt 1394 :

Decrypted Data = (cd)mod n

Thus our Encrypted Data comes out to be 89

8 = H and I = 9 i.e. "HI"

CONCLUSION:

After Completion of this assignment students will be able to understand how RSA enables public key encryption and is widely used to secure sensitive data.

ORAL QUESTION:

1. Are strong primes necessary in RSA?
2. How fast RSA is?
3. What would it take to break RSA?
4. How is RSA used for authentication in practice?

ASSIGNMENT 5

PROBLEM STATEMENT:

Implementation of ECC algorithm.

OBJECTIVE:

1. Students should be able to understand ECC algorithm using Python.

PREREQUISITE:

1. Basics of Python programming.

THEORY:

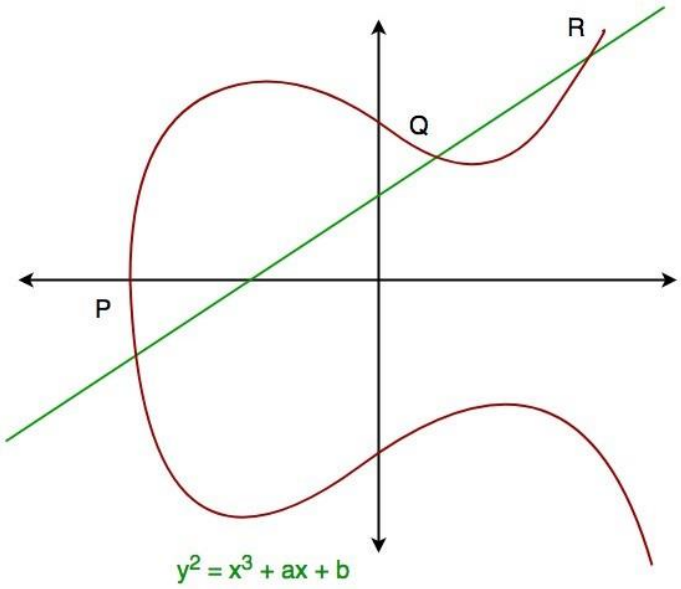
ECC is an approach to public-key cryptography, based on the algebraic structure of elliptic curves over finite fields. ECC requires a smaller key as compared to non-ECC cryptography to provide equivalent security (a 256-bit ECC security has equivalent security attained by 3072-bit RSA cryptography).

For a better understanding of Elliptic Curve Cryptography, it is very important to understand the basics of the Elliptic Curve. An elliptic curve is a planar algebraic curve defined by an equation of the form

Where ‘a’ is the co-efficient of x and ‘b’ is the constant of the equation

The curve is non-singular; that is, its graph has no cusps or self-intersections (when the characteristic of the Coefficient field is equal to 2 or 3).

In general, an elliptic curve looks like as shown below. Elliptic curves can intersect almost 3 points when a straight line is drawn intersecting the curve. As we can see, the elliptic curve is symmetric about the x-axis. This property plays a key role in the algorithm.



This approach uses six tuple {P, a, b, G, n, h}

P = Field that the curve is define over

G = Generator point

a, b = Values define the curve

h = Co- factor

n = Prime order of G

CONCLUSION:

After completion of this students will be able to understand public-key cryptography, based on the algebraic structure of elliptic curves over finite fields.

ORAL QUESTIONS:

1. What are pros and cons of public key cryptography?
2. What is generator point in ECC?
3. What is the size of key used in ECC?
4. Difference between RSA and ECC? Which one is more effective?

Part B : Elective II : Cloud Computing

ASSIGNMENT 1

PROBLEM STATEMENT:

Setting up AWS Environment: Create a new AWS account, Securethe root user, Create an IAM user to use in the account Set up the AWS CLI, Set up a Cloud 9 environment

OBJECTIVE: Student Should be able to understand AWS CLI to create an AWS Cloud9 development environment.

PREREQUISITE:

1) Hardware: 500 GB, HDD, Internet Connection
2) Software: Web Browser

THEORY:

AWS Cloud9 is an integrated development environment, or IDE.

The AWS Cloud9 IDE offers a rich code-editing experience with support for several programming languages and runtime debuggers, and a built-in terminal. It contains a collection of tools that you use to code, build, run, test, and debug software, and helps you release software to the cloud.

