Sanjivani Rural Education Society's
Sanjivani College Of Engineering, Kopargaon

Laboratory Manual

Laboratory Practice-III (ICS)

Final Year – Computer Engineering (2015)

Examination Scheme

Practical: 50 marks **TW: 50 Marks**

Teaching Scheme

Practical: 02 Hrs/Week/Batch

Prepared By

Prof. B.J.Dange

Department of Computer Engineering
SRES Sanjivani College of Engineering, Kopargaon
Dist- Ahmednagar (M.S.)
INDIA

Sanjivani Rural Education Society's

Sanjivani College Of Engineering, Kopargaon

Department of Computer Engineering

Approval Sheet

Laboratory Manual

Laboratory Practice-III (ICS)

Final Year – Computer Engineering (2015)

Prof. B.J.Dange
Subject Incharge

Dr. D. B. Kshirsagar
Head Dept

Dr. D. N. Kyatanavar
Principal

List of Assignments

Sr.No.	Title
1	Implementation of Diffie-Hellman key exchange
2	Implementation of RSA
3	Implementation of S-DES
4	Implementation of S-AES
5	Mini-project

Syllabus:

Savitribai Phule Pune University
Fourth Year of Computer Engineering (2015 Course)
410254:Laboratory Practice III

Teaching Scheme:	Credit	Examination Scheme:
Practical : 04 Hours/Week	**02**	**Term Work: 50 Marks** **Practical: 50 Marks**

Companion Courses: 410250 and 410251

Course Objectives and Outcomes: Practical hands on is the absolute necessity as far as employability of the learner is concerned. The presented course is solely intended to enhance the competency by undertaking the laboratory assignments of the core courses.

About

Laboratory Practice III is for practical hands on for core courses Machine Learning and Information & Cyber Security.

Guidelines for Laboratory Conduction

- **List of recommended programming assignments and sample mini-projects is provided for reference.**
- Referring these, Course Teacher or Lab Instructor may frame the assignments/mini-project by understanding the prerequisites, technological aspects, utility and recent trends related to the respective courses.
- Preferably there should be multiple sets of assignments/mini-project and distribute among batches of students.
- Real world problems/application based assignments/mini-projects create interest among learners serving as foundation for future research or startup of business projects.
- Mini-project can be completed in group of 2 to 3 students.
- Software Engineering approach with proper documentation is to be strictly followed.
- Use of open source software is to be encouraged.
- Instructor may also set one assignment or mini-project that is suitable to respective course beyond the scope of syllabus.

Operating System recommended :- 64-bit Open source Linux or its derivative
Programming Languages: C++/JAVA/PYTHON/R
Programming tools recommended: Front End: Java/Perl/PHP/Python/Ruby/.net, Backend : MongoDB/ MYSQL/Oracle, Database Connectivity : ODBC/JDBC, Additional Tools: Octave, Matlab, WEKA**.**

Guidelines for Student Journal

The laboratory assignments are to be submitted by student in the form of journal. Journal may consists of prologue, Certificate, table of contents, and **handwritten write-up** of each assignment (Title, Objectives, Problem Statement, Outcomes, software and Hardware requirements, Date of Completion, Assessment grade/marks and assessor's sign, Theory- Concept in brief, Algorithm/Database design, test cases, 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 may be avoided. Use of digital storage media/DVD containing students programs maintained by lab In-charge is highly encouraged. For reference one or two journals may be maintained with program prints at Laboratory.

Guidelines for Assessment

Continuous assessment of laboratory work is to be done based on overall performance and lab assignments performance of student. Each lab assignment assessment will assign grade/marks based on parameters with appropriate weightage. Suggested parameters for overall assessment as well as each lab assignment assessment include- timely completion, performance, innovation, efficient codes, punctuality and neatness **reserving weightage for successful mini-project completion and related documentation.**

Guidelines for Practical Examination

- Both internal and external examiners should jointly frame suitable problem statements for practical examination based on the term work completed.
- During practical assessment, the expert evaluator should give the maximum weightage to the satisfactory implementation of the problem statement.
- The supplementary and relevant questions may be asked at the time of evaluation to test the student‟s for advanced learning, understanding of the fundamentals, effective and efficient implementation.
- Encouraging efforts, transparent evaluation and fair approach of the evaluator 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 boost to the student's academics.

