Vision and Mission of the Institute

Vision of the Institute

To achieve excellence in technical education through innovative teaching and multidisciplinary research with professionalism to serve the global society.

Mission of the Institute

Jaya Engineering College (JEC) will Endeavor

M1-To provide state of art teaching and learning for Engineering and Technology Research and Management studies.

M2-To provide quality education, self-discipline and ethical values.

M3-To associate with R&D and industries to have connectivity with the society.

M4–To impart knowledge to become empowered professionals in the field of Engineering and Management.

Vision and Mission of the Department

Vision of the department

To achieve Excellence in Computer Science and Engineering by providing quality technical education to cater the need of industry and society through research and development.

Mission of the department

The Computer Science and Engineering Department is committed to:

M1: Provide strong fundamentals and technical skills in Computer Science Engineering through effective teaching and learning methods.

M2: Impart high quality experiential learning to get expertise in modern software tools and to procure the real time requirements of the industry.

M3: Inculcate problem solving and team building skills and promote lifelong learning with a sense of societal and ethical responsibilities.

Program Educational Objectives (PEOs)

PEO1: Apply their technical competence in computer science to solve real world problems, with technical and people leadership**.**

PEO2: Conduct cutting edge research and develop solutions on problems of social relevance.

PEO3: Work in a business environment, exhibiting team skills, work ethics, adaptability and life ling learning

Program specific outcome

- Exhibit design and programming skills to build and automate business solutions using cutting edge technologies.
- Strong theoretical foundation leading to excellence and excitement towards research, to provide elegant solutions to complex problems.
- Ability to work effectively with various engineering fields as a team to design, build and develop system applications.

Program Outcome

PO1: Engineering Knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems

PO2: Problem analysis: Identify, formulate, review research literature, and analyse complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences

PO3: Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO4: Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5: Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.

PO6: The engineer and society: Apply reasoning informed by the contextual knowledge to access societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice

PO7: Environment and sustainability: Understand the impact of the professional engineering Solutions in societal and environmental contexts and demonstrate the knowledge of and need for sustainable development.

PO8: Ethics: Apply ethical principles and commit to professional ethics, responsibilities, and norms of the engineering practice.
PO9: Individual and team work: Function effectively as an individual and as member or leader in diverse teams and in multidisciplinary settings
PO10: Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large such as being able to comprehend and write effective reports and design documentation, make effective presentations and give and receive clear instructions.
PO11: Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to once own work as a member and leader in a team to manage projects and multidisciplinary environments
PO12: Life –long learning: Recognize the need for and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change

COURSE OUTCOME:

CO1 Use the Substitution, Transposition cipher techniques
CO2 Analyse the DES, AES cryptographic algorithm
CO3 Solve the algorithms RSA, Diffie-Hellman Key Exchange algorithm
CO4 Design various Authentication schemes
CO5 Implement the network security tools and vulnerability assessment tools

DATE:	**CAESAR CIPHER**
EXPT NO: **1**(a)	

AIM:

Write a C++ program for Substitution ciphers using Caesar Cipher.

ALGORITHM:

1. Enter the plaintext to be encrypted and enter the key value.
2. Function encrypt ( ) will be invoked for encryption.
   a) Take the ascii value of each plain text char and add it with value of key % 26
   b) Take the equivalent char value for the resultant value of sub-step (a)
   c) Return the cipher text.
3. Function decrypt ( ) will be invoked for decryption.
   a) Take the ascii value of each cipher text char and subtract the value of key % 26 from it
   b) Take the equivalent char value for the resultant value of sub-step(a)
   c) Return the plain text.

PROGRAM :

```
#include <iostream>
using namespace std;
void decrypt(char[], int);
int main( )
{
char plaintext[20];
int key;
cout<<"\nCAESAR CIPHER\n\n";
cout<<"\nEnter any String:";
cin>>plaintext;
cout<<"\n Enter the Key: ";
cin>>key;
encrypt(plaintext,key);
return 0;
}
void  encrypt(char str[20], int   key)
{
char ch;
int   length= strlen(str);
        for(int i = 0; i < length; i++)
        {
         ch = str[i];
                if (isupper(ch))
                {
                ch = ch + (key % 26);
                     if (ch > 'Z')
                        ch = ch - 26;
                }
                else if (islower(ch))
                {
                ch = ch + (key % 26);
                        if (ch > 'z')
                                ch = ch - 26;
                }
        str[i] = (char) ch;
        }
cout<<"\n\nEncrypted String is:" <<str;
decrypt(str,key);
}
void  decrypt(char str[20], int key)
{
  char  ch;
  int length= strlen(str);
        for(int i = 0; i < length; i++)
        {
        ch = str[i];
                if (isupper(ch))
                {
                ch = ch - (key % 26);
                        if (ch < 'A')
                                ch = ch - 26;
                }
                else if (islower(ch))
                {
                ch = ch - (key % 26);
                        if (ch < 'a')
                                ch = ch - 26;
                }
        str[i] = (char) ch;
        }
   cout<<"\n\nDecrypted String is:"<<str;
}
```

OUTPUT :

```
[s@localhost ~]$ g++ ceasernew.cpp    [ –o   ceasernew.exe ]
[s@localhost ~]$ ./a.out              [ceaser]
CAESAR CIPHER
Enter any String:heltin
Enter the Key: 3
Encrypted String is:khowlq
 Decrypted String is:heltin
```

RESULT:

Thus the implementation of Caesar cipher had been executed successfully.

DATE:	**PLAYFAIR CIPHER**
EXPT NO: **1**(b)	

AIM:

To write a C program to implement the Playfair Substitution technique.

ALGORITHM:

STEP-1: Read the plain text from the user.
STEP-2: Read the keyword from the user.
STEP-3: Arrange the keyword without duplicates in a 5*5 matrix in the row order and fill the remaining cells with missed out letters in alphabetical order. Note that ‘i’ and ‘j’ takes the same cell.
STEP-4: Group the plain text in pairs and match the corresponding corner letters by forming a rectangular grid.
STEP-5: Display the obtained cipher text.

PROGRAM:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

#define SIZE 30

void toLowerCase(char plain[], int ps)
{
   int i;
   for (i = 0; i < ps; i++) {
      if (plain[i] > 64 && plain[i] < 91)
         plain[i] += 32;
   }
}

int removeSpaces(char* plain, int ps)
{
   int i, count = 0;
   for (i = 0; i < ps; i++)
      if (plain[i] != ' ')
         plain[count++] = plain[i];
   plain[count] = '\0';
   return count;
}

void generateKeyTable(char key[], int ks, char keyT[5][5])
{
   int i, j, k, flag = 0, *dicty;

   dicty = (int*)calloc(26, sizeof(int));
   for (i = 0; i < ks; i++) {
      if (key[i] != 'j')
```

```
            dicty[key[i] - 97] = 2;
    }

    dicty['j' - 97] = 1;

    i = 0;
    j = 0;

    for (k = 0; k < ks; k++) {
        if (dicty[key[k] - 97] == 2) {
            dicty[key[k] - 97] -= 1;
            keyT[i][j] = key[k];
            j++;
            if (j == 5) {
                i++;
                j = 0;
            }
        }
    }

    for (k = 0; k < 26; k++) {
        if (dicty[k] == 0) {
            keyT[i][j] = (char)(k + 97);
            j++;
            if (j == 5) {
                i++;
                j = 0;
            }
        }
    }
}

void search(char keyT[5][5], char a, char b, int arr[])
{
    int i, j;

    if (a == 'j')
        a = 'i';
    else if (b == 'j')
        b = 'i';

    for (i = 0; i < 5; i++) {

        for (j = 0; j < 5; j++) {

            if (keyT[i][j] == a) {
                arr[0] = i;
                arr[1] = j;
            }
            else if (keyT[i][j] == b) {
                arr[2] = i;
                arr[3] = j;
            }
        }
    }
}

int mod5(int a)
{
    return (a % 5);
}

int prepare(char str[], int ptrs)
{
    if (ptrs % 2 != 0) {
        str[ptrs++] = 'z';
        str[ptrs] = '\0';
    }
    return ptrs;
}

void encrypt(char str[], char keyT[5][5], int ps)
{
    int i, a[4];

    for (i = 0; i < ps; i += 2) {

        search(keyT, str[i], str[i + 1], a);

        if (a[0] == a[2]) {
            str[i] = keyT[a[0]][mod5(a[1] + 1)];
            str[i + 1] = keyT[a[0]][mod5(a[3] + 1)];
        }
        else if (a[1] == a[3]) {
            str[i] = keyT[mod5(a[0] + 1)][a[1]];
            str[i + 1] = keyT[mod5(a[2] + 1)][a[1]];
        }
        else {
            str[i] = keyT[a[0]][a[3]];
            str[i + 1] = keyT[a[2]][a[1]];
        }
    }
}

void encryptByPlayfairCipher(char str[], char key[])
{
    char ps, ks, keyT[5][5];

    ks = strlen(key);
    ks = removeSpaces(key, ks);
    toLowerCase(key, ks);
    ps = strlen(str);
    toLowerCase(str, ps);
    ps = removeSpaces(str, ps);
     ps = prepare(str, ps);
     generateKeyTable(key, ks, keyT);
     encrypt(str, keyT, ps);
}

int main()
{
    char str[SIZE], key[SIZE];

    strcpy(key, "Monarchy");
    printf("Key text: %s\n", key);
     strcpy(str, "instruments");
    printf("Plain text: %s\n", str);

    encryptByPlayfairCipher(str, key);
     printf("Cipher text: %s\n", str);
     return 0;
}
```

OUTPUT:

```
[s@localhost ~]$ g++ playfair.c    [ –o   playfair.exe ]
[s@localhost ~]$ ./a.out                        [playfair]
PLAYFAIR CIPHER
KEY = monarchy
PLAIN TEXT= instruments
Encrypted String(CT) is: gatlmzclrqtx
 Decrypted String is(PT): instruments..

            Enter the text : rocket
1       r
2       o
3       c
4       k
5       e
6       t
r o c k e
t a b d f
g h i l m
n p q s u
v w x y z
Enter the plain text : balloon
Final string       : dbiykhcwr
```

RESULT:

Thus the Playfair cipher substitution technique had been implemented successfully.

DATE:	**HILL CIPHER**
EXPT NO: **1**(c)	

AIM : To write a C++ program to implement Hill Cipher Technique.

ALGORITHM :

STEP 1. Enter the plain text.
STEP 2, Enter the matrix, named key matrix for encryption. The elements of the matrix will be randomly chosen and of modulo 26.
STEP 3. Plain text characters are multiplied by the encryption matrix.
STEP 4. Display the cipher text.
STEP 5. Find the inverse of the key matrix.
STEP 6. For decryption, multiply the characters of cipher text by key matrix to get plaintext and display it.

PROGRAM :

```
#include<iostream>
using namespace std;
int check(int);
int main(int argc,char **argv)
{
int l,i,j,temp1,k[3][3],p[3][1],c[3][1];
char ch;
cout<<"\nThis cipher has a key of length 9";
cout<<"\nEnter the 9 character key";
        for(i=0;i<3;i++)
        {
                for(j=0;j<3;j++)
                {
                scanf("%c",&ch);
                        if (65<=ch&&ch<=91)
                                k[i][j]=(int)ch%65;
                        else
                                k[i][j]=(int)ch%97;
                }
        }
        for(i=0;i<3;i++)
        {
                for(j=0;j<3;j++)
                {
                cout<<k[i][j]<<"  ";
                }
        cout<<endl;
        }
   cout<<"\nEnter the length of string to be encoded(without spaces). ";
   cin>>l;
   temp1=check(l);
```

```
    cout<<temp1;
    if(temp1>0)
    cout<<"You have to enter "<<temp1<<" bogus characters.";
    char pi[l+temp1];
    cout<<"\nEnter the string. ";
        for(i=-1;i<l+temp1;i++)
        {
        cin>>pi[i];
        }
    int temp2=l;
    int n=(l+temp1)/3;
    int temp3,k1;
    int flag=0;
    int count;
    cout<<"\n\nThe encoded cipher is : ";
        while(n>0)
        {
        count=0;
              for(i=flag;i<flag+3;i++)
              {
              if(65<=pi[i]&&pi[i]<=91)
                    temp3=(int)pi[i]%65;
              else
                    temp3=(int)pi[i]%97;
              p[count][0]=temp3;
              count=count+1;
              }
        for(i=0;i<3;i++)
              c[i][0]=0;
              for(i=0;i<3;i++)
              {
                    for(j=0;j<1;j++)
                    {
                          for (k1=0;k1<3;k1++)
                          c[i][j]+=k[i][k1]*p[k1][j];
                    }
              }
              for(i=0;i<3;i++)
              {
              c[i][0]=c[i][0]%26;
              printf("%c",(char)(c[i][0]+65));

              }
      n=n-1;
      flag=flag+3;
        }
}
```

```
int check(int x)
{
   int a,b,c;
   if(x%3==0)
      return 0;
   a=x/3;
   b=3*(a+1);
   c=b-x;
   return c;
}
```

Output:

```
[s@localhost ~]$ g++ hill.cpp   [–o hill.exe]
[s@localhost ~]$ ./a.out              hill
This cipher has a key of length 9
Enter the 9 character key heltingen
7   4  11
19  8  13
6   4  13
Enter the length of string to be encoded(without spaces). 10
You have to enter 2 bogus characters.
Enter the string. sanfoundryabc
The encoded cipher is : DNNJXVPLIAIE
```

RESULT:

Thus the hill cipher substitution technique had been implemented successfully in C++

DATE:	**VIGENERE CIPHER**
EXPT NO: **1**(d)	

AIM : To write a C++ program to implement Vigenere Cipher technique.

ALGORITHM :

STEP 1. Enter the plain text for encryption.
STEP 2. Enter the encryption key pharse.
STEP 3. Cipher text is obtained by modular addition of a key pharse and plain text.
STEP 4. For decryption to get the plaintext again the key pharse is modularly subtracted from the cipher text.
STEP 5. Display the cipher text and plaintext.

PROGRAM :

```
// Vigenere Cipher
#include <iostream>
#include <string>
using namespace std;
class Vigenere
{
    public:
        string key;
        Vigenere(string key)
        {
            for (int i = 0; i < key.size(); ++i)
            {
                if (key[i] >= 'A' && key[i] <= 'Z')
                    this->key += key[i];
                else if (key[i] >= 'a' && key[i] <= 'z')
                    this->key += key[i] + 'A' - 'a';
            }
        }
        string encrypt(string text)
        {
            string out;
            for (int i = 0, j = 0; i < text.length(); ++i)
            {
                char c = text[i];
             if (c >= 'a' && c <= 'z')
                    c += 'A' - 'a';
                else if (c < 'A' || c > 'Z')
                    continue;
                out += (c + key[j] - 2 * 'A') % 26 + 'A';
                j = (j + 1) % key.length();
            }
            return out;
       }
```

```
        string decrypt(string text)
        {
            string out;
            for (int i = 0, j = 0; i < text.length(); ++i)
            {
                char c = text[i];
                if (c >= 'a' && c <= 'z')
                    c += 'A' - 'a';
                else if (c < 'A' || c > 'Z')
                    continue;
                out += (c - key[j] + 26) % 26 + 'A';
                j = (j + 1) % key.length();
            }
            return out;
 }
};
int main()
{
    Vigenere cipher("VIGENERECIPHER");
    string original = "Beware of Dogs”;
    string encrypted = cipher.encrypt(original);
    string decrypted = cipher.decrypt(encrypted);
    cout<<"original"<< endl;
    cout<<"Encrypted: "<<"encrypted"<<endl;
    cout<<"Decrypted: "<<"decrypted"<<endl;
}
```

Output:

```
[s@localhost ~]$ g++ vignere.cpp
[s@localhost ~]$ ./a.out
Beware of Dogs
Encrypted:WMCEEIFJFWVZ
Decrypted:BEWAREOFDOGS
```

RESULT:

Thus, the Vigenere Cipher substitution technique had been implemented successfully.

DATE:	**RAIL FENCE TRANSPOSITION**
EXPT NO: 2(a)	

AIM : To write a C++ program to Implement Rail fence row & column transformation.

ALGORITHM :

STEP 1. Enter the plain text for encryption.
STEP 2. Enter the depth in integer (number of rows)
STEP 3. In the rail fence cipher, the plaintext is written downwards and diagonally on successive "rails" of an imaginary fence, then moving up when we reach the bottom rail.
STEP 4. When we reach the top rail, the message is written downwards again until the whole plaintext is written out.
STEP 5. The message is then read off in rows. Thus, the cipher text is generated.

PROGRAM:

```
#include<iostream>
using namespace std;
int main()
{
        int i,j=0,d,k=0;
        char p[50],ct[50][50];
        cout<<"Enter the plain text:\n";
        cin>>p;
        cout<<"\nEnter the depth in the integer:";
        cin>>d;
        //declare null for empty array values
        for(i=0;i<50;i++)
        {
                for(j=0;j<50;j++)
               {
                        ct[i][j]='\0';
                }
        }
        k=0;
    //loop up to string lenght of the plaintext
        {
       for(i=0;i<strlen(p);i++)
        {
                for(j=0;j<d;j++)
                {
                        if(k<=strlen(p))
                    ct[i][j]=p[k];

                        k++;
                }
               ct[i][j]='\0';
                }
        }
        for(i=0;i<d;i++)
        {
        for(j=0;j<strlen(p);j++)
                {
                        if(ct[j][i]!='\0')
                         {
                           printf("%c",ct[j][i]);
                         }
                }
                cout<<"\n";
        }
        // Read the text
        cout<<"\nThe encrypted text is:\n";
        for(i=0;i<d;i++)
        {
                for(j=0;j<strlen(p);j++)
                {
                        if(ct[j][i]!='\0')
                        cout<<ct[j][i];
                   }
        }
        return 0;
}
```

OUTPUT

```
[s@localhost ~]$ g++ railfence.cpp
[s@localhost ~]$ ./a.out
Enter the plain text:heltin
Enter the depth in the integer:2
hli
etn
The encrypted text is:
hlietn
```

RESULT:

Thus the rail fence algorithm had been executed successfully.

DATE:	**ROW AND COLUMN TRANSPOSITION**
EXPT NO: 2(b)	

AIM:

To write a C program to implement row, column transposition technique.

ALGORITHM:

STEP-1: Read the Plain text.
STEP-2: Arrange the plain text in row columnar matrix format.
STEP-3: Now read the keyword depending on the number of columns of the plain text.
STEP-4: Arrange the characters of the keyword in sorted order and the corresponding columns of the plain text.
STEP-5: Read the characters row wise or column wise in the former order to get the cipher text.

The Columnar Transposition Cipher is a form of transposition cipher just like Rail Fence Cipher. Columnar Transposition involves writing the plaintext out in rows, and then reading the ciphertext off in columns one by one.

PROGRAM:

```
// CPP program for illustrating
// Columnar Transposition Cipher
#include<bits/stdc++.h>
using namespace std;

// Key for Columnar Transposition
string const key = "HACK";
map<int,int> keyMap;

void setPermutationOrder()
{
        // Add the permutation order into map
        for(int i=0; i < key.length(); i++)
        {
                keyMap[key[i]] = i;
        }
}

// Encryption
string encryptMessage(string msg)
{
        int row,col,j;
        string cipher = "";

        /* calculate column of the matrix*/
        col = key.length();

        /* calculate Maximum row of the matrix*/
        row = msg.length()/col;

        if (msg.length() % col)
```

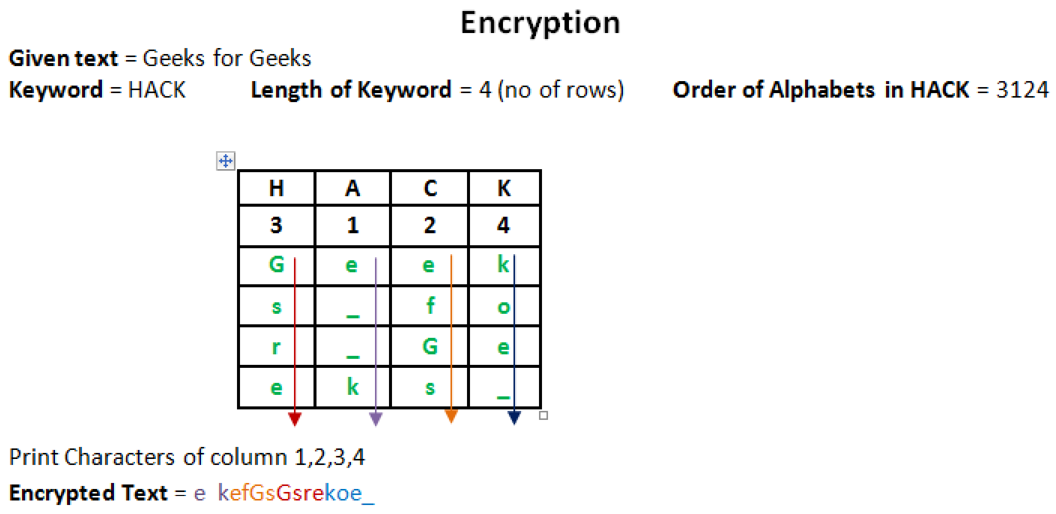
```
                row += 1;

        char matrix[row][col];

        for (int i=0,k=0; i < row; i++)
        {
                for (int j=0; j<col; )
                {
                        if(msg[k] == '\0')
                        {
                                /* Adding the padding character '_' */
                                matrix[i][j] = '_';
                                j++;
                        }

                        if( isalpha(msg[k]) || msg[k]==' ')
                        {
                                /* Adding only space and alphabet into matrix*/
                                matrix[i][j] = msg[k];
                                j++;
                        }
                        k++;
                }
        }

        for (map<int,int>::iterator ii = keyMap.begin(); ii!=keyMap.end(); ++ii)
        {
                j=ii->second;

                // getting cipher text from matrix column wise using permuted key
                for (int i=0; i<row; i++)
                {
                        if( isalpha(matrix[i][j]) || matrix[i][j]==' ' || matrix[i][j]=='_')
                                cipher += matrix[i][j];
                }
        }

        return cipher;
}

// Decryption
string decryptMessage(string cipher)
{
        /* calculate row and column for cipher Matrix */
        int col = key.length();

        int row = cipher.length()/col;
        char cipherMat[row][col];

        /* add character into matrix column wise */
```

```
		for (int j=0,k=0; j<col; j++)
			for (int i=0; i<row; i++)
				cipherMat[i][j] = cipher[k++];

		/* update the order of key for decryption */
		int index = 0;
		for( map<int,int>::iterator ii=keyMap.begin(); ii!=keyMap.end(); ++ii)
			ii->second = index++;

		/* Arrange the matrix column wise according
		to permutation order by adding into new matrix */
		char decCipher[row][col];
		map<int,int>::iterator ii=keyMap.begin();
		int k = 0;
		for (int l=0,j; key[l]!='\0'; k++)
		{
			j = keyMap[key[l++]];
			for (int i=0; i<row; i++)
			{
				decCipher[i][k]=cipherMat[i][j];
			}
		}

		/* getting Message using matrix */
		string msg = "";
		for (int i=0; i<row; i++)
		{
			for(int j=0; j<col; j++)
			{
				if(decCipher[i][j] != '_')
					msg += decCipher[i][j];
			}
		}
		return msg;
	}

	// Driver Program
	int main(void)
	{
		/* message */
		string msg = "Geeks for Geeks";

		setPermutationOrder();

		// Calling encryption function
		string cipher = encryptMessage(msg);
		cout << "Encrypted Message: " << cipher << endl;

		// Calling Decryption function
		cout << "Decrypted Message: " << decryptMessage(cipher) << endl;
```

```
        return 0;
}
```

OUTPUT:

```
[s@localhost ~]$ g++ rowcol.cpp –o rowcol.exe
[s@localhost ~]$ ./a.out             [rowcol]

PT:Geeks for Geeks
Key=HACK
Encrypted Message: e   kefGsGsrekoe
Decrypted Message: Geeks for Geeks
```



Encryption

Given text = Geeks for Geeks

Keyword = HACK **Length of Keyword** = 4 (no of rows) **Order of Alphabets in HACK** = 3124

H	A	C	K
3	1	2	4
G	e	e	k
s	_	f	o
r	_	G	e
e	k	s	_

Print Characters of column 1,2,3,4

Encrypted Text = e kefGsGsrekoe_

RESULT:

Thus the Row column transposition algorithm had been executed successfully.

DATE:	**IMPLEMENTATION OF DES**
EXPT NO: 3	

AIM : To write a java program to implement DES Cipher Encryption and decryption technique.

ALGORITHM :

```
 STEP 1. Enter the plain text and key
 STEP 2. Initial permutation (IP) will be done for the plain text and key.
 STEP 3. 16 rounds of a complex key dependent calculation f
          Function f  is
           L(i) = R(i-1)
         R(i) = L(i-1) Å P(S( E(R(i-1)) Å K(i) ))
STEP  4.  A final permutation, being the inverse of IP will be done to get cipher text.
STEP  5.  The above steps  4, 3, 2 are repeated in reverse order to get decrypted text.
```

PROGRAM :

```
import java.util.*;
import java.io.BufferedReader;
import java.io.InputStreamReader;
import java.security.spec.KeySpec;
import javax.crypto.Cipher;
import javax.crypto.SecretKey;
import javax.crypto.SecretKeyFactory;
import javax.crypto.spec.DESedeKeySpec;
import sun.misc.BASE64Decoder;
import sun.misc.BASE64Encoder;
public class DES {
private static final String UNICODE_FORMAT = "UTF8";
public static final String DESEDE_ENCRYPTION_SCHEME = "DESede";
private KeySpec myKeySpec;
private SecretKeyFactory mySecretKeyFactory;
private Cipher cipher;
byte[] keyAsBytes;
private String myEncryptionKey;
private String myEncryptionScheme;
SecretKey key;
static BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
public DES() throws Exception {
 // TODO code application logic here
myEncryptionKey = "ThisIsSecretEncryptionKey";
myEncryptionScheme = DESEDE_ENCRYPTION_SCHEME;
keyAsBytes = myEncryptionKey.getBytes(UNICODE_FORMAT);
myKeySpec = new DESedeKeySpec(keyAsBytes);
mySecretKeyFactory = SecretKeyFactory.getInstance(myEncryptionScheme);
cipher = Cipher.getInstance(myEncryptionScheme);
key = mySecretKeyFactory.generateSecret(myKeySpec);
 }
public String encrypt(String unencryptedString) {
 String encryptedString = null;
try {
cipher.init(Cipher.ENCRYPT_MODE, key);
byte[] plainText = unencryptedString.getBytes(UNICODE_FORMAT);
byte[] encryptedText = cipher.doFinal(plainText);
 BASE64Encoder base64encoder = new BASE64Encoder();
encryptedString = base64encoder.encode(encryptedText); }
catch (Exception e) {
e.printStackTrace(); }
return encryptedString; }
public String decrypt(String encryptedString) {
 String decryptedText=null;

try {
cipher.init(Cipher.DECRYPT_MODE, key);
 BASE64Decoder base64decoder = new BASE64Decoder();
byte[] encryptedText = base64decoder.decodeBuffer(encryptedString);
byte[] plainText = cipher.doFinal(encryptedText);
decryptedText= bytes2String(plainText); }
catch (Exception e) {
e.printStackTrace(); }
return decryptedText; }
private static String bytes2String(byte[] bytes) {
StringBuffer stringBuffer = new StringBuffer();
for (int i = 0; i <bytes.length; i++) {
stringBuffer.append((char) bytes[i]); }
return stringBuffer.toString();
 }
public static void main(String args []) throws Exception {
System.out.print("Enter the string: ");
 DES myEncryptor= new DES();
 String stringToEncrypt = br.readLine();
 String encrypted = myEncryptor.encrypt(stringToEncrypt);
String decrypted = myEncryptor.decrypt(encrypted);
System.out.println("\nString To Encrypt: " +stringToEncrypt);
System.out.println("\nEncrypted Value : " +encrypted);
System.out.println("\nDecrypted Value : " +decrypted);
System.out.println("");
 }
}
```

OUTPUT:

```
Enter the string: Welcome
String To Encrypt: Welcome
Encrypted Value : BPQMwc0wKvg=
Decrypted Value : Welcome
```

RESULT:

Thus the data encryption standard algorithm had been implemented successfully using JAVA language.

DATE:	**IMPLEMENTATION OF AES**
EXPT NO:4	

AIM:

To write a Java program to implement Advanced Encryption Standard (AES-256).

ALGORITHM:

STEP-1: Read the 128-bit plain text with different key length 128/192/256 bits
AES uses 10 rounds for 128-bit keys, 12 rounds for 192-bit keys and 14 rounds for 256-bit keys.
STEP-2: 128 bit block mention 4 * 4 matrix of bytes.
STEP-3: In AES , word consist 4 bytes so each column and row of state array is word
STEP-4: Each round consist of 4 stage: Byte substitution , Shift row, Mix column, Add round key and last round only 3 stage except Mix column
STEP-5: Thus the encrypted 128-bit cipher text is obtained in this way.

PROGRAM:

```
import java.io.UnsupportedEncodingException;
import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;
import java.util.Arrays;
import java.util.Base64;
 import javax.crypto.Cipher;
import javax.crypto.spec.SecretKeySpec;

public class AES {

    private static SecretKeySpec secretKey;
    private static byte[] key;

    public static void setKey(String myKey)
    {
        MessageDigest sha = null;
        try {
            key = myKey.getBytes("UTF-8");
            sha = MessageDigest.getInstance("SHA-1");
            key = sha.digest(key);
            key = Arrays.copyOf(key, 16);
            secretKey = new SecretKeySpec(key, "AES");
        }
        catch (NoSuchAlgorithmException e) {
            e.printStackTrace();
        }
        catch (UnsupportedEncodingException e) {
            e.printStackTrace();
        }
    }

    public static String encrypt(String strToEncrypt, String secret)
    {
        try
        {
            setKey(secret);
            Cipher cipher = Cipher.getInstance("AES/ECB/PKCS5Padding");
            cipher.init(Cipher.ENCRYPT_MODE, secretKey);
            return
Base64.getEncoder().encodeToString(cipher.doFinal(strToEncrypt.getBytes("UTF-8")));
        }
        catch (Exception e)
        {
            System.out.println("Error while encrypting: " + e.toString());
        }
        return null;
    }

    public static String decrypt(String strToDecrypt, String secret)
    {
        try
        {
            setKey(secret);
            Cipher cipher = Cipher.getInstance("AES/ECB/PKCS5PADDING");
            cipher.init(Cipher.DECRYPT_MODE, secretKey);
            return new String(cipher.doFinal(Base64.getDecoder().decode(strToDecrypt)));
        }
        catch (Exception e)
        {
            System.out.println("Error while decrypting: " + e.toString());
        }
        return null;
    }

public static void main(String[] args)
{
    final String secretKey = "ssshhhhhhhhhhh!!!!";

    String originalString = "howtodoinjava.com";
    String encryptedString = AES.encrypt(originalString, secretKey) ;
    String decryptedString = AES.decrypt(encryptedString, secretKey) ;
     System.out.println(originalString);
    System.out.println(encryptedString);
    System.out.println(decryptedString);
}
}
```

OUTPUT:

```
$javac AES.java
$java -Xmx128M -Xms16M AES
howtodoinjava.com
Tg2Nn7wUZOQ6Xc+1lenkZTQ9ZDf9a2/RBRiqJBCIX6o=
howtodoinjava.com
```

RESULT:

Thus the Advanced encryption standard algorithm had been implemented successfully using JAVA language

DATE:	**IMPLEMENTATION OF RSA**
EXPT NO:5	

AIM:

To write a C program to implement RSA. (Rivest–Shamir–Adleman)

ALGORITHM:

STEP-1: Select two co-prime numbers as p and q.
STEP-2: Compute n as the product of p and q.
STEP-3: Compute (p-1)*(q-1) and store it in z.
STEP-4: Select a random prime number e that is less than that of z.
STEP-5: Compute the private key, d as e * mod^{-1}(z).
STEP-6: The cipher text is computed as message^{e}* mod n.
STEP-7: Decryption is done as cipher^{d}mod n.

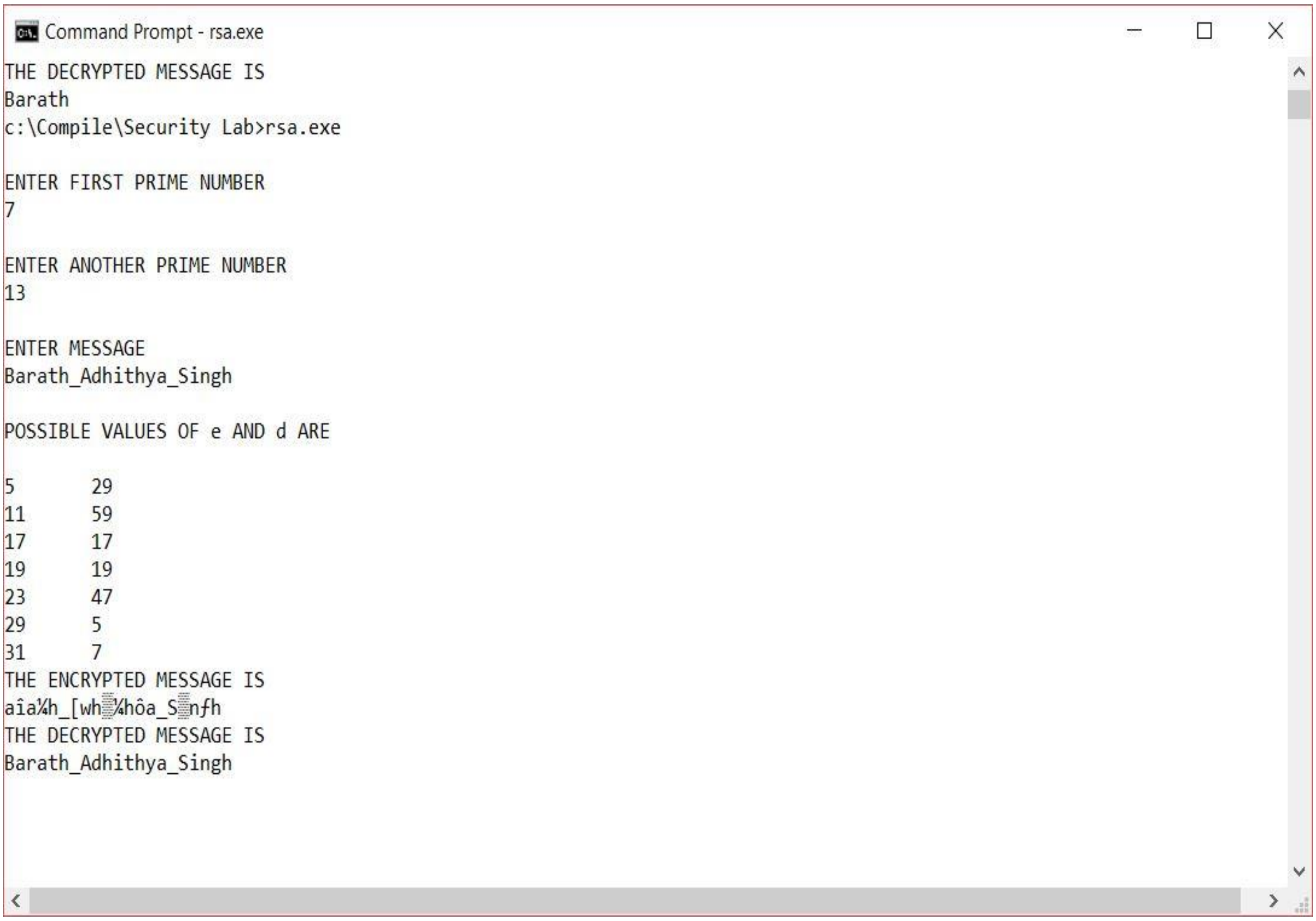
PROGRAM:

```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
#include<math.h>
#include<string.h>
long intp,q,n,t,flag,e[100],d[100],temp[100],j,m[100],en[100],i;
char msg[100];
int prime(long int);
void ce();
long int cd(long int);
void encrypt();
void decrypt();
void main()
{
printf("\nENTER FIRST PRIME NUMBER\n");
scanf("%d",&p);
flag=prime(p);
if(flag==0)
{
printf("\nWRONG INPUT\n");
getch();
}
printf("\nENTER ANOTHER PRIME NUMBER\n");
scanf("%d",&q);
flag=prime(q);
if(flag==0||p==q)
{
printf("\nWRONG INPUT\n");
getch();
}
printf("\nENTER MESSAGE\n");
fflush(stdin);
scanf("%s",msg);
for(i=0;msg[i]!=NULL;i++)
m[i]=msg[i];
n=p*q;
```