You access the AWS Cloud9 IDE through a web browser. You can configure the IDE to your preferences. You can switch color themes, bind shortcut keys, enable programming language-specific syntax coloring and code formatting, and more.

AWS Cloud9 environments
An AWS Cloud9 environment is a place where you store your project's files and where you run the tools to develop your applications.

Using the AWS Cloud9 IDE, you can:

Store your project's files locally on the instance or server.

Clone a remote code repository—such as a repo in AWS CodeCommit—into your environment.

Work with a combination of local and cloned files in the environment.

You can create and switch between multiple environments, with each environment set up for a specific development project. By storing the environment in the cloud, your projects no longer need to be tied

to a single computer or server setup. This enables you to do things such as easily switch between computers and more quickly onboard developers to your team.

CONCLUSION: After Completion of this assignment students will be able to understand how to Setting up AWS Environment: Create a new AWS account, Securethe root user, Create an IAM user to use in the account Set up the AWS CLI, Set up a Cloud 9 environment

ASSIGNMENT 2

PROBLEM STATEMENT: Setup, Create and visualize data in an Amazon Relational Database (Amazon RDS) MS SQL Express server using Amazon Quick Sight

OBJECTIVE: Understand the process of setting up on amazon RDS instance with Microsoft SQL server express.

PREREQUISITE:

1) Hardware: 500 GB, HDD, Internet Connection, 2GB RAM
2) Software: Web Browser, Aws account, SQL Link

THEORY:

Amazon RDS provides a managed relational database service that makes it easy to set up, operate, and scale a relational database in the cloud. With Amazon RDS, you can choose from several database engines, including Microsoft SQL Server (MS SQL). Amazon QuickSight, on the other hand, is a cloud-powered business intelligence service that allows you to easily create and publish interactive dashboards.

Services :

1. Amazon RDS (Relational Database Service):

Amazon RDS provides a managed relational database service in the cloud.

We use Amazon RDS to set up and manage an MS SQL Express database instance.

2. Amazon QuickSight:

Amazon QuickSight is a cloud-powered business intelligence service that enables users to create and publish interactive dashboards.

We use QuickSight to connect to the MS SQL Express database hosted on Amazon RDS and visualize the data stored within it.

3. **AAWS Identity and Access Management (IAM)**

IAM is used to manage access to AWS services securely.

We utilize IAM to define roles and permissions for accessing resources such as Amazon RDS and QuickSight.

IAM ensures that only authorized users or applications can interact with the data and services in the AWS environment.

4. **Amazon S3 (Simple Storage Service):**

Amazon S3 provides scalable object storage in the cloud.

While not explicitly mentioned in the project outline, S3 can be used to store data files, backups, or other resources related to the project.

5. Amazon EC2 (Elastic Compute Cloud):

Amazon EC2 offers resizable computing capacity in the cloud.

CONCLUSION: After Completion of this assignment students will be able to understand how to Setup, Create and visualize data in an Amazon Relational Database (Amazon RDS) MS SQL Express server using Amazon Quick Sight

PROBLEM STATEMENT: Setup, Create and connect your Word Press site to an object storage bucket using Lightsail service

OBJECTIVE: 1) understanding cloud storage integration
2) familiarity with lightsail services

PREREQUISITE:

1) Hardware: 500 GB HDD, 2GB RAM
2) Software: lightsail account, wordpress

THEORY:

Amazon Web Services (AWS) offer reliable, scalable, and inexpensive cloud computing services. These cloud computing web services provide various services related to networking, compute, storage, middleware, IOT, and other processing capacities, as well as software tools via AWS. And in this blog, we will understand the uses of Amazon Lightsail.

Amazon Lightsail is a simple-to-use vendor of virtual private servers. Lightsail provides an object storage service in which we will walk through the steps required to set it up on a WordPress site on Amazon Lightsail and then connect the website through Lightsail bucket storage to store the website's images and attachments.

To begin with this, firstly, we need to install the 'WP Offload Media Lite Plugin' on our WordPress site and configure it to connect our Lightsail bucket. Once the connections are established, our WordPress site will automatically upload all media files to the bucket instead of the instance's disk.

CONCLUSION: After Completion of this assignment students will be able to understand how to Setup, Create and connect your Word Press site to an object storage bucket using Lightsail service