Guidelines for Instructor's Manual

The instructor‟s manual is to be developed as a hands-on resource and as ready reference. The instructor's manual need to include prologue (about University/program/ institute/ department/foreword/ preface etc), University syllabus, conduction and Assessment guidelines, topics under consideration-concept, objectives, outcomes, set of typical applications/assignments/ guidelines, references among others.

Suggested List of Laboratory Assignments

410251:: : Information and Cyber Security

1.	Implementation of S-DES
2.	Implementation of S-AES
3.	Implementation of Diffie-Hellman key exchange
4.	Implementation of RSA.
5.	Implementation of ECC algorithm.
6.	**Mini Project 1:** SQL Injection attacks and Cross -Site Scripting attacks are the two most common attacks on web application. Develop a new policy based Proxy Agent, which classifies the request as a scripted request or query based request, and then, detects the respective type of attack, if any in the request. It should detect both SQL injection attack as well as the Cross-Site Scripting attacks.
7.	**Mini Project 2:**This task is to demonstrate insecure and secured website. Develop a web site and demonstrate how the contents of the site can be changed by the attackers if it is http based and not secured. You can also add payment gateway and demonstrate how money transactions can be hacked by the hackers. Then support your website having https with SSL and demonstrate how secured website is.

Sanjivani Rural Education Society's

Sanjivani College of Engineering, Kopargaon-423603

DEPARTMENT OF COMPUTER ENGINEERING

Instruction No. 01	Title: Implementation of Diffie-Hellman key exchange
LP-III (ICS)/ SOP / Sr. No. 01	
Rev 00 Date : 27/12/2017	

Aim: Implementation of Diffie-Hellman key exchange

Input:

The value of P : 23

The value of G : 9

The private key a for Alice : 4 The private key b for Bob : 3

Output:

Secret key for the Alice is :9

Secret Key for the Bob is : 9

Theory :

Diffie Hellman algorithm is a public-key algorithm used to establish a shared secret that can be used for secret communications while exchanging data over a public network.

It is primarily used as a method of exchanging cryptography keys for use in symmetric encryption algorithms. It was Proposed in 1976 by Whitfield Diffie and Martin Hellman. Diffie-Hellman is currently used in many protocols like Secure Sockets Layer (SSL)/Transport Layer Security (TLS), Secure Shell (SSH), Internet Protocol Security (IPSec), Public Key Infrastructure (PKI).

- 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 received the key and from that generates a secret key after which they have the same secret key to encrypt.

Step by Step Explanation

Alice	Bob
Public Keys available = P, G	Public Keys available = P, G

Alice	Bob
Private Key Selected = a	Private Key Selected = b
Key generated = $x = G^a mod P$	Key generated = $y = G^b mod P$
Exchange of generated keys takes place	
Key received = y	key received = x
Generated Secret Key = $k_a = y^a mod P$	Generated Secret Key = $k_b = x^b mod P$