```
t=(p-1)*(q-1);
ce();
printf("\nPOSSIBLE VALUES OF e AND d ARE\n");
for(i=0;i<j-1;i++)
printf("\n%ld\t%ld",e[i],d[i]);
encrypt();
decrypt();
getch();
}
int prime(long int pr)
{
int i;
j=sqrt(pr);
for(i=2;i<=j;i++)
{
if(pr%i==0)
return 0;
}
return 1;
}
void ce()
{
int k;
k=0;
for(i=2;i<t;i++)
{
if(t%i==0)
continue;
flag=prime(i);
if(flag==1&&i!=p&&i!=q)
{
e[k]=i;
flag=cd(e[k]);
if(flag>0)
{
d[k]=flag;
k++;
}
if(k==99)
break;
} } }
long int cd(long int x)
{
long int k=1;
while(1)
{
k=k+t;
if(k%x==0)
return(k/x);
} }
void encrypt() {
long int pt,ct,key=e[0],k,len;
i=0;
```

```
len=strlen(msg);
while(i!=len) {
pt=m[i];
pt=pt-96;
k=1;
for(j=0;j<key;j++)
{ k=k*pt;
k=k%n;
}
temp[i]=k;
ct=k+96;
en[i]=ct;
i++;
}
en[i]=-1;
printf("\nTHE ENCRYPTED MESSAGE IS\n");
for(i=0;en[i]!=-1;i++)
printf("%c",en[i]);
}
void decrypt()
{
long int pt,ct,key=d[0],k;
i=0;
while(en[i]!=-1)
{
ct=temp[i];
k=1;
for(j=0;j<key;j++)
{
k=k*ct;
k=k%n;
}
pt=k+96;
m[i]=pt;
i++;
}
m[i]=-1;
printf("\nTHE DECRYPTED MESSAGE IS\n");
for(i=0;m[i]!=-1;i++)
printf("%c",m[i]);
}
```

OUTPUT:



```
THE DECRYPTED MESSAGE IS
Barath
c:\Compile\Security Lab>rsa.exe

ENTER FIRST PRIME NUMBER
7

ENTER ANOTHER PRIME NUMBER
13

ENTER MESSAGE
Barath_Adhithya_Singh

POSSIBLE VALUES OF e AND d ARE

5       29
11      59
17      17
19      19
23      47
29      5
31      7
THE ENCRYPTED MESSAGE IS
aîa¼h_[wh▒¼hôa_S▒nƒh
THE DECRYPTED MESSAGE IS
Barath_Adhithya_Singh
```

RESULT:

Thus the C program to implement RSA encryption technique had been implementedsuccessfully

DATE:	**IMPLEMENTATION OF DIFFIE HELLMAN KEY EXCHANGE**
EXPT NO: 6	

AIM:

To implement the Diffie-Hellman Key Exchange algorithm using C language.

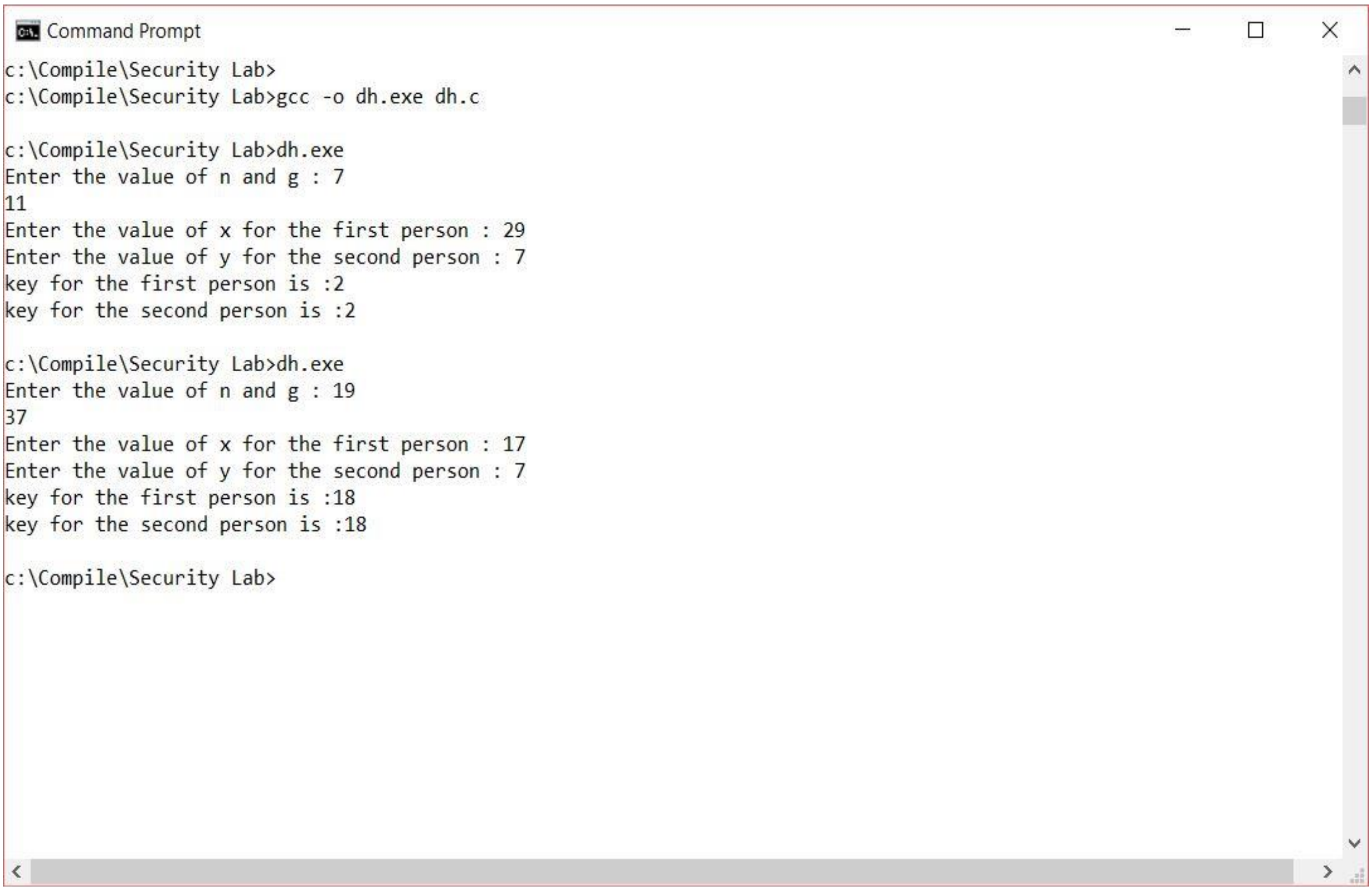
ALGORITHM:

STEP-1: Both Alice and Bob shares the same public keys g and p.
STEP-2: Alice selects a random public key a.
STEP-3: Alice computes his secret key A as $g_a$mod p.
STEP-4: Then Alice sends A to Bob.
STEP-5: Similarly Bob also selects a public key b and computes his secret key as B and sends the same back to Alice.
STEP-6: Now both of them compute their common secret key as the other one's secret key power of a mod p.

PROGRAM:

```
#include<stdio.h>
#include<conio.h>
long long int power(int a, int b, int mod)
{
long long int t;
if(b==1)
return a;
t=power(a,b/2,mod);
if(b%2==0)
return (t*t)%mod;
else
return (((t*t)%mod)*a)%mod;
}
long int calculateKey(int a, int x, int n)
{
return power(a,x,n);
}
void main()
{
int n,g,x,a,y,b;
printf("Enter the value of n and g : ");
scanf("%d%d",&n,&g);
printf("Enter the value of x for the first person : ");
scanf("%d",&x);
a=power(g,x,n);
printf("Enter the value of y for the second person : ");
scanf("%d",&y);
b=power(g,y,n);
printf("key for the first person is :%lld\n",power(b,x,n));
printf("key for the second person is :%lld\n",power(a,y,n));
}
```

OUTPUT:



```
c:\Compile\Security Lab>
c:\Compile\Security Lab>gcc -o dh.exe dh.c

c:\Compile\Security Lab>dh.exe
Enter the value of n and g : 7
11
Enter the value of x for the first person : 29
Enter the value of y for the second person : 7
key for the first person is :2
key for the second person is :2

c:\Compile\Security Lab>dh.exe
Enter the value of n and g : 19
37
Enter the value of x for the first person : 17
Enter the value of y for the second person : 7
key for the first person is :18
key for the second person is :18

c:\Compile\Security Lab>
```

RESULT:

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

DATE:	**IMPLEMENTATION OF SHA I**
EXPT NO:7	

AIM:

To implement the SHA – I hashing technique using Java Program.

ALGORITHM:

STEP-1: Read the 256-bit key values.

STEP-2: Divide into five equal-sized blocks named A, B, C, D and E.

STEP-3: The blocks B, C and D are passed to the function F.

STEP-4: The resultant value is permuted with block E.

STEP-5: The block A is shifted right by ‘s’ times and permuted with the result of step-4.

STEP-6: Then it is permuted with a weight value and then with some other key pair and taken as the first block.

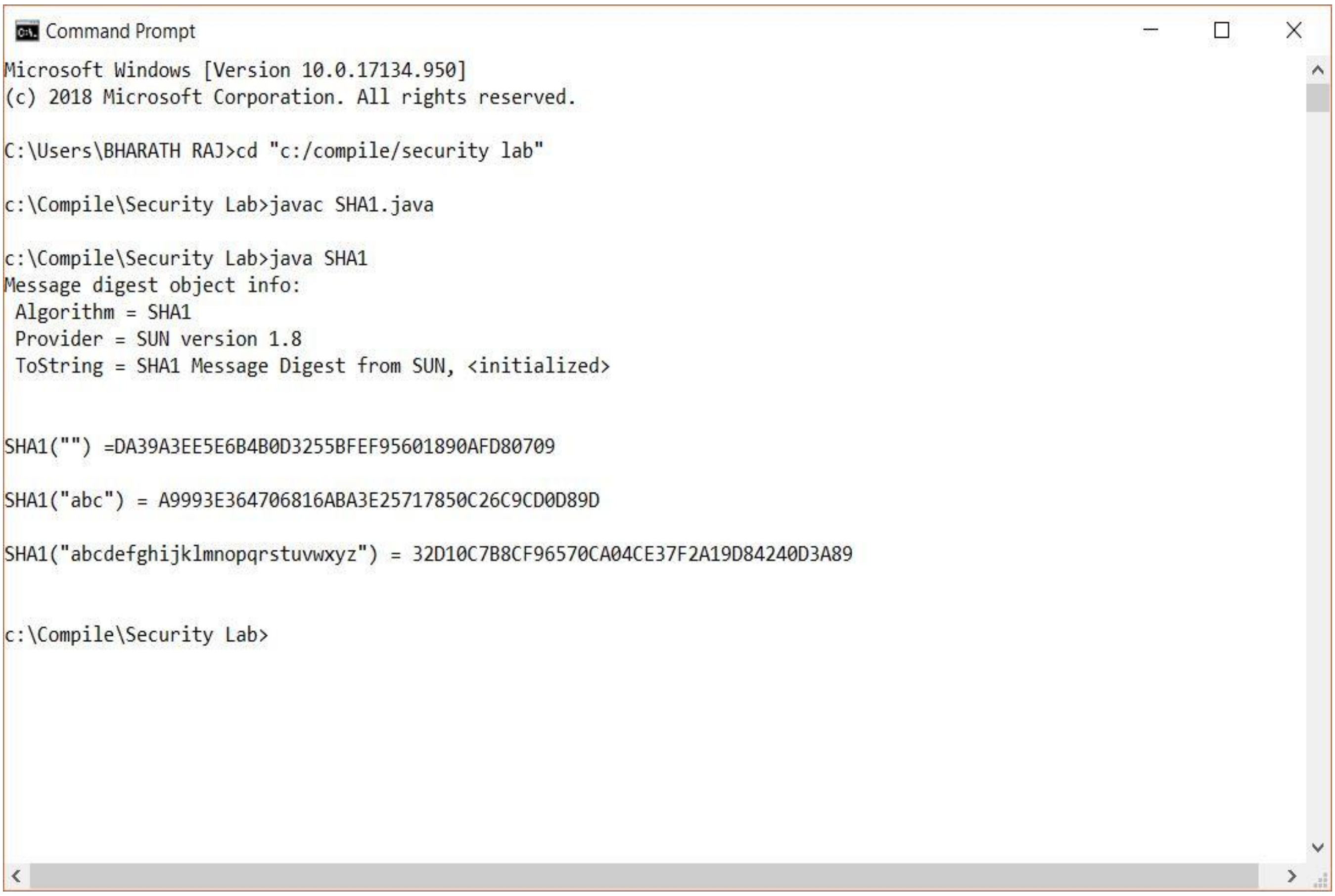
STEP-7: Block A is taken as the second block and the block B is shifted by ‘s’ times and taken as the third block.

STEP-8: The blocks C and D are taken as the block D and E for the final output.

PROGRAM: (Secure Hash Algorithm)

```
import java.security.*;
public class SHA1 {
public static void main(String[] a) {
try {
MessageDigest md = MessageDigest.getInstance("SHA1");
System.out.println("Message digest object info: ");
System.out.println(" Algorithm = " +md.getAlgorithm());
System.out.println(" Provider = " +md.getProvider());
System.out.println(" ToString = " +md.toString());
 String input = "";
md.update(input.getBytes());
byte[] output = md.digest();
System.out.println();
System.out.println("SHA1(\""+input+"\") = " +bytesToHex(output));
input = "abc";
md.update(input.getBytes());
output = md.digest();
System.out.println();
System.out.println("SHA1(\""+input+"\") = " +bytesToHex(output));
input = "abcdefghijklmnopqrstuvwxyz";
md.update(input.getBytes());
output = md.digest();
System.out.println();
System.out.println("SHA1(\"" +input+"\") = " +bytesToHex(output));
System.out.println(""); }
catch (Exception e) {
System.out.println("Exception: " +e);
 }
 }
public static String bytesToHex(byte[] b) {
 char hexDigit[] = {'0', '1', '2', '3', '4', '5', '6', '7', '8', '9', 'A', 'B', 'C', 'D', 'E', 'F'};
StringBuffer buf = new StringBuffer();
for (int j=0; j<b.length; j++) {
buf.append(hexDigit[(b[j] >> 4) & 0x0f]);
buf.append(hexDigit[b[j] & 0x0f]); }
return buf.toString();
}}
```

OUTPUT:



```
Microsoft Windows [Version 10.0.17134.950]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\BHARATH RAJ>cd "c:/compile/security lab"

c:\Compile\Security Lab>javac SHA1.java

c:\Compile\Security Lab>java SHA1
Message digest object info:
 Algorithm = SHA1
 Provider = SUN version 1.8
 ToString = SHA1 Message Digest from SUN, <initialized>


SHA1("") =DA39A3EE5E6B4B0D3255BFEF95601890AFD80709

SHA1("abc") = A9993E364706816ABA3E25717850C26C9CD0D89D

SHA1("abcdefghijklmnopqrstuvwxyz") = 32D10C7B8CF96570CA04CE37F2A19D84240D3A89


c:\Compile\Security Lab>
```

RESULT:

Thus the implementation of SHA1 using Java is implemented.

DATE:	**IMPLEMENTATION OF MD5 HASHING TECHNIQUE**
EXPT NO: 8	

AIM:

To implement a MD5 Hashing Technique using C Program.

ALGORITHM:

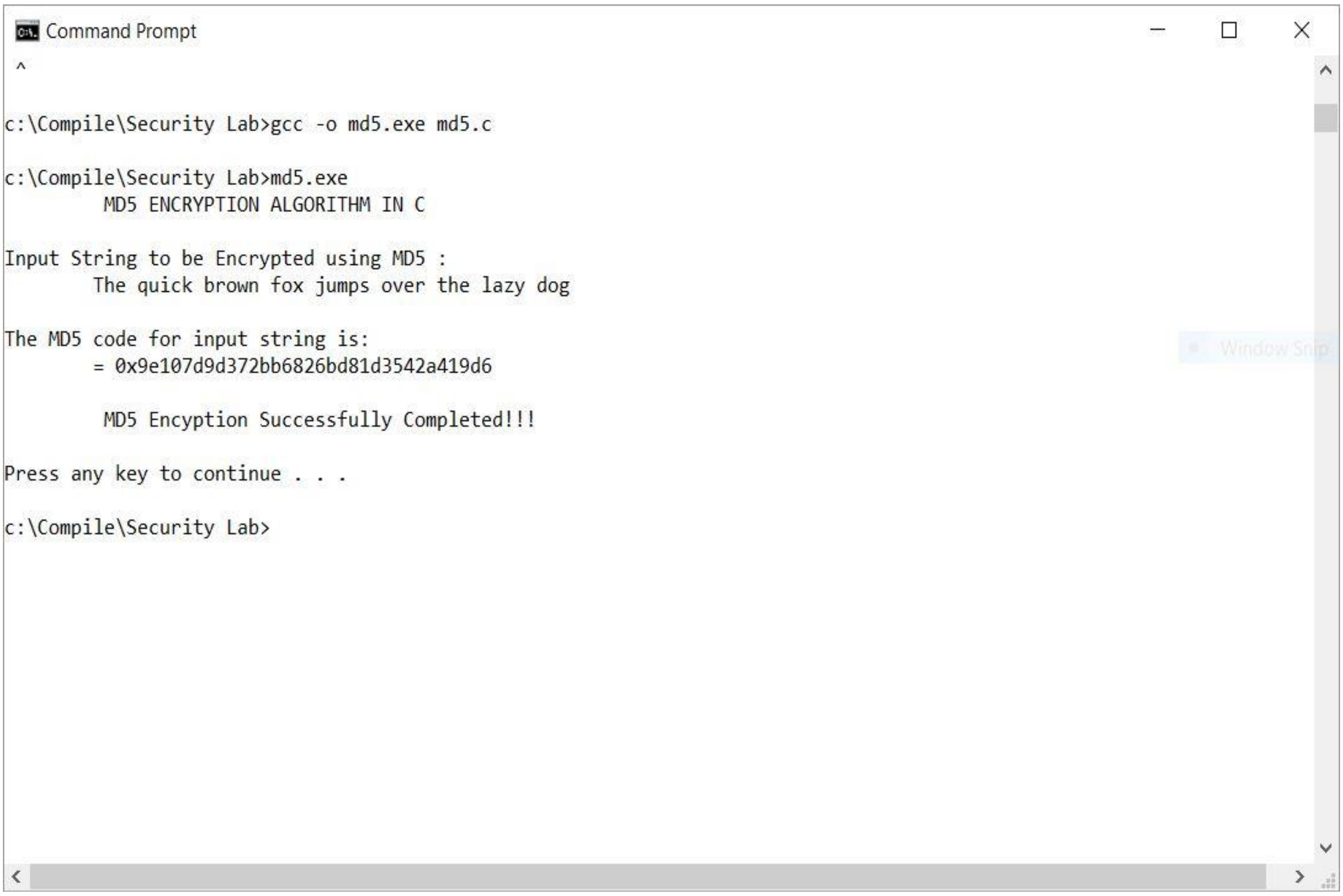
STEP-1: Read the 128-bit plain text.
STEP-2: Divide into four blocks of 32-bits named as A, B, C and D.
STEP-3: Compute the functions f, g, h and i with operations such as, rotations, permutations, etc,.
STEP-4: The output of these functions are combined together as F and performed circular shifting and then given to key round.
STEP-5: Finally, right shift of ‘s’ times are performed and the results are combined together to produce the final output.

PROGRAM:

```
#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#include <math.h>
#include<conio.h>
typedef union uwb
{
unsigned w;
unsigned char b[4];
} MD5union;
typedef unsigned DigestArray[4];
unsigned func0( unsignedabcd[] ){
return ( abcd[1] &abcd[2]) | (~abcd[1] &abcd[3]);}
unsigned func1( unsignedabcd[] ){
return ( abcd[3] &abcd[1]) | (~abcd[3] &abcd[2]);}
unsigned func2( unsignedabcd[] ){
return abcd[1] ^ abcd[2] ^ abcd[3];}
unsigned func3( unsignedabcd[] ){
return abcd[2] ^ (abcd[1] |~ abcd[3]);}
typedef unsigned (*DgstFctn)(unsigned a[]);
unsigned *calctable( unsigned *k)
{
double s, pwr;
int i;
pwr = pow( 2, 32);
for (i=0; i<64; i++)
{
s = fabs(sin(1+i));
k[i] = (unsigned)( s * pwr );
}
return k;
}
unsigned rol( unsigned r, short N )
{
unsigned mask1 = (1<<N) -1;
return ((r>>(32-N)) & mask1) | ((r<<N) & ~mask1);
```

```
}
unsigned *md5( const char *msg, int mlen)
{
static DigestArray h0 = { 0x67452301, 0xEFCDAB89,
0x98BADCFE, 0x10325476 };
static DgstFctnff[] = { &func0, &func1, &func2, &func3};
static short M[] = { 1, 5, 3, 7 };
static short O[] = { 0, 1, 5, 0 };
static short rot0[] = { 7,12,17,22};
static short rot1[] = { 5, 9,14,20};
static short rot2[] = { 4,11,16,23};
static short rot3[] = { 6,10,15,21};
static short *rots[] = {rot0, rot1, rot2, rot3 };
static unsigned kspace[64];
static unsigned *k;
static DigestArray h;
DigestArrayabcd;
DgstFctnfctn;
short m, o, g;
unsigned f;
short *rotn;
union
{
unsigned w[16];
char b[64];
}mm;
int os = 0;
int grp, grps, q, p;
unsigned char *msg2;
if (k==NULL) k= calctable(kspace);
for (q=0; q<4; q++) h[q] = h0[q]; // initialize
{
grps = 1 + (mlen+8)/64;
msg2 = malloc( 64*grps);
memcpy( msg2, msg, mlen);
msg2[mlen] = (unsigned char)0x80;
q = mlen + 1;
while (q < 64*grps){ msg2[q] = 0; q++ ; }
{
MD5union u;
u.w = 8*mlen;
q -= 8;
memcpy(msg2+q, &u.w, 4 );
}
}
for (grp=0; grp<grps; grp++)
{
memcpy( mm.b, msg2+os, 64);
for(q=0;q<4;q++) abcd[q] = h[q];
for (p = 0; p<4; p++)
{
fctn = ff[p];
rotn = rots[p];
m = M[p]; o= O[p];
for (q=0; q<16; q++)
{
g = (m*q + o) % 16;
f = abcd[1] + rol( abcd[0]+ fctn(abcd)+k[q+16*p]
+ mm.w[g], rotn[q%4]);
abcd[0] = abcd[3];
abcd[3] = abcd[2];
abcd[2] = abcd[1];
abcd[1] = f;
}}
for (p=0; p<4; p++)
h[p] += abcd[p];
os += 64;
}
return h;}
void main()
{
int j,k;
const char *msg = "The quick brown fox jumps over the lazy dog";
unsigned *d = md5(msg, strlen(msg));
MD5union u;
printf("\t MD5 ENCRYPTION ALGORITHM IN C \n\n");
printf("Input String to be Encrypted using MD5 : \n\t%s",msg);
printf("\n\nThe MD5 code for input string is: \n");
printf("\t= 0x");
for (j=0;j<4; j++){
u.w = d[j];
for (k=0;k<4;k++) printf("%02x",u.b[k]);
}
printf("\n");
printf("\n\t MD5 Encyption Successfully Completed!!!\n\n");
system("pause");
}
```

OUTPUT:



```
 ^

c:\Compile\Security Lab>gcc -o md5.exe md5.c

c:\Compile\Security Lab>md5.exe
         MD5 ENCRYPTION ALGORITHM IN C

Input String to be Encrypted using MD5 :
        The quick brown fox jumps over the lazy dog

The MD5 code for input string is:
        = 0x9e107d9d372bb6826bd81d3542a419d6

         MD5 Encyption Successfully Completed!!!

Press any key to continue . . .

c:\Compile\Security Lab>
```

RESULT:

Thus the implementation of MD5 hashing algorithm had been implemented successfully using C.

DATE:	**IMPLEMENTATION OF DIGITAL SIGNATURE STANDARD**
EXPT NO: 9	

AIM:

To implement Digital Signature Scheme using java.

Algorithm:

Input the plain text
Get the Claimed Signatory’s Identifier.
Generate the Domain Parameters and Public Key
Generate a Message Digest
Verify the Digital Signature
Digital Signature Validation Complete

Coding:

```
import java.security.KeyPair;
import java.security.KeyPairGenerator;
import java.security.Signature;
import sun.misc.BASE64Encoder;
public class DigSign {
public static void main(String[] args) throws Exception {
 // TODO code application logic here
KeyPairGenerator kpg = KeyPairGenerator.getInstance("RSA");
kpg.initialize(1024);
KeyPair keyPair = kpg.genKeyPair();
byte[] data = "Sample Text".getBytes("UTF8");
 Signature sig = Signature.getInstance("MD5WithRSA");
sig.initSign(keyPair.getPrivate());
sig.update(data);
byte[] signatureBytes = sig.sign();
System.out.println("Signature: \n" + new BASE64Encoder().encode(signatureBytes));
sig.initVerify(keyPair.getPublic());
sig.update(data);
System.out.println(sig.verify(signatureBytes));
}
}
```

OUTPUT:

```
Signature:
imwaKe99tkM6H6hiiP0rubmb/MrYJZLiwLdRSjslF2KlA5B23az5M2LKftQFCB+NH
Ce5F5/YfN8OsNSNLtucrrZTah0SrdWSzdGCOfYLdUZmPQ72j1SkLhYspsTsUb/U6
FPSYT4QebNSYobDtjKujkHdRimHI9TO4lLuqVQRdWU= true
```

RESULT:

Thus the Digital Signature Standard using Java Programming is done and verified.

DATE:	**SECURE DATA TRANSMISSION FOR CREATING DIGITAL SIGNATURE**
EXPT NO: 10	

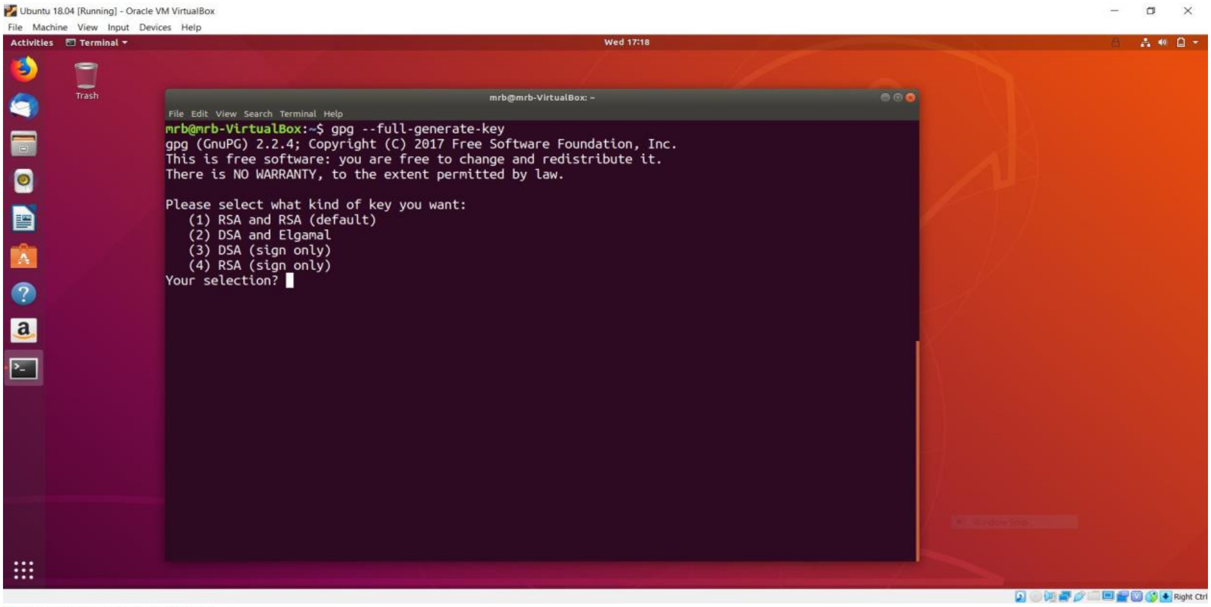
AIM:

To provide secure data storage, secure data transmission and for creating digital signatures (GnuPG).

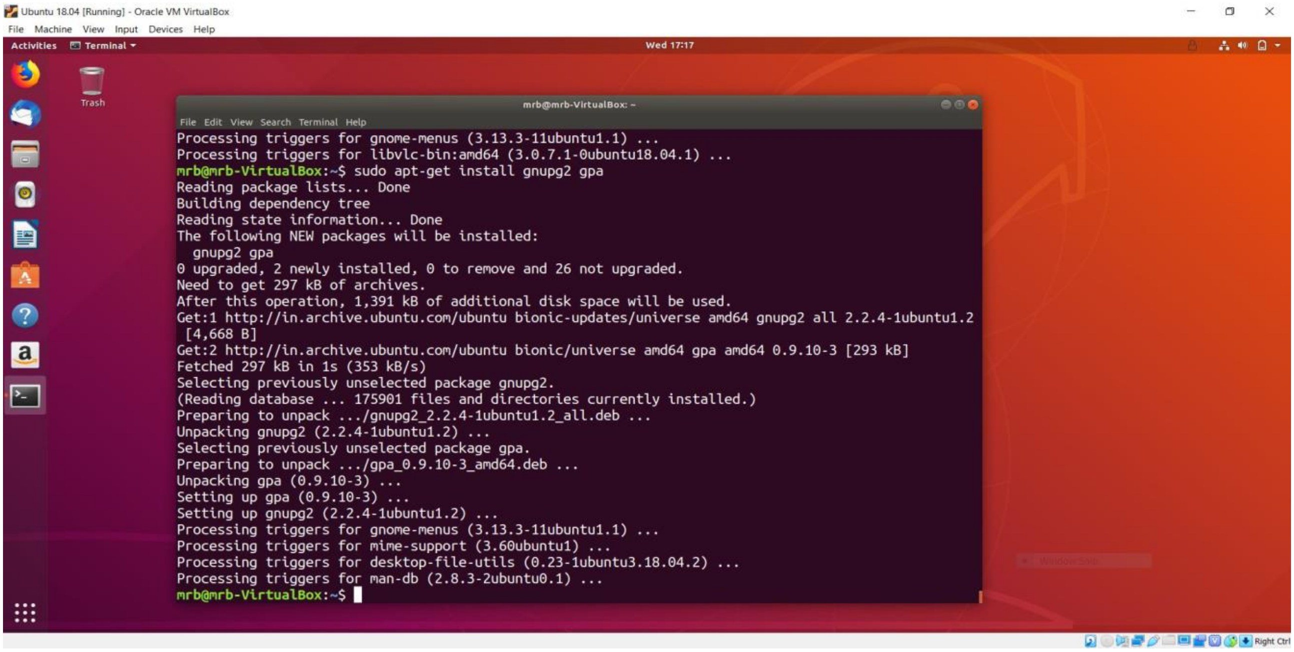
PROCEDURE:

Step1: Install GnuPG in Ubuntu Environment along with GPA Kleopatra.
Step2: Select Encryption Algorithm and choose the Validity of PassPhrase.
Provide details like Name, Mail ID to generate the PassPhrase.

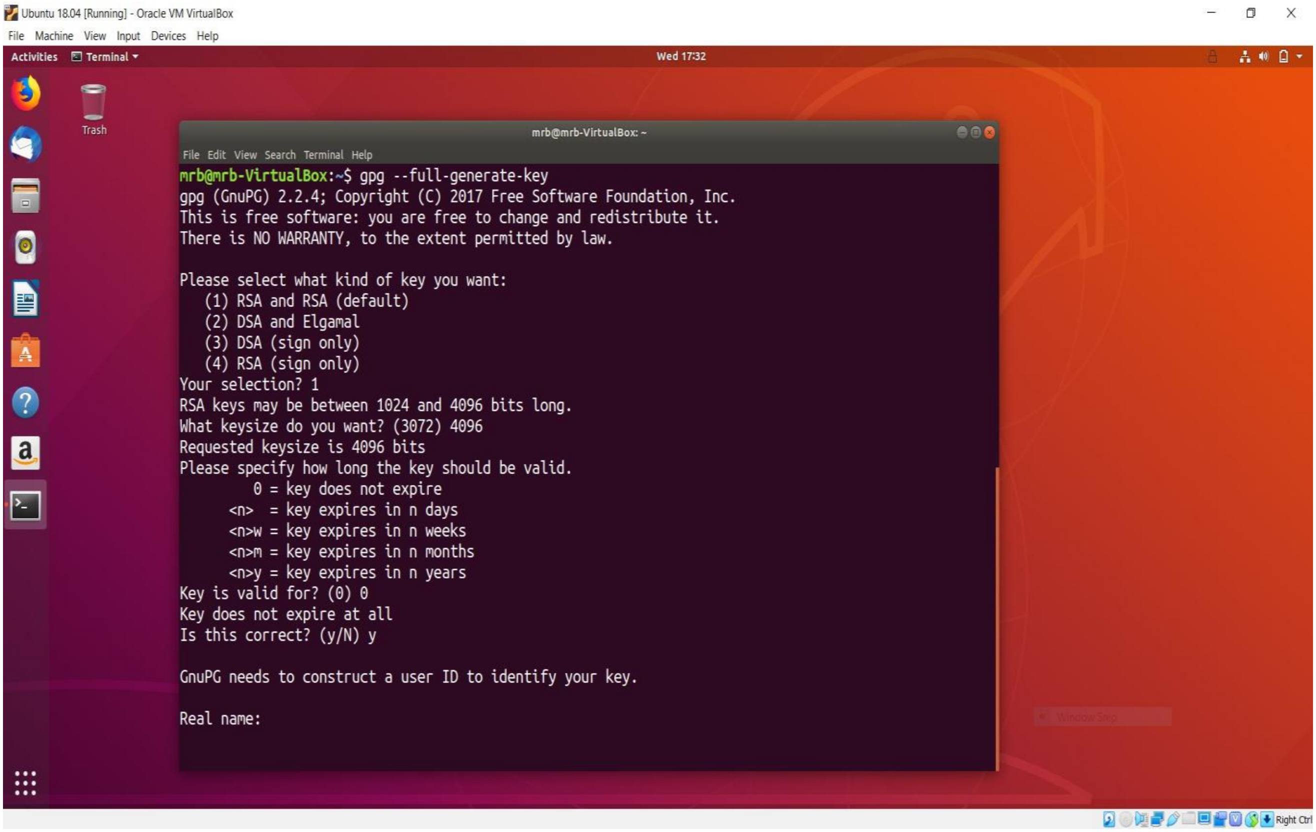
OUTPUT:


Ubuntu 18.04 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Terminal
Wed 17:18
Trash
mrb@mrb-VirtualBox: ~
File Edit View Search Terminal Help

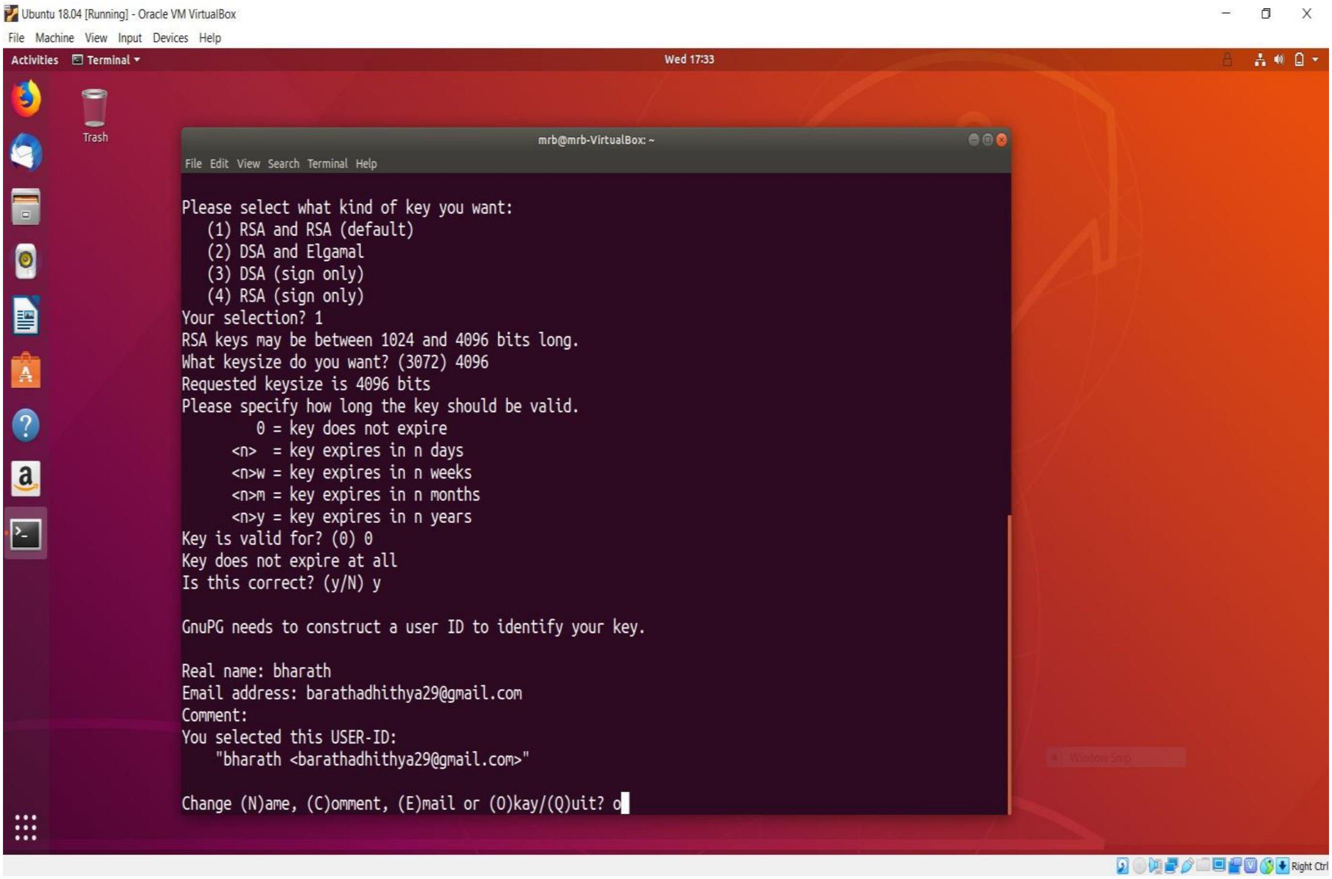
```
mrb@mrb-VirtualBox:~$ gpg --full-generate-key
gpg (GnuPG) 2.2.4; Copyright (C) 2017 Free Software Foundation, Inc.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Please select what kind of key you want:
   (1) RSA and RSA (default)
   (2) DSA and Elgamal
   (3) DSA (sign only)
   (4) RSA (sign only)
Your selection?
```


Ubuntu 18.04 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Terminal
Wed 17:17
Trash
mrb@mrb-VirtualBox: ~
File Edit View Search Terminal Help

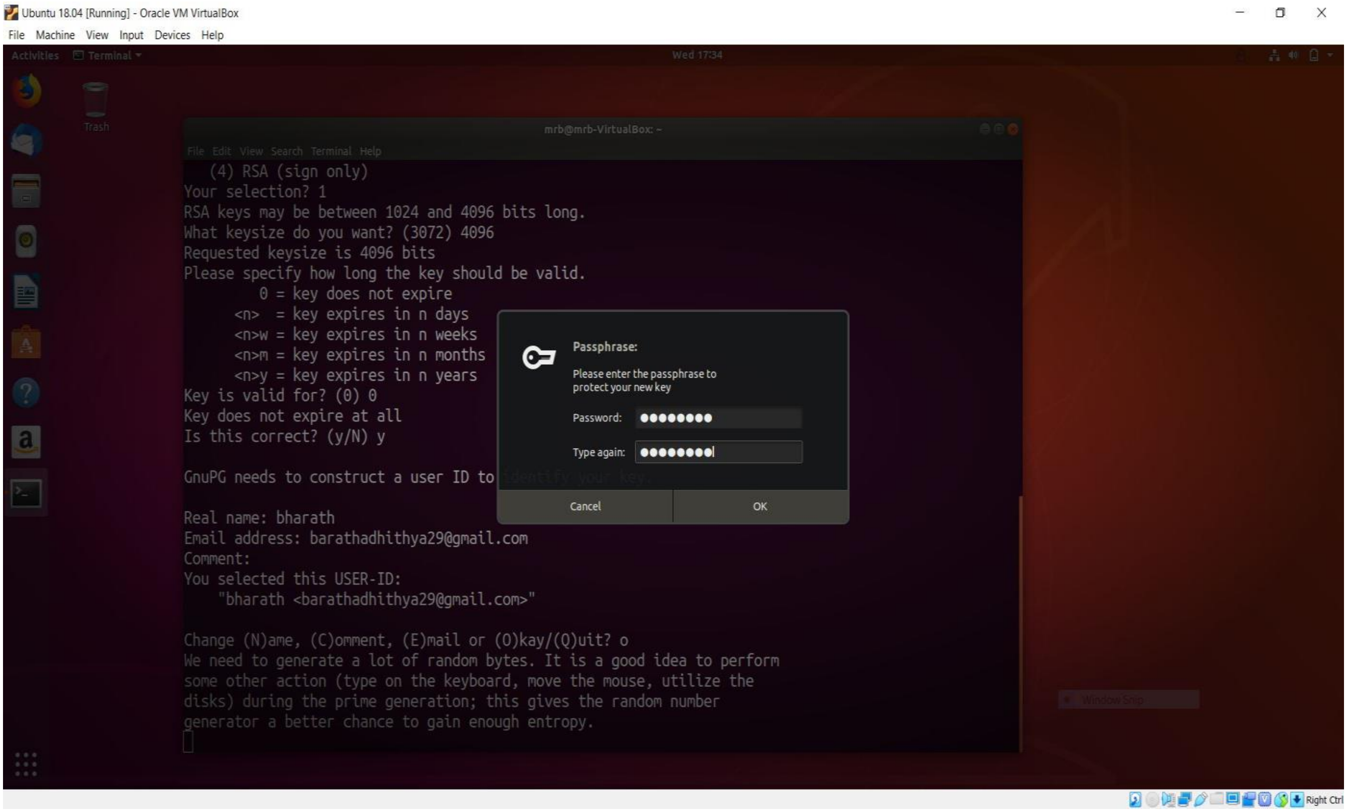
```
Processing triggers for gnome-menus (3.13.3-11ubuntu1.1) ...
Processing triggers for libvlc-bin:amd64 (3.0.7.1-0ubuntu18.04.1) ...
mrb@mrb-VirtualBox:~$ sudo apt-get install gnupg2 gpa
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  gnupg2 gpa
0 upgraded, 2 newly installed, 0 to remove and 26 not upgraded.
Need to get 297 kB of archives.
After this operation, 1,391 kB of additional disk space will be used.
Get:1 http://in.archive.ubuntu.com/ubuntu bionic-updates/universe amd64 gnupg2 all 2.2.4-1ubuntu1.2
 [4,668 B]
Get:2 http://in.archive.ubuntu.com/ubuntu bionic/universe amd64 gpa amd64 0.9.10-3 [293 kB]
Fetched 297 kB in 1s (353 kB/s)
Selecting previously unselected package gnupg2.
(Reading database ... 175901 files and directories currently installed.)
Preparing to unpack .../gnupg2_2.2.4-1ubuntu1.2_all.deb ...
Unpacking gnupg2 (2.2.4-1ubuntu1.2) ...
Selecting previously unselected package gpa.
Preparing to unpack .../gpa_0.9.10-3_amd64.deb ...
Unpacking gpa (0.9.10-3) ...
Setting up gpa (0.9.10-3) ...
Setting up gnupg2 (2.2.4-1ubuntu1.2) ...
Processing triggers for gnome-menus (3.13.3-11ubuntu1.1) ...
Processing triggers for mime-support (3.60ubuntu1) ...
Processing triggers for desktop-file-utils (0.23-1ubuntu3.18.04.2) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
mrb@mrb-VirtualBox:~$
```


Ubuntu 18.04 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Terminal
Wed 17:32
Trash
mrb@mrb-VirtualBox: ~
File Edit View Search Terminal Help
Right Ctrl