Algebraically it can be shown that $k_a = k_b$

Users now have a symmetric secret key to encrypt

Example

```
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.
```

Code-

```
/* This program calculates the Key for two persons using the Diffie-Hellman Key exchange
algorithm */
#include<stdio.h>
#include<math.h>

// Power function to return value of a ^ b mod P
long long int power(long long int a, long long int b,
                                                long long int P)
{
        if (b == 1)
```

```
            return a;

        else
            return (((long long int)pow(a, b)) % P);
}
//Driver program
int main()
{
        long long int P, G, x, a, y, b, ka, kb;

        // Both the persons will be agreed upon the
            // public keys G and P
        P = 23; // A prime number P is taken
         printf("The value of P : %lld\n", P);

        G = 9; // A primitve root for P, G is taken
        printf("The value of G : %lld\n\n", G);

        // Alice will choose the private key a
        a = 4; // a is the chosen private key
        printf("The private key a for Alice : %lld\n", a);
        x = power(G, a, P); // gets the generated key

        // Bob will choose the private key b
        b = 3; // b is the chosen private key
        printf("The private key b for Bob : %lld\n\n", b);
        y = power(G, b, P); // gets the generated key

        // Generating the secret key after the exchange
            // of keys
        ka = power(y, a, P); // Secret key for Alice
        kb = power(x, b, P); // Secret key for Bob
        printf("Secret key for the Alice is : %lld\n", ka);
        printf("Secret Key for the Bob is : %lld\n", kb);
```

```
    return 0;
}
```

Conclusion- Thus the Diffie-Hellman key exchange algorithm had been successfully implemented using C.

Questions:

1. What is Prime Number?
2. What primitive roots of prime number ?
3. How to calculate primitive roots of prime number?
4. What is shared secret key ?
5. Explain the Diffe-Hellman Key exchange algorithm ?
6. What is key in cryptography ?
7. What is private key?
8. Whats is Public key ?
9. What is the use of Diffe-Hellman Key Exchange algorithm ?
10. What is advantage of Diffe-hellman Key exchange algorithm ?

Prepared By	Approved By
Prof.B.J.Dange	Dr..D.B.Kshirsagar
Subject Incharge	H.O.D.

Sanjivani Rural Education Society's

Sanjivani College of Engineering, Kopargaon-423603

DEPARTMENT OF COMPUTER ENGINEERING

Instruction No. 02	Title: Implementation of RSA Algorithm
LP-III(ICS)/ SOP/ Sr.No.02	
Rev 00 Date : 27/12/2017	

Aim: Implementation of RSA Algorithm.

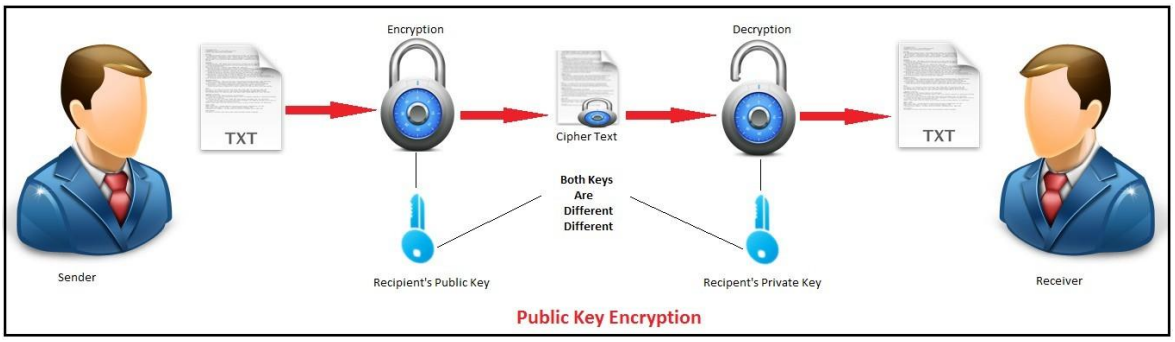
Input: two prime numbers p,q

Output: public and private key

Theory :

RSA Algorithm is used to encrypt and decrypt data in modern computer systems and other electronic devices. RSA algorithm is an asymmetric cryptographic algorithm as it creates 2 different keys for the purpose of encryption and decryption. It is public key cryptography as one of the keys involved is made public. RSA stands for Ron Rivest, Adi Shamir and Leonard Adleman who first publicly described it in 1978.

RSA makes use of prime numbers (arbitrary large numbers) to function. The public key is made available publicly (means to everyone) and only the person having the private key with them can decrypt the original message.



Working of RSA Algorithm

RSA involves use of public and private key for its operation. The keys are generated using the following steps:-

1. Two prime numbers are selected as **p** and **q**
2. **n = pq** which is the modulus of both the keys.
3. Calculate **totient = (p-1)(q-1)**
4. Choose **e** such that **e > 1** and coprime to **totient** which means **gcd (e, totient)** must be equal to **1**, **e** is the public key

5. Choose **d** such that it satisfies the equation **de = 1 + k (totient)**, **d** is the private key not known to everyone.
6. Cipher text is calculated using the equation **c = m^e mod n** where **m** is the message.
7. With the help of **c** and **d** we decrypt message using equation **m = c^d mod n** where **d** is the private key.

Code-

//Program for RSA asymmetric cryptographic algorithm//for demonstration values are relatively small compared to practical application

```
#include<stdio.h>
#include<math.h>

//to find gcd
int gcd(int a, int h)
{
    int temp;
    while(1)
 {
   temp = a%h;
        if(temp==0)
        return h;
        a = h;
        h = temp;
    }
}
int main()
{
    //2 random prime numbers
    double p = 3;
    double q = 7;
    double n=p*q;
    double count;
    double totient = (p-1)*(q-1);
     //public key
    //e stands for encrypt
    double e=2;
    //for checking co-prime which satisfies e>1
    while(e<totient)
{
    count = gcd(e,totient);
    if(count==1)
        break;
    else
        e++;
    }
   //private key
    //d stands for decrypt
    double d;
```

```
    //k can be any arbitrary value
    double k = 2;

    //choosing d such that it satisfies d*e = 1 + k * totient
    d = (1 + (k*totient))/e;
    double msg = 12;
    double c = pow(msg,e);
    double m = pow(c,d);
    c=fmod(c,n);
    m=fmod(m,n);

    printf("Message data = %lf",msg);
    printf("\np = %lf",p);
printf("\nq = %lf",q);
    printf("\nn = pq = %lf",n);
    printf("\ntotient = %lf",totient);
  printf("\ne = %lf",e);
    printf("\nd = %lf",d);
    printf("\nEncrypted data = %lf",c);
    printf("\nOriginal Message Sent = %lf",m);
   return 0;}
```

Conclusion- Thus the RSA algorithm had been successfully implemented using C.

Questions:

1. What is the use of RSA algorithm?
2. What is public key in RSA algorithm?
3. Is RSA secure?
4. What is totient function ?
5. Explain the working of RSA ?
6. What is the condition for RSA algorithm?
7. What is application of RSA?

Prepared By	Approved By
Prof.B.J.Dange	Dr..D.B.Kshirsagar
Subject Incharge	H.O.D.

Sanjivani Rural Education Society's

Sanjivani College of Engineering, Kopargaon-423603

DEPARTMENT OF COMPUTER ENGINEERING

Instruction No. 03	Title: Implementation of S-DES Algorithm
LP-III(ICS)/ SOP/ Sr.No.03	
Rev 00 Date : 27/12/2017	

Aim: Title: Implementation of S-DES Algorithm

Input:

Plain Text: 01101101

Key:10011000

Output:

Cipher Text: 01000110

Theory : The S-DES encryption algorithm takes an 8-bit block of plaintext. and a 10-bit key as input and produces an 8-bit block of ciphertext as output. The S-DES decryption algorithm takes an 8-bit block of ciphertext and the same 10-bit key used to produce that ciphertext as input and produces the original 8-bit block of plaintext.

The encryption algorithm involves five functions: an initial permutation (IP); a complex function labeled fK, which involves both permutation and substitution operations and depends on a key input; a simple permutation function that switches (SW) the two halves of the data; the function fK again; and finally a permutation function that is the inverse of the initial permutation (IP–1).

A compromise is to use a 10-bit key from which two 8-bit subkeys are generated, as depicted in **Figure 1**. In this case, the key is first subjected to a permutation (P10). Then a shift operation is performed. The output of the shift operation then passes through a permutation function that produces an 8-bit output (P8) for the first subkey (K1). The output of the shift operation

also feeds into another shift and another instance of P8 to produce the second subkey (K2).