```
mrb@mrb-VirtualBox:~$ gpg --full-generate-key
gpg (GnuPG) 2.2.4; Copyright (C) 2017 Free Software Foundation, Inc.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Please select what kind of key you want:
   (1) RSA and RSA (default)
   (2) DSA and Elgamal
   (3) DSA (sign only)
   (4) RSA (sign only)
Your selection? 1
RSA keys may be between 1024 and 4096 bits long.
What keysize do you want? (3072) 4096
Requested keysize is 4096 bits
Please specify how long the key should be valid.
         0 = key does not expire
      <n>  = key expires in n days
      <n>w = key expires in n weeks
      <n>m = key expires in n months
      <n>y = key expires in n years
Key is valid for? (0) 0
Key does not expire at all
Is this correct? (y/N) y

GnuPG needs to construct a user ID to identify your key.

Real name:
```


Ubuntu 18.04 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Terminal
Wed 17:33
Trash
mrb@mrb-VirtualBox: ~
File Edit View Search Terminal Help
Right Ctrl

```
Please select what kind of key you want:
   (1) RSA and RSA (default)
   (2) DSA and Elgamal
   (3) DSA (sign only)
   (4) RSA (sign only)
Your selection? 1
RSA keys may be between 1024 and 4096 bits long.
What keysize do you want? (3072) 4096
Requested keysize is 4096 bits
Please specify how long the key should be valid.
         0 = key does not expire
      <n>  = key expires in n days
      <n>w = key expires in n weeks
      <n>m = key expires in n months
      <n>y = key expires in n years
Key is valid for? (0) 0
Key does not expire at all
Is this correct? (y/N) y

GnuPG needs to construct a user ID to identify your key.

Real name: bharath
Email address: barathadhithya29@gmail.com
Comment:
You selected this USER-ID:
    "bharath <barathadhithya29@gmail.com>"

Change (N)ame, (C)omment, (E)mail or (O)kay/(Q)uit? o
```


Ubuntu 18.04 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal
Wed 17:34
Trash
mrb@mrb-VirtualBox: ~
File Edit View Search Terminal Help
Cancel
OK
Right Ctrl

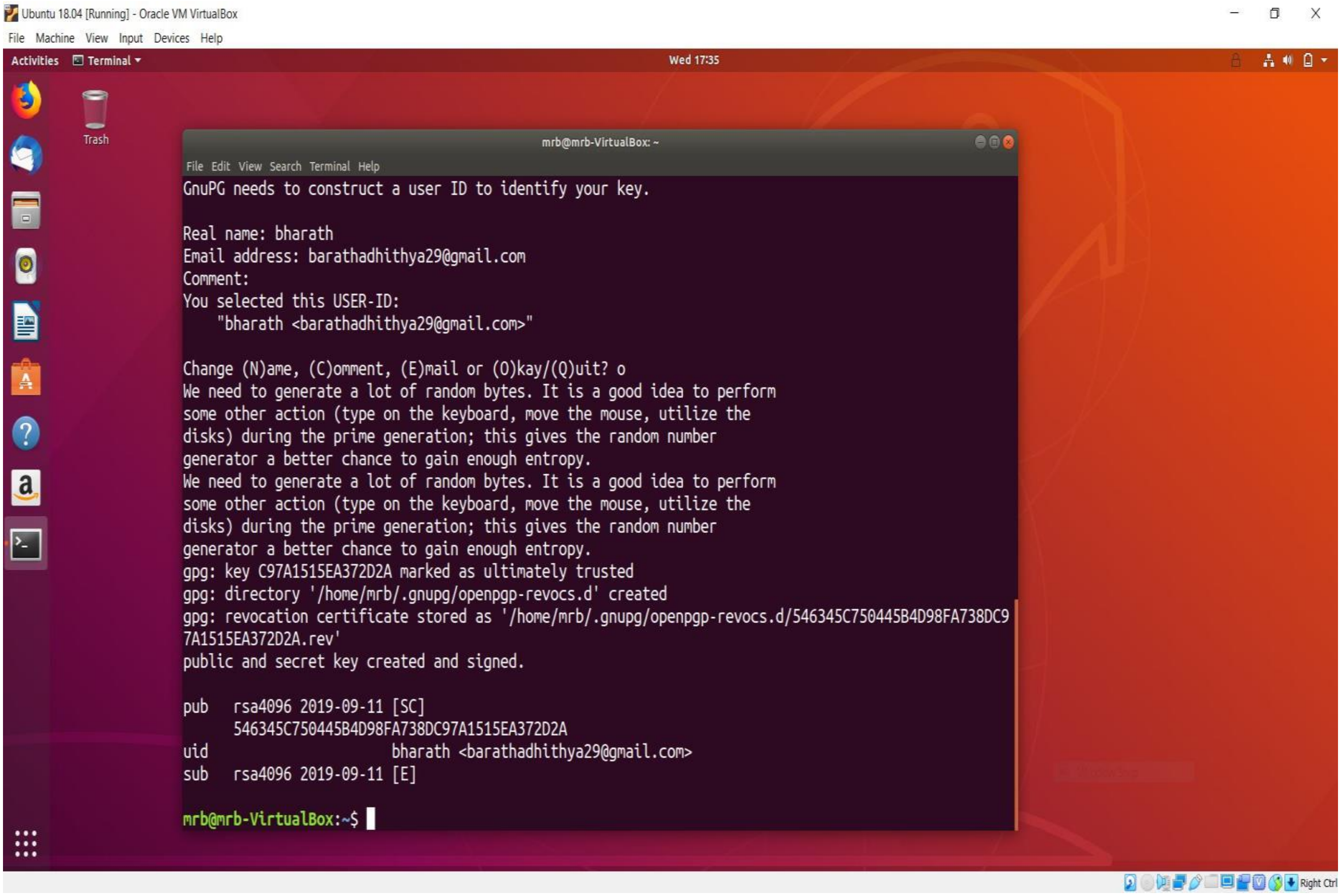
```
   (4) RSA (sign only)
Your selection? 1
RSA keys may be between 1024 and 4096 bits long.
What keysize do you want? (3072) 4096
Requested keysize is 4096 bits
Please specify how long the key should be valid.
         0 = key does not expire
      <n>  = key expires in n days
      <n>w = key expires in n weeks
      <n>m = key expires in n months
      <n>y = key expires in n years
Key is valid for? (0) 0
Key does not expire at all
Is this correct? (y/N) y

GnuPG needs to construct a user ID to identify your key.

Real name: bharath
Email address: barathadhithya29@gmail.com
Comment:
You selected this USER-ID:
    "bharath <barathadhithya29@gmail.com>"

Change (N)ame, (C)omment, (E)mail or (O)kay/(Q)uit? o
We need to generate a lot of random bytes. It is a good idea to perform
some other action (type on the keyboard, move the mouse, utilize the
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
```

Passphrase:

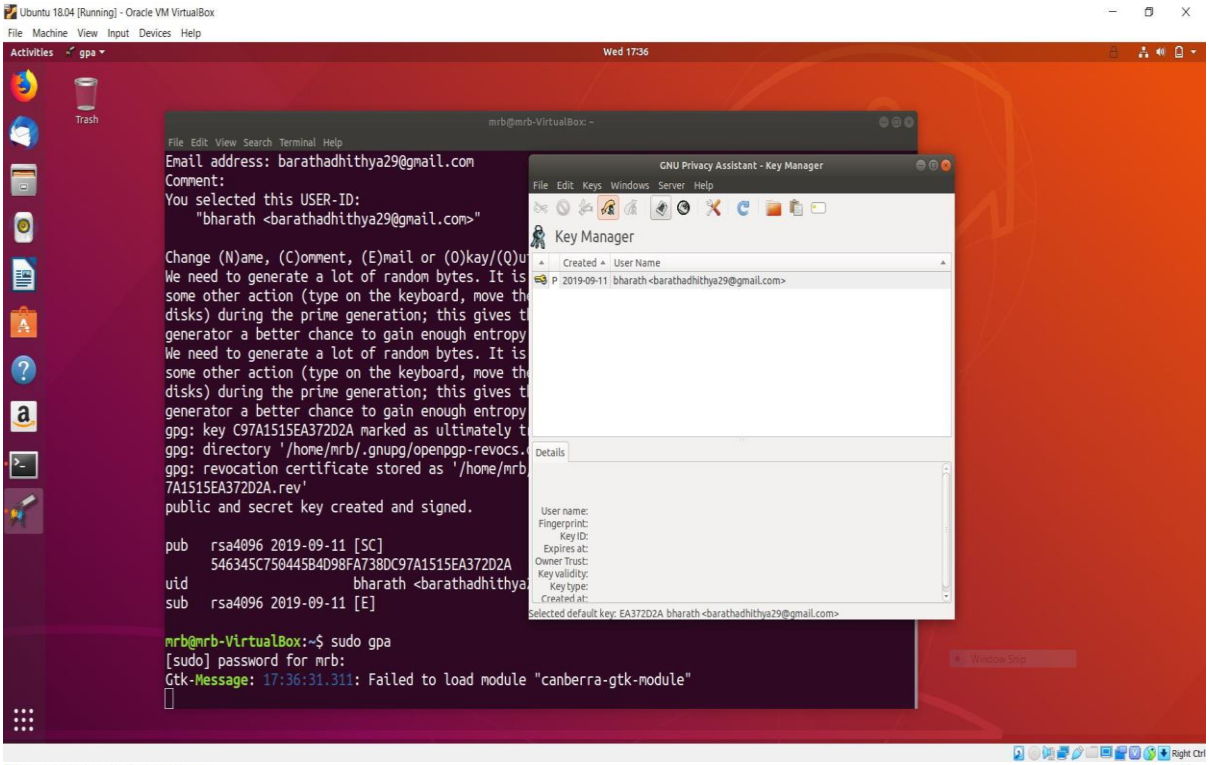
Please enter the passphrase to protect your new key

Password: ●●●●●●●●

Type again: ●●●●●●●●


Ubuntu 18.04 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal
Wed 17:35
Trash
mrb@mrb-VirtualBox: ~
File Edit View Search Terminal Help
Right Ctrl

```
GnuPG needs to construct a user ID to identify your key.

Real name: bharath
Email address: barathadhithya29@gmail.com
Comment:
You selected this USER-ID:
    "bharath <barathadhithya29@gmail.com>"

Change (N)ame, (C)omment, (E)mail or (O)kay/(Q)uit? o
We need to generate a lot of random bytes. It is a good idea to perform
some other action (type on the keyboard, move the mouse, utilize the
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
We need to generate a lot of random bytes. It is a good idea to perform
some other action (type on the keyboard, move the mouse, utilize the
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
gpg: key C97A1515EA372D2A marked as ultimately trusted
gpg: directory '/home/mrb/.gnupg/openpgp-revocs.d' created
gpg: revocation certificate stored as '/home/mrb/.gnupg/openpgp-revocs.d/546345C750445B4D98FA738DC9
7A1515EA372D2A.rev'
public and secret key created and signed.

pub   rsa4096 2019-09-11 [SC]
      546345C750445B4D98FA738DC97A1515EA372D2A
uid                      bharath <barathadhithya29@gmail.com>
sub   rsa4096 2019-09-11 [E]

mrb@mrb-VirtualBox:~$
```

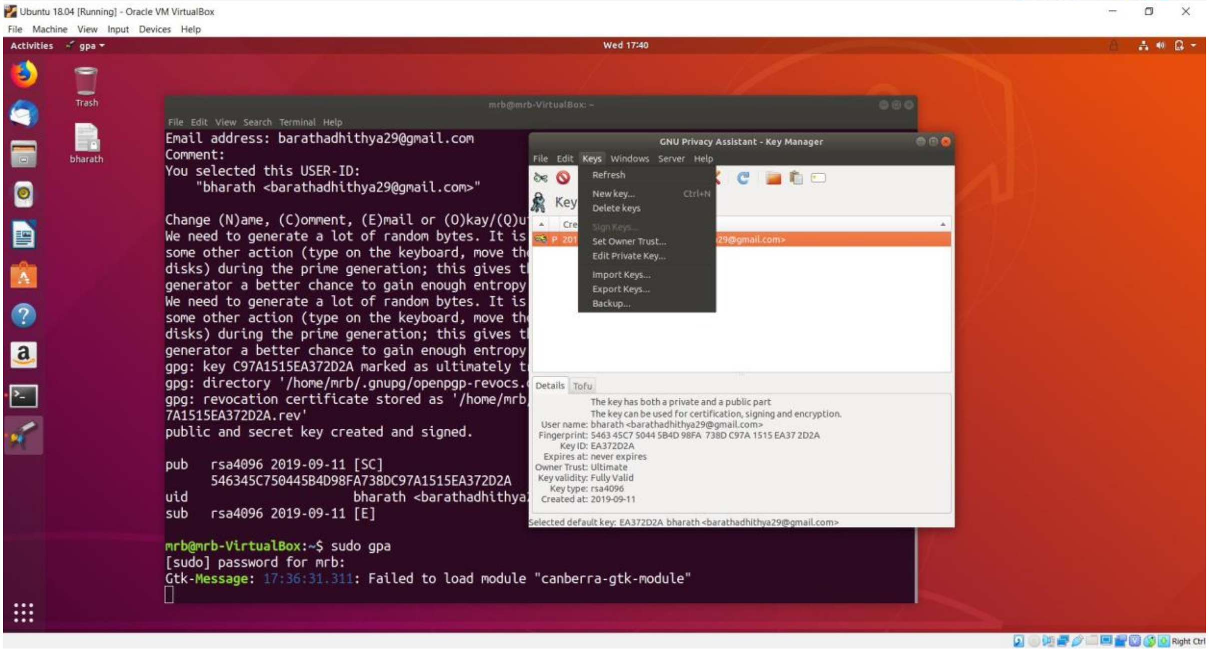

Ubuntu 18.04 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
gpa
Wed 17:36
Trash
mrb@mrb-VirtualBox: ~
File Edit View Search Terminal Help
GNU Privacy Assistant - Key Manager
File Edit Keys Windows Server Help
Key Manager
Created
User Name
Details
Right Ctrl

```
Email address: barathadhithya29@gmail.com
Comment:
You selected this USER-ID:
    "bharath <barathadhithya29@gmail.com>"

Change (N)ame, (C)omment, (E)mail or (O)kay/(Q)u
We need to generate a lot of random bytes. It is
some other action (type on the keyboard, move th
disks) during the prime generation; this gives t
generator a better chance to gain enough entropy
We need to generate a lot of random bytes. It is
some other action (type on the keyboard, move th
disks) during the prime generation; this gives t
generator a better chance to gain enough entropy
gpg: key C97A1515EA372D2A marked as ultimately t
gpg: directory '/home/mrb/.gnupg/openpgp-revocs.
gpg: revocation certificate stored as '/home/mrb
7A1515EA372D2A.rev'
public and secret key created and signed.

pub   rsa4096 2019-09-11 [SC]
      546345C750445B4D98FA738DC97A1515EA372D2A
uid                      bharath <barathadhithya
sub   rsa4096 2019-09-11 [E]

mrb@mrb-VirtualBox:~$ sudo gpa
[sudo] password for mrb:
Gtk-Message: 17:36:31.311: Failed to load module "canberra-gtk-module"
```

P 2019-09-11 bharath <barathadhithya29@gmail.com>

User name:
Fingerprint:
Key ID:
Expires at:
Owner Trust:
Key validity:
Key type:
Created at:

Selected default key: EA372D2A bharath <barathadhithya29@gmail.com>



```
Email address: barathadhithya29@gmail.com
Comment:
You selected this USER-ID:
    "bharath <barathadhithya29@gmail.com>"

Change (N)ame, (C)omment, (E)mail or (O)kay/(Q)u
We need to generate a lot of random bytes. It is
some other action (type on the keyboard, move th
disks) during the prime generation; this gives t
generator a better chance to gain enough entropy
We need to generate a lot of random bytes. It is
some other action (type on the keyboard, move th
disks) during the prime generation; this gives t
generator a better chance to gain enough entropy
gpg: key C97A1515EA372D2A marked as ultimately t
gpg: directory '/home/mrb/.gnupg/openpgp-revocs.
gpg: revocation certificate stored as '/home/mrb
7A1515EA372D2A.rev'
public and secret key created and signed.

pub   rsa4096 2019-09-11 [SC]
      546345C750445B4D98FA738DC97A1515EA372D2A
uid                      bharath <barathadhithya
sub   rsa4096 2019-09-11 [E]

mrb@mrb-VirtualBox:~$ sudo gpa
[sudo] password for mrb:
Gtk-Message: 17:36:31.311: Failed to load module "canberra-gtk-module"
```

P 201 ...29@gmail.com>

The key has both a private and a public part
The key can be used for certification, signing and encryption.
User name: bharath <barathadhithya29@gmail.com>
Fingerprint: 5463 45C7 5044 5B4D 98FA 738D C97A 1515 EA37 2D2A
Key ID: EA372D2A
Expires at: never expires
Owner Trust: Ultimate
Key validity: Fully Valid
Key type: rsa4096
Created at: 2019-09-11

Selected default key: EA372D2A bharath <barathadhithya29@gmail.com>

RESULT:

Thus the secure data storage, secure data transmission and for creating digital signatures (GnuPG) was generated successfully

DATE:	**DEMONSTRATE INTRUSION DETECTION SYSTEM (IDS) USING SNORT**
EXPT NO: 11	

AIM:

To demonstrate intrusion detection system using Snort

INSTALLATION PROCEDURE:

Step 1: Download SNORT from snort.org

Step 2: Install snort with or without database support.

Step 3: Select all the components and Click Next.

Step 4: Install and Close.

Step 5: Skip the WinPcap driver installation

Step 6: Add the path variable in windows environment variable by selecting new classpath.

Step 7: Create a path variable and point it at snort.exe variable name□path and variable value□c:\snort\bin.

Step 8: Click OK button and then close all dialog boxes.

Step 9: Open command prompt and type the commands.

STEPS:

SNORT can be configured to run in three modes:
1. Sniffer mode 2. Packet Logger mode 3. Network Intrusion Detection System mode

Sniffer mode

i. snort –v →Print out the TCP/IP packets header on the screen

ii. snort –vd→Show the TCP/IP ICMP header with application data in transit.

Packet Logger mode

i. snort –dev –l c:\log → snort will automatically know to go into packet logger mode, it collects every packet it sees and places it in log directory.

ii. snort –dev –l c:\log –h ipaddress/24 →This rule tells snort that you want to print out the data link and TCP/IP headers as well as application data into the log directory.

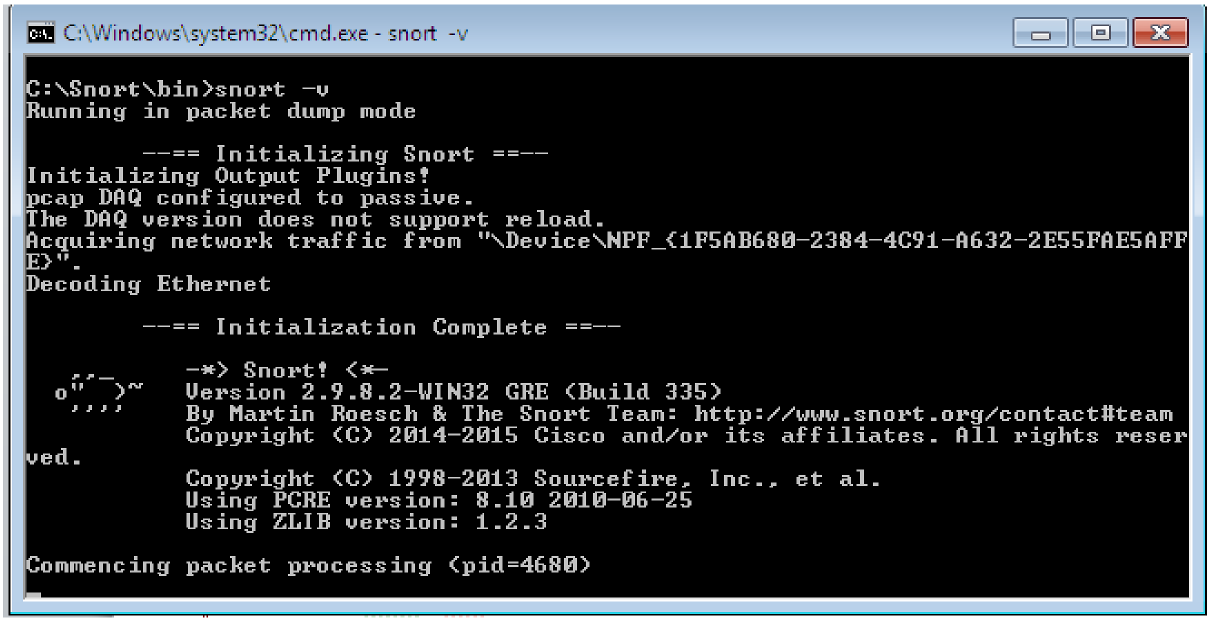
iii. snort –l c:\log –b →This is binary mode logs everything into a single file.

Network Intrusion Detection System mode

i. snort –d c:\log –h ipaddress/24 –c snort.conf→This is a configuration file applies rule to each packet to decide it an action based upon the rule type in the file.

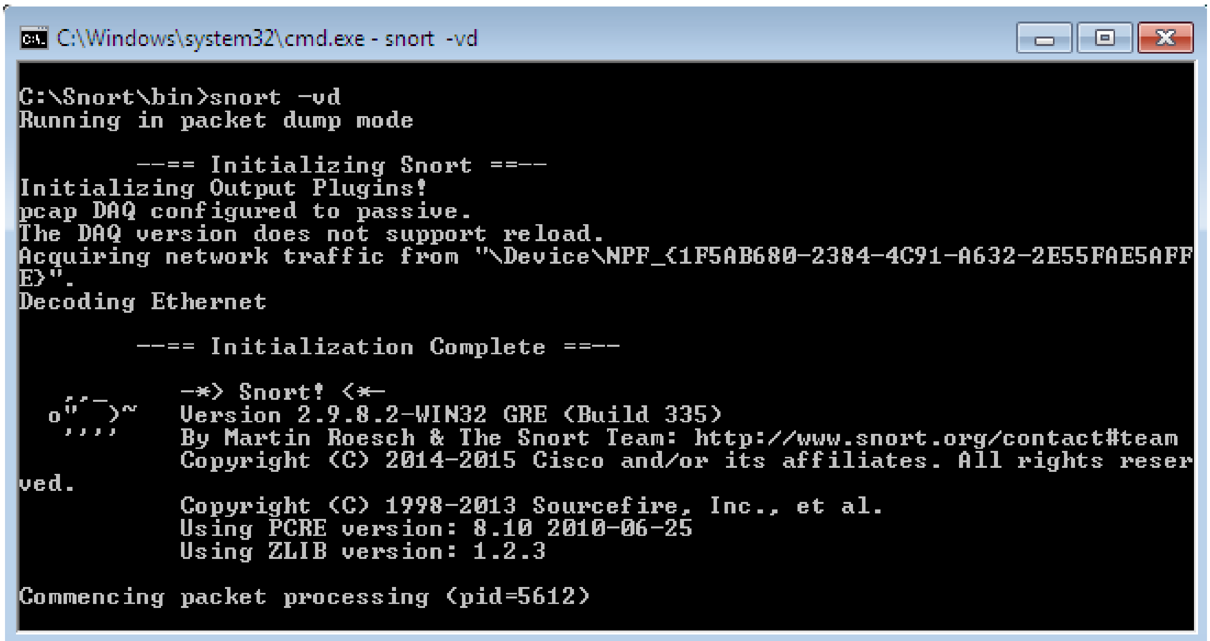
ii. snort –d –h ipaddress/24 –l c:\log –c snort.conf→This will configure snort to run in its most basic NIDS form, logging packets that trigger rules specifies in the snort.conf.

C:\Snort\bin\snort –v



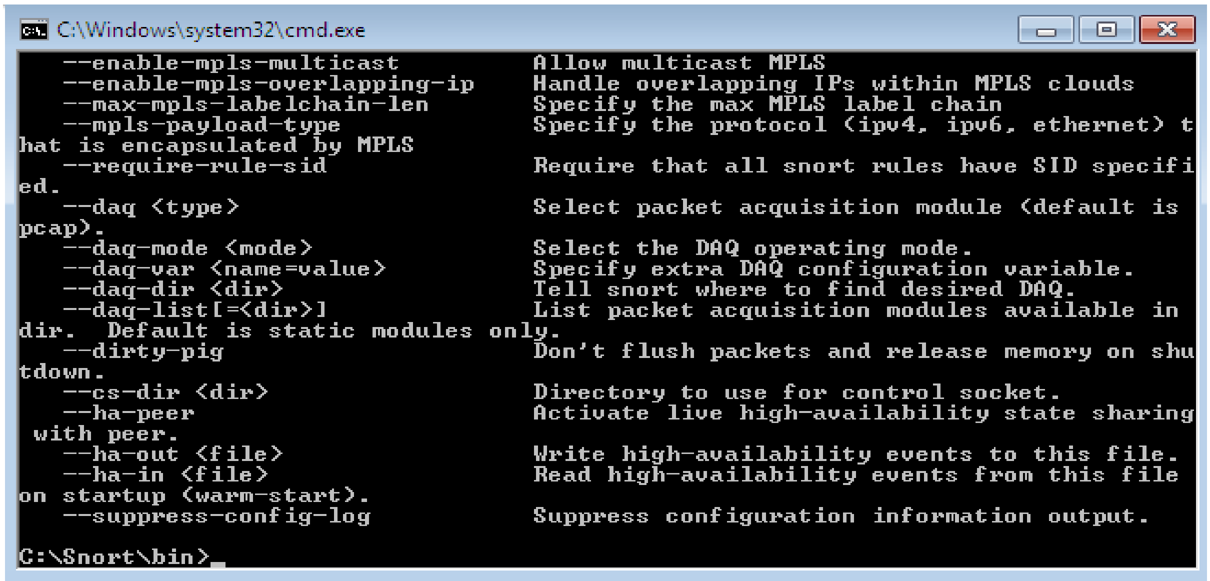
```
C:\Snort\bin>snort -v
Running in packet dump mode

        --== Initializing Snort ==--
Initializing Output Plugins!
pcap DAQ configured to passive.
The DAQ version does not support reload.
Acquiring network traffic from "\Device\NPF_{1F5AB680-2384-4C91-A632-2E55FAE5AFF
E}".
Decoding Ethernet

        --== Initialization Complete ==--

   ,,_     -*> Snort! <*-
  o"  )~   Version 2.9.8.2-WIN32 GRE (Build 335)
   ''''    By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
           Copyright (C) 2014-2015 Cisco and/or its affiliates. All rights reser
ved.
           Copyright (C) 1998-2013 Sourcefire, Inc., et al.
           Using PCRE version: 8.10 2010-06-25
           Using ZLIB version: 1.2.3

Commencing packet processing (pid=4680)
```

C:\Snort\bin\snort –vd



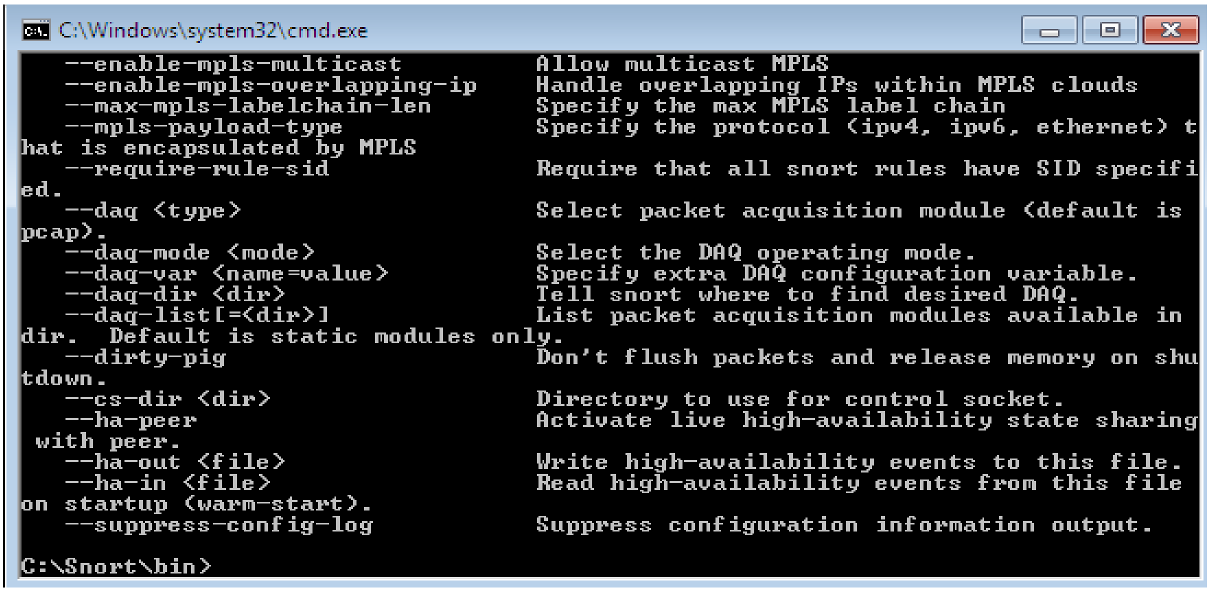
```
C:\Snort\bin>snort -vd
Running in packet dump mode

        --== Initializing Snort ==--
Initializing Output Plugins!
pcap DAQ configured to passive.
The DAQ version does not support reload.
Acquiring network traffic from "\Device\NPF_{1F5AB680-2384-4C91-A632-2E55FAE5AFF
E}".
Decoding Ethernet

        --== Initialization Complete ==--

   ,,_     -*> Snort! <*-
  o"  )~   Version 2.9.8.2-WIN32 GRE (Build 335)
   ''''    By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
           Copyright (C) 2014-2015 Cisco and/or its affiliates. All rights reser
ved.
           Copyright (C) 1998-2013 Sourcefire, Inc., et al.
           Using PCRE version: 8.10 2010-06-25
           Using ZLIB version: 1.2.3