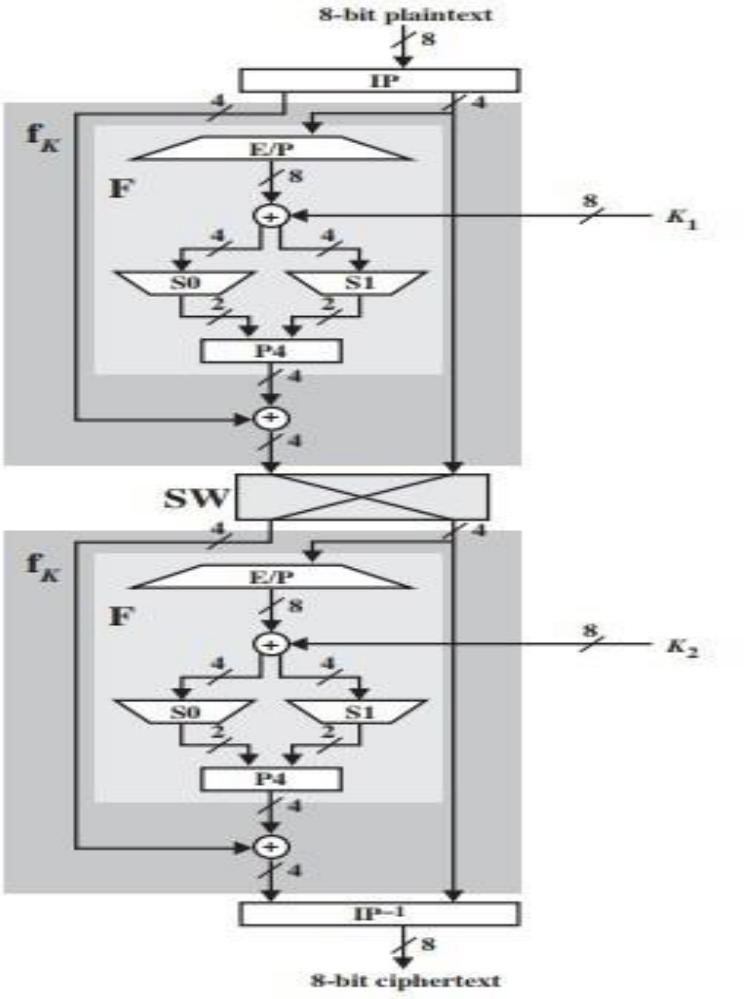


Fig 1.Simplified DES Encryption detail.

Algorithm to generate key

As there are two rounds we have to generate two keys from the given 10-bit key

1: Apply permutation function P10 to 10 bit key

2: divide the result into two part each containing 5-bit L0 and L1

3: apply Circular Left Shift to both L0 and L1

4: combine both L0 and L1 which will form out 10-bit number

5: apply permutation function P8 on result to select 8 out of 10 bits for key K1 (for the first round)

6: again apply second Circular Left Shift to L0 and L1

7: combine the result, which will form out 10-bit number

8: apply permutation function P8 on result to select 8 out of 10 bits for key K2 (for the second round)

Algorithm for S-DES Encryption

1: get 8 bit message text (M) applied it to Initial permutation function (IP)

2: divide IP(M) into nibbles M0 and M1

3: apply function Fk on M0

4: XOR the result with M1 (M1 (+) Fk(M0))

5: Swap the result with M1 (i.e. make M1 as lower nibble (M0) and result as higher nibble (M1))

6: repeat the step 1 to 4 (go for the next round)

7: apply (IP_{-1}) on the result to get the encrypted data.