Commencing packet processing (pid=5612)
```

C:\Snort\bin\ snort –dev –l c:\log



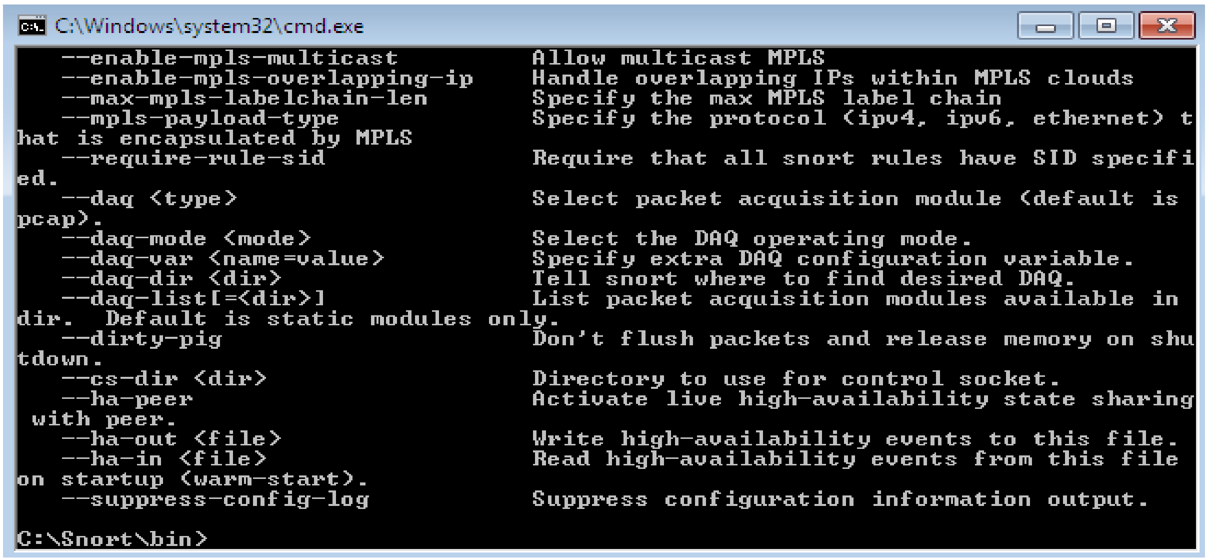
```
   --enable-mpls-multicast          Allow multicast MPLS
   --enable-mpls-overlapping-ip     Handle overlapping IPs within MPLS clouds
   --max-mpls-labelchain-len        Specify the max MPLS label chain
   --mpls-payload-type              Specify the protocol (ipv4, ipv6, ethernet) t
hat is encapsulated by MPLS
   --require-rule-sid               Require that all snort rules have SID specifi
ed.
   --daq <type>                     Select packet acquisition module (default is
pcap).
   --daq-mode <mode>                Select the DAQ operating mode.
   --daq-var <name=value>           Specify extra DAQ configuration variable.
   --daq-dir <dir>                  Tell snort where to find desired DAQ.
   --daq-list[=<dir>]               List packet acquisition modules available in
dir.  Default is static modules only.
   --dirty-pig                      Don't flush packets and release memory on shu
tdown.
   --cs-dir <dir>                   Directory to use for control socket.
   --ha-peer                        Activate live high-availability state sharing
 with peer.
   --ha-out <file>                  Write high-availability events to this file.
   --ha-in <file>                   Read high-availability events from this file
on startup (warm-start).
   --suppress-config-log            Suppress configuration information output.

C:\Snort\bin>_
```

C:\Snort\bin\snort –dev –l c:\log –h ipaddress/24



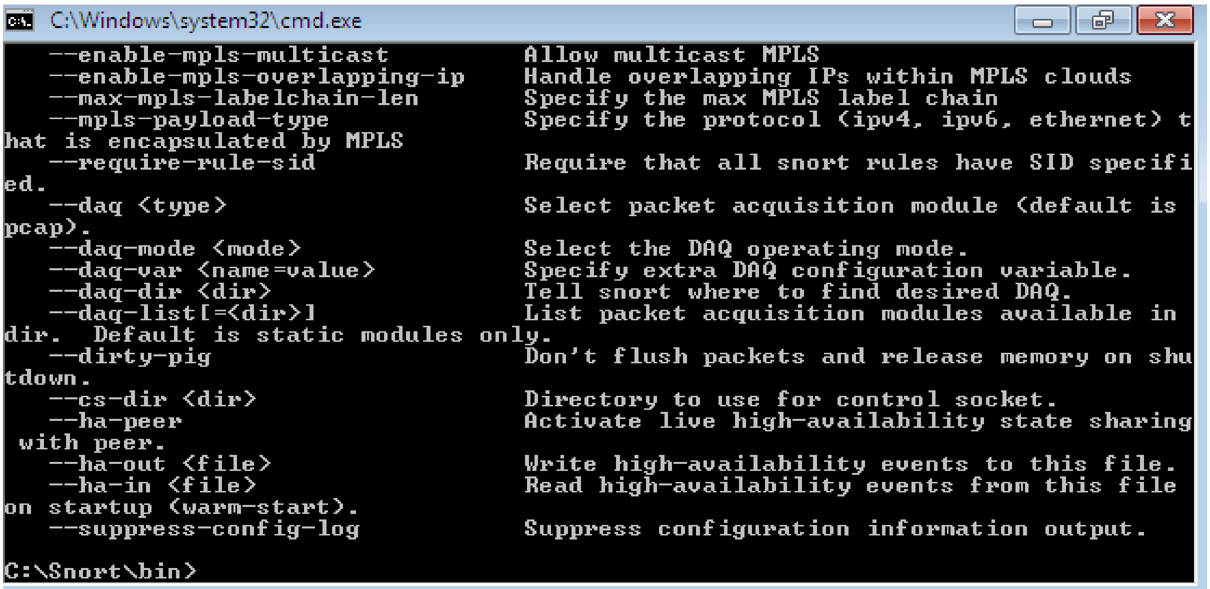
```
   --enable-mpls-multicast          Allow multicast MPLS
   --enable-mpls-overlapping-ip     Handle overlapping IPs within MPLS clouds
   --max-mpls-labelchain-len        Specify the max MPLS label chain
   --mpls-payload-type              Specify the protocol (ipv4, ipv6, ethernet) t
hat is encapsulated by MPLS
   --require-rule-sid               Require that all snort rules have SID specifi
ed.
   --daq <type>                     Select packet acquisition module (default is
pcap).
   --daq-mode <mode>                Select the DAQ operating mode.
   --daq-var <name=value>           Specify extra DAQ configuration variable.
   --daq-dir <dir>                  Tell snort where to find desired DAQ.
   --daq-list[=<dir>]               List packet acquisition modules available in
dir.  Default is static modules only.
   --dirty-pig                      Don't flush packets and release memory on shu
tdown.
   --cs-dir <dir>                   Directory to use for control socket.
   --ha-peer                        Activate live high-availability state sharing
 with peer.
   --ha-out <file>                  Write high-availability events to this file.
   --ha-in <file>                   Read high-availability events from this file
on startup (warm-start).
   --suppress-config-log            Suppress configuration information output.

C:\Snort\bin>
```

C:\Snort\bin\snort –l c:\log –b



```
   --enable-mpls-multicast         Allow multicast MPLS
   --enable-mpls-overlapping-ip    Handle overlapping IPs within MPLS clouds
   --max-mpls-labelchain-len       Specify the max MPLS label chain
   --mpls-payload-type             Specify the protocol (ipv4, ipv6, ethernet) t
hat is encapsulated by MPLS
   --require-rule-sid              Require that all snort rules have SID specifi
ed.
   --daq <type>                    Select packet acquisition module (default is
pcap).
   --daq-mode <mode>               Select the DAQ operating mode.
   --daq-var <name=value>          Specify extra DAQ configuration variable.
   --daq-dir <dir>                 Tell snort where to find desired DAQ.
   --daq-list[=<dir>]              List packet acquisition modules available in
dir.  Default is static modules only.
   --dirty-pig                     Don't flush packets and release memory on shu
tdown.
   --cs-dir <dir>                  Directory to use for control socket.
   --ha-peer                       Activate live high-availability state sharing
 with peer.
   --ha-out <file>                 Write high-availability events to this file.
   --ha-in <file>                  Read high-availability events from this file
on startup (warm-start).
   --suppress-config-log           Suppress configuration information output.

C:\Snort\bin>
```

snort –d –h ipaddress/24 –l c:\log –c snort.conf



```
   --enable-mpls-multicast         Allow multicast MPLS
   --enable-mpls-overlapping-ip    Handle overlapping IPs within MPLS clouds
   --max-mpls-labelchain-len       Specify the max MPLS label chain
   --mpls-payload-type             Specify the protocol (ipv4, ipv6, ethernet) t
hat is encapsulated by MPLS
   --require-rule-sid              Require that all snort rules have SID specifi
ed.
   --daq <type>                    Select packet acquisition module (default is
pcap).
   --daq-mode <mode>               Select the DAQ operating mode.
   --daq-var <name=value>          Specify extra DAQ configuration variable.
   --daq-dir <dir>                 Tell snort where to find desired DAQ.
   --daq-list[=<dir>]              List packet acquisition modules available in
dir.  Default is static modules only.
   --dirty-pig                     Don't flush packets and release memory on shu
tdown.
   --cs-dir <dir>                  Directory to use for control socket.
   --ha-peer                       Activate live high-availability state sharing
 with peer.
   --ha-out <file>                 Write high-availability events to this file.
   --ha-in <file>                  Read high-availability events from this file
on startup (warm-start).
   --suppress-config-log           Suppress configuration information output.

C:\Snort\bin>
```

RESULT:

Thus the documentation of the instruction detection using Snort tool was done successfully.

DATE:	**ATTACK AND PENETRATION TOOL: PACKET ANALYSIS USING WIRESHARK**
EXPT NO: 12	

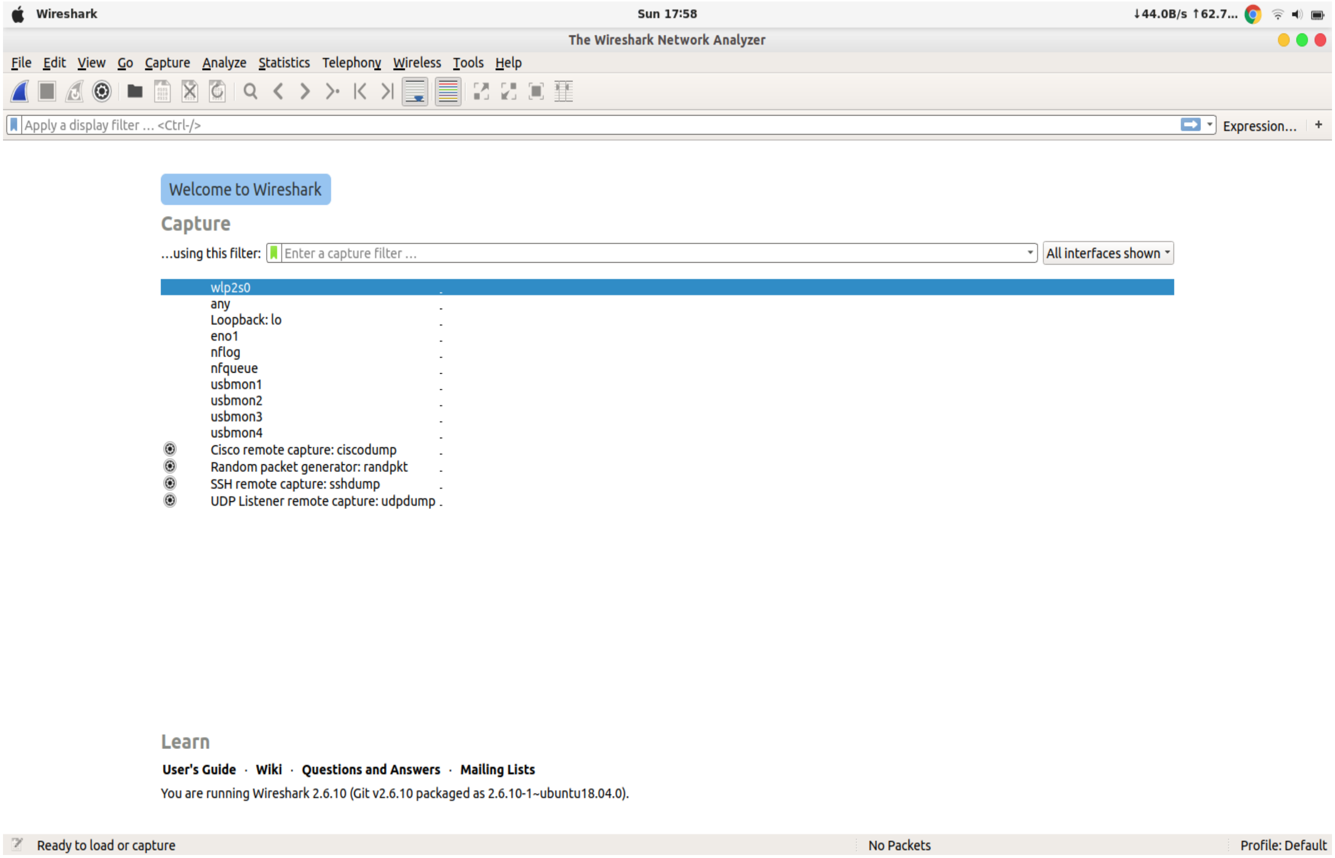
AIM:

To demonstrate the Packet Analysis using Wireshark in LAN Network.

PROCEDURE:

Step 1: Start the Wireshark Network Analyzer.
Step 2: Check whether the System is connected in LAN Network.
Step 3: Select One or More Networks by Clicking On them.
Step 4: Select Capture at the top of the Wireshark Interface.
Step 5: Select Start.
Step 6: Select File > Save as Or Choose One of the Export Options to Record the Capture.
Step 7: To Stop capture the Packets, Press Ctrl + E or Select Red Stop Button next to the Sharkfin on the Wireshark Toolbar.

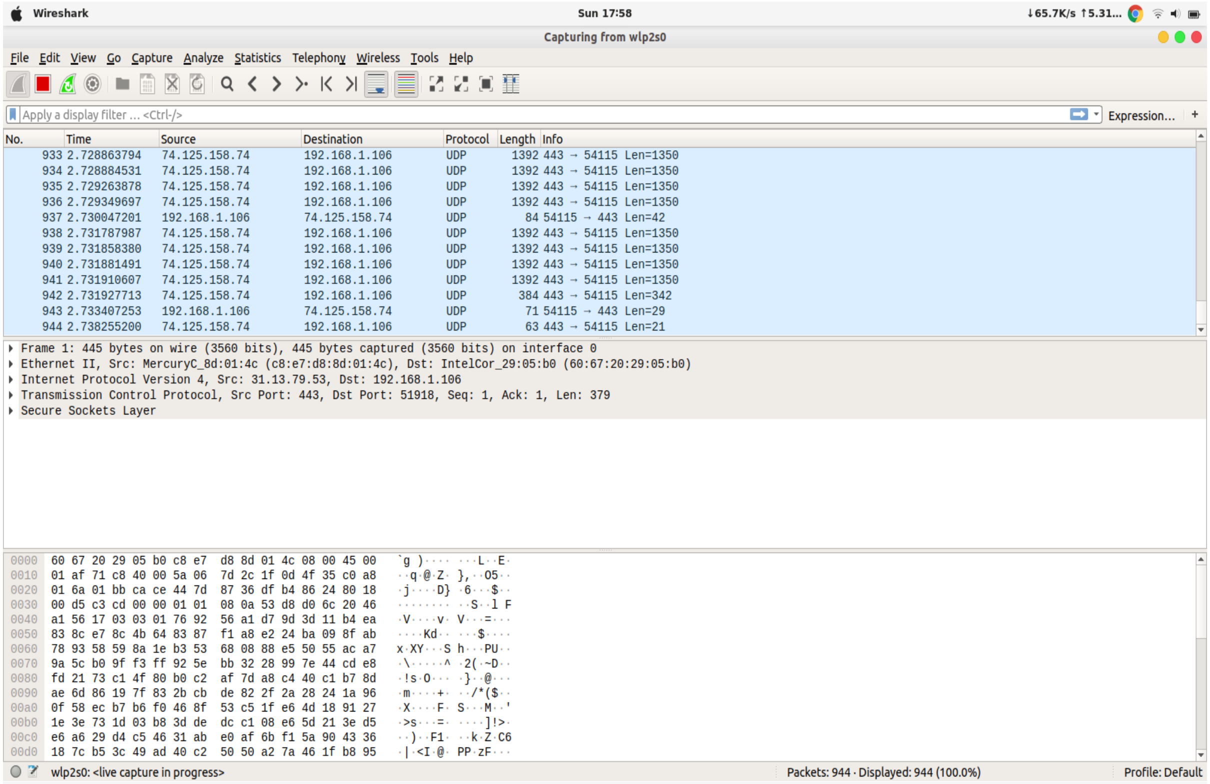
EXECUTION:


Wireshark
Sun 17:58
↓44.0B/s ↑62.7...
The Wireshark Network Analyzer
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...
+
Welcome to Wireshark
Capture
...using this filter:
Enter a capture filter ...
All interfaces shown
Learn
User's Guide · Wiki · Questions and Answers · Mailing Lists

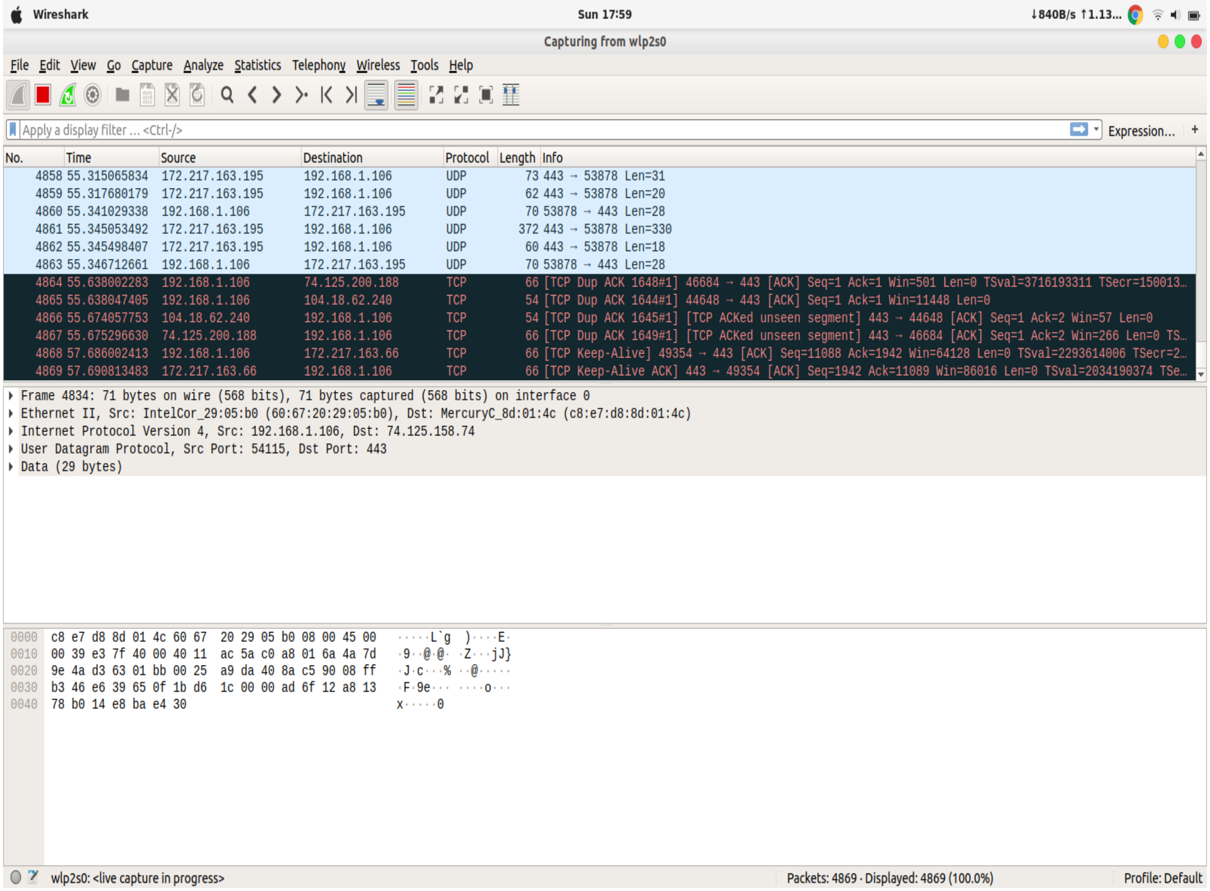
wlp2s0
any
Loopback: lo
eno1
nflog
nfqueue
usbmon1
usbmon2
usbmon3
usbmon4
Cisco remote capture: ciscodump
Random packet generator: randpkt
SSH remote capture: sshdump
UDP Listener remote capture: udpdump

You are running Wireshark 2.6.10 (Git v2.6.10 packaged as 2.6.10-1~ubuntu18.04.0).

Ready to load or capture | No Packets | Profile: Default