Algorithm for function Fk

1: give the 4-bit input to EP (Expansion function) the result will be a 8-bit expanded data

2: XOR the 8-bit expanded data with 8-bit key (K1 for the first round and K2 for the

second round)

2: divide result into upper (P1) and lower (P2) nibble

3: apply compression function S0 to P0 and S1 to P1, which will compress the 4-bit input

to 2-bit output

4: combine 2-bit output from S0 and S1 to form a 4-bit digit

5: apply permutation function P4 to 4-bit result.

Functions using in S-DES

P10 = 3 5 2 7 4 10 1 9 8 6

P8 = 6 3 7 4 8 5 10 9

P4 = 2 4 3 1

IP = 2 6 3 1 4 8 5 7

IP^{-1} = 4 1 3 5 7 2 8 6

EP = 4 1 2 3 2 3 4 1

S0:

1 0 3 2

3 2 1 0

0 2 1 3

3 1 3 2

S1:

0 1 2 3

2 0 1 3

3 0 1 0

2 1 0 3

Conclusion- Thus the Simplified Data Encryption Standard algorithm had been successfully implemented.

Questions:

1. What is encryption and Decryption?

2. Differentiate block Ciphers and Stream Ciphers?

3. What is Symmetric key Encryption?

4. What is asymmetric key encryption?

5. What is the size of the key in the SDES algorithm?

6. Explaiin the functions of S-DES algorithm?

7. What is S-DES encryption?

8. What is Cryptanalysis?

9. Differentiate S-DES and DES in Detail.

Prepared By	Approved By
Prof. B.J.Dange	Dr. D.B.Kshirsagar
Subject Incharge	H.O.D.

Sanjivani Rural Education Society's

Sanjivani College of Engineering, Kopargaon-423603

DEPARTMENT OF COMPUTER ENGINEERING

Instruction No. 04	Title: Implementation of S-AES Algorithm
LP-III(ICS)/ SOP/ Sr.No.04	
Rev 00 Date : 27/12/2017	

Aim: Title: Implementation of S-AES Algorithm

Input:

Encryption

16-bit Plaintext, *P*: 1101 0111 0010 1000

16-bit Key, *K*: 0100 1010 1111 0101

Decryption:

Input: Cipher Text: 0010 1110 1110 1110
16-bit Key, *K*: 0100 1010 1111 0101

Output:

Cipher Text: 0010 1110 1110 1110(Encryption)

Plaintext: 16-bit Plaintext, *P*: 1101 0111 0010 1000(Decryption)

Theory: S-AES is to AES as S-DES is to DES. In fact, the structure of S-AES is exactly the same as AES. The differences are in the key size (16 bits), the block size (16 bits) and the number of rounds (2 rounds).Each round Consists of Four transformations. These four transformations are described below.

Substitute nibbles: Instead of dividing the block into a four by four array of bytes, S-AES divides it into a two by two array of "nibbles", which are four bits long. This is called the state array and is shown below.

Shift Rows The next stage is to shift the rows. In fact, the first row is left alone and the second row is shifted.

Mix Columns: After shifting the rows, we mix the columns. Each column is multiplied by the matrix.

Add Round Key: The last stage of each round of encryption is to add the round key. (In fact, this is also done before the first round.) Before the first round, the first two words ($W0$ and $W1$) of the expanded key are added. In the first round, $W2$ and $W3$ are added. In the last round, $W4$ and $W5$ are added. All additions are done modulo 2, that is, with XOR.

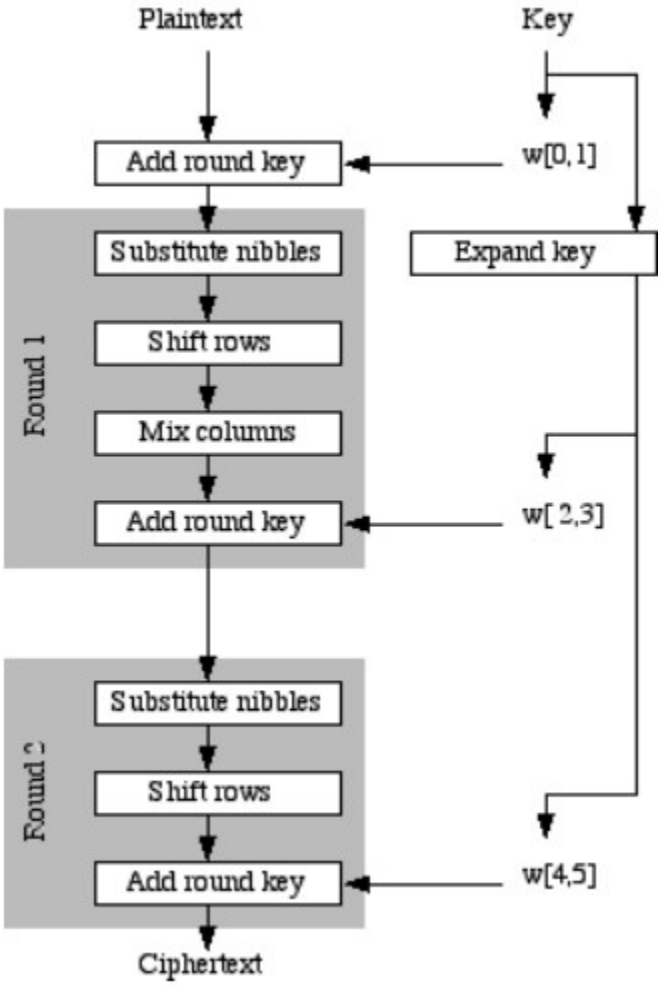


Fig.1: Simplified AES Encryption

Key Expansion: Key expansion is done very similarly to AES. The four nibbles in the key are grouped into two 8-bit "words", which will be

expanded into 6 words. The first part of the expansion, which produces the third and fourth words, is shown below. The rest of the expansion is done in exactly the same way, replacing $W0$ and $W1$ with $W2$ and $W3$, and replacing $W2$ and $W3$ with $W4$ and $W5$.

Algorithm for S-AES Encryption

11. Derive the set of round keys from the cipher key.
12. Initialize the state array with the block data (plaintext).
13. Add the initial round key to the starting state array.
14. Perform two rounds of state manipulation.
15. Perform the tenth and final round of state manipulation.
16. Copy the final state array out as the encrypted data (ciphertext).

Conclusion- Thus the Simplified Advance Encryption Standard algorithm had been successfully implemented.

Questions:

1. What is the key size used in S-AES.
2. Which are the Four stages in each round of S-AES?
3. Differentiate S-DES and S-AES?
4. Describe the function of shift rows Transformation?
5. What is the difference between AES and S-AES algorithm?
6. What is linear cryptanalysis?
7. What is differential cryptanalysis?
8. State the advantages of AES algorithm?

Prepared By	Approved By
Prof .B .J. Dange	Dr. D.B.Kshirsagar
Subject Incharge	H.O.D.

Sanjivani Rural Education Society's

Sanjivani College of Engineering, Kopargaon-423603

DEPARTMENT OF COMPUTER ENGINEERING

Instruction No. 05	Title: Miniproject
LP-III(ICS)/ SOP/ Sr.No.05	
Rev 00 Date : 27/12/2017	

Aim: Miniproject

Input: -

Output: -

Theory :

Mini Project 1: SQL Injection attacks and Cross -Site Scripting attacks are the two most common attacks on web application. Develop a new policy based Proxy Agent, which classifies the request as a scripted request or query based request, and then, detects the respective type of attack, if any in the request. It should detect both SQL injection attack as well as the Cross-Site Scripting attacks.

OR

Mini Project 2: This task is to demonstrate insecure and secured website. Develop a web site and demonstrate how the contents of the site can be changed by the attackers if it is http based and not secured. You can also add payment gateway and demonstrate how money transactions can be hacked by the hackers. Then support your website having https with SSL and demonstrate how secured website is.

Prepared By	Approved By
Prof. B.J.Dange	Dr..D.B.Kshirsagar
Subject Incharge	H.O.D.