```
Wireshark                                   Sun 17:58                    ↓65.7K/s ↑5.31...
                                   Capturing from wlp2s0
File  Edit  View  Go  Capture  Analyze  Statistics  Telephony  Wireless  Tools  Help
Apply a display filter ... <Ctrl-/>                                          Expression...  +
No.    Time         Source          Destination     Protocol Length Info
   933 2.728863794  74.125.158.74   192.168.1.106   UDP      1392 443 → 54115 Len=1350
   934 2.728884531  74.125.158.74   192.168.1.106   UDP      1392 443 → 54115 Len=1350
   935 2.729263878  74.125.158.74   192.168.1.106   UDP      1392 443 → 54115 Len=1350
   936 2.729349697  74.125.158.74   192.168.1.106   UDP      1392 443 → 54115 Len=1350
   937 2.730047201  192.168.1.106   74.125.158.74   UDP        84 54115 → 443 Len=42
   938 2.731787987  74.125.158.74   192.168.1.106   UDP      1392 443 → 54115 Len=1350
   939 2.731858380  74.125.158.74   192.168.1.106   UDP      1392 443 → 54115 Len=1350
   940 2.731881491  74.125.158.74   192.168.1.106   UDP      1392 443 → 54115 Len=1350
   941 2.731910607  74.125.158.74   192.168.1.106   UDP      1392 443 → 54115 Len=1350
   942 2.731927713  74.125.158.74   192.168.1.106   UDP       384 443 → 54115 Len=342
   943 2.733407253  192.168.1.106   74.125.158.74   UDP        71 54115 → 443 Len=29
   944 2.738255200  74.125.158.74   192.168.1.106   UDP        63 443 → 54115 Len=21

▸ Frame 1: 445 bytes on wire (3560 bits), 445 bytes captured (3560 bits) on interface 0
▸ Ethernet II, Src: MercuryC_8d:01:4c (c8:e7:d8:8d:01:4c), Dst: IntelCor_29:05:b0 (60:67:20:29:05:b0)
▸ Internet Protocol Version 4, Src: 31.13.79.53, Dst: 192.168.1.106
▸ Transmission Control Protocol, Src Port: 443, Dst Port: 51918, Seq: 1, Ack: 1, Len: 379
▸ Secure Sockets Layer

0000  60 67 20 29 05 b0 c8 e7  d8 8d 01 4c 08 00 45 00   `g )···· ···L··E·
0010  01 af 71 c8 40 00 5a 06  7d 2c 1f 0d 4f 35 c0 a8   ··q·@·Z· },··O5··
0020  01 6a 01 bb ca ce 44 7d  87 36 df b4 86 24 80 18   ·j····D} ·6···$··
0030  00 d5 c3 cd 00 00 01 01  08 0a 53 d8 d0 6c 20 46   ········ ··S··l F
0040  a1 56 17 03 03 01 76 92  56 a1 d7 9d 3d 11 b4 ea   ·V····v· V···=···
0050  83 8c e7 8c 4b 64 83 87  f1 a8 e2 24 ba 09 8f ab   ····Kd·· ···$····
0060  78 93 58 59 8a 1e b3 53  68 08 88 e5 50 55 ac a7   x·XY···S h···PU··
0070  9a 5c b0 9f f3 ff 92 5e  bb 32 28 99 7e 44 cd e8   ·\·····^ ·2(·~D··
0080  fd 21 73 c1 4f 80 b0 c2  af 7d a8 c4 40 c1 b7 8d   ·!s·O··· ·}··@···
0090  ae 6d 86 19 7f 83 2b cb  de 82 2f 2a 28 24 1a 96   ·m····+· ··/*($··
00a0  0f 58 ec b7 b6 f0 46 8f  53 c5 1f e6 4d 18 91 27   ·X····F· S···M··'
00b0  1e 3e 73 1d 03 b8 3d de  dc c1 08 e6 5d 21 3e d5   ·>s···=· ····]!>·
00c0  e6 a6 29 d4 c5 46 31 ab  e0 af 6b f1 5a 90 43 36   ··)··F1· ··k·Z·C6
00d0  18 7c b5 3c 49 ad 40 c2  50 50 a2 7a 46 1f b8 95   ·|·<I·@· PP·zF···

wlp2s0: <live capture in progress>     Packets: 944 · Displayed: 944 (100.0%)     Profile: Default
```



```
Wireshark                                   Sun 17:59                    ↓840B/s ↑1.13...
                                   Capturing from wlp2s0
File  Edit  View  Go  Capture  Analyze  Statistics  Telephony  Wireless  Tools  Help
Apply a display filter ... <Ctrl-/>                                          Expression...  +
No.    Time          Source           Destination      Protocol Length Info
  4858 55.315065834  172.217.163.195  192.168.1.106    UDP        73 443 → 53878 Len=31
  4859 55.317680179  172.217.163.195  192.168.1.106    UDP        62 443 → 53878 Len=20
  4860 55.341029338  192.168.1.106    172.217.163.195  UDP        70 53878 → 443 Len=28
  4861 55.345053492  172.217.163.195  192.168.1.106    UDP       372 443 → 53878 Len=330
  4862 55.345498407  172.217.163.195  192.168.1.106    UDP        60 443 → 53878 Len=18
  4863 55.346712661  192.168.1.106    172.217.163.195  UDP        70 53878 → 443 Len=28
  4864 55.638002283  192.168.1.106    74.125.200.188   TCP        66 [TCP Dup ACK 1648#1] 46684 → 443 [ACK] Seq=1 Ack=1 Win=501 Len=0 TSval=3716193311 TSecr=150013…
  4865 55.638047405  192.168.1.106    104.18.62.240    TCP        54 [TCP Dup ACK 1644#1] 44648 → 443 [ACK] Seq=1 Ack=1 Win=11448 Len=0
  4866 55.674057753  104.18.62.240    192.168.1.106    TCP        54 [TCP Dup ACK 1645#1] [TCP ACKed unseen segment] 443 → 44648 [ACK] Seq=1 Ack=2 Win=57 Len=0
  4867 55.675296630  74.125.200.188   192.168.1.106    TCP        66 [TCP Dup ACK 1649#1] [TCP ACKed unseen segment] 443 → 46684 [ACK] Seq=1 Ack=2 Win=266 Len=0 TS…
  4868 57.686002413  192.168.1.106    172.217.163.66   TCP        66 [TCP Keep-Alive] 49354 → 443 [ACK] Seq=11088 Ack=1942 Win=64128 Len=0 TSval=2293614006 TSecr=2…
  4869 57.690813483  172.217.163.66   192.168.1.106    TCP        66 [TCP Keep-Alive ACK] 443 → 49354 [ACK] Seq=1942 Ack=11089 Win=86016 Len=0 TSval=2034190374 TSe…

▸ Frame 4834: 71 bytes on wire (568 bits), 71 bytes captured (568 bits) on interface 0
▸ Ethernet II, Src: IntelCor_29:05:b0 (60:67:20:29:05:b0), Dst: MercuryC_8d:01:4c (c8:e7:d8:8d:01:4c)
▸ Internet Protocol Version 4, Src: 192.168.1.106, Dst: 74.125.158.74
▸ User Datagram Protocol, Src Port: 54115, Dst Port: 443
▸ Data (29 bytes)

0000  c8 e7 d8 8d 01 4c 60 67  20 29 05 b0 08 00 45 00   ·····L`g  )····E·
0010  00 39 e3 7f 40 00 40 11  ac 5a c0 a8 01 6a 4a 7d   ·9··@·@· ·Z···jJ}
0020  9e 4a d3 63 01 bb 00 25  a9 da 40 8a c5 90 08 ff   ·J·c···% ··@·····
0030  b3 46 e6 39 65 0f 1b d6  1c 00 00 ad 6f 12 a8 13   ·F·9e··· ····o···
0040  78 b0 14 e8 ba e4 30                               x·····0

wlp2s0: <live capture in progress>     Packets: 4869 · Displayed: 4869 (100.0%)     Profile: Default
```

RESULT:

Thus the demonstration for the Packet Analysing using Wireshark in a LAN Network is implemented successfully.

DATE:	**INSTALLATION OF RK HUNTER AND STUDY ABOUT THE VARIETY OF OPTION**
EXPT NO: 13	

AIM

To install rkhunter and study about the variety of options.

PROCEDURE:

Step1:RKhunter is a stealth type of malicious software designed to hide the existence of certain process from normal methods of detection and enables continued privileged access to a computer.
Step2:Download Rkhunter Tool from GMER website. (www.gmer.net)

Step3:This displays the Processes, Modules, Services, Files, Registry, Rkhunter/Malwares, Autostart, CMD of local host.

Step4:Select Processes menu and kill any unwanted process if any.

Step5:Modules menu displays the various system files like .sys, .dll

Step6:Services menu displays the complete services running with Autostart, Enable, Disable, System, Boot.

Step7:Files menu displays full files on Hard-Disk volumes.

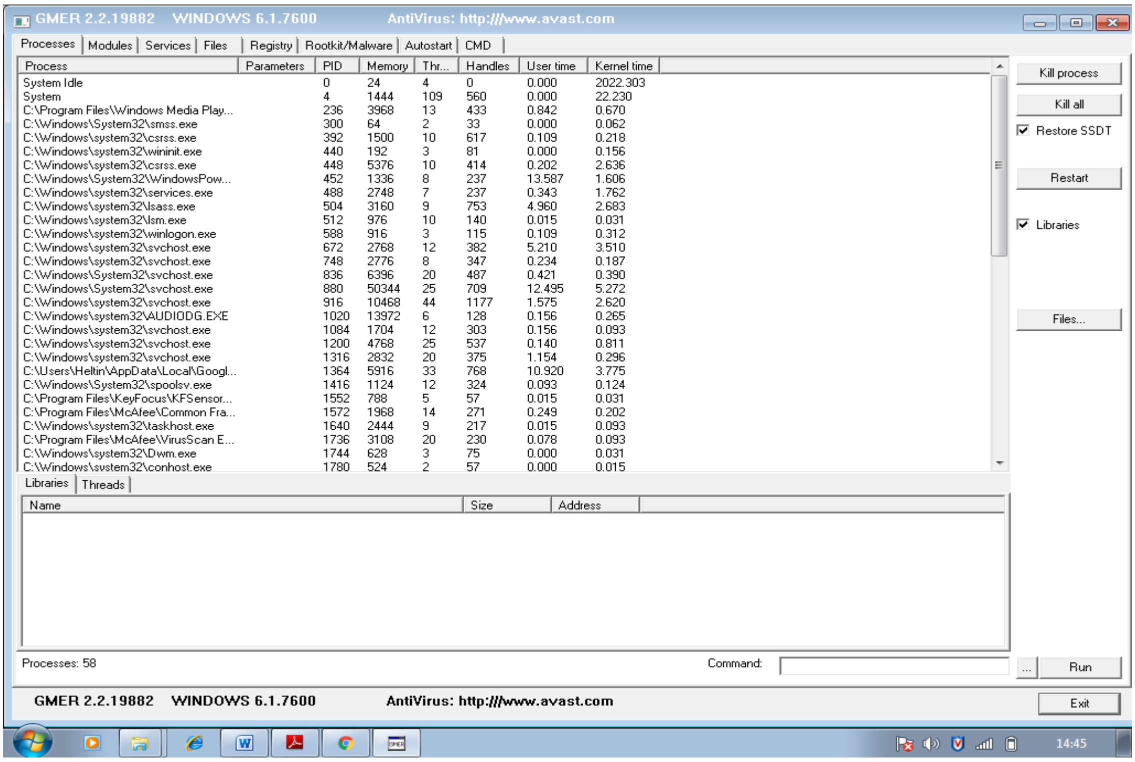
Step8:Registry displays Hkey_Current_user and Hkey_Local_Machine.

Step9:Rkhunter/Malawares scans the local drives selected.

Step10:Autostart displays the registry base Autostart applications.

Step11:CMD allows the user to interact with command line utilities or Registry.

EXECUTION:

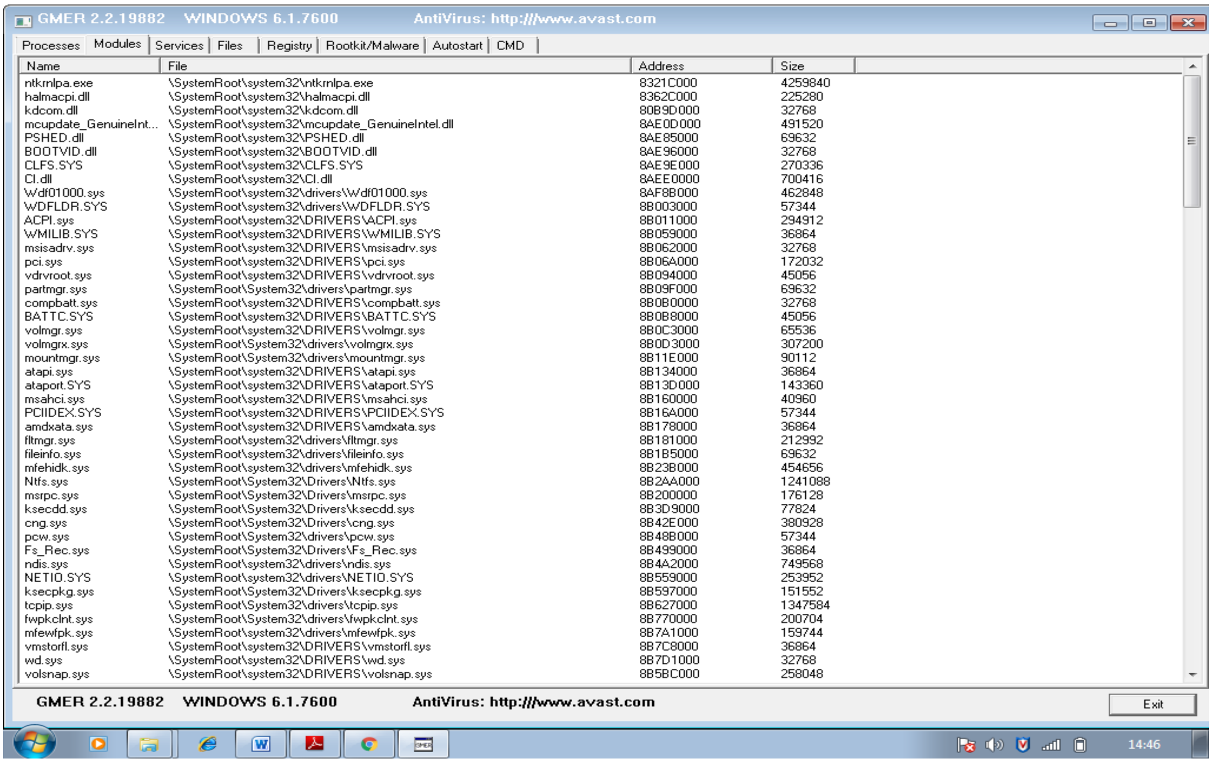


Process	Parameters	PID	Memory	Thr...	Handles	User time	Kernel time
System Idle		0	24	4	0	0.000	2022.303
System		4	1444	109	560	0.000	22.230
C:\Program Files\Windows Media Play...		236	3968	13	433	0.842	0.670
C:\Windows\System32\smss.exe		300	64	2	33	0.000	0.062
C:\Windows\system32\csrss.exe		392	1500	10	617	0.109	0.218
C:\Windows\system32\wininit.exe		440	192	3	81	0.000	0.156
C:\Windows\system32\csrss.exe		448	5376	10	414	0.202	2.636
C:\Windows\System32\WindowsPow...		452	1336	8	237	13.587	1.606
C:\Windows\system32\services.exe		488	2748	7	237	0.343	1.762
C:\Windows\system32\lsass.exe		504	3160	9	753	4.960	2.683
C:\Windows\system32\lsm.exe		512	976	10	140	0.015	0.031
C:\Windows\system32\winlogon.exe		588	916	3	115	0.109	0.312
C:\Windows\system32\svchost.exe		672	2768	12	382	5.210	3.510
C:\Windows\system32\svchost.exe		748	2776	8	347	0.234	0.187
C:\Windows\System32\svchost.exe		836	6396	20	487	0.421	0.390
C:\Windows\System32\svchost.exe		880	50344	25	709	12.495	5.272
C:\Windows\system32\svchost.exe		916	10468	44	1177	1.575	2.620
C:\Windows\system32\AUDIODG.EXE		1020	13972	6	128	0.156	0.265
C:\Windows\system32\svchost.exe		1084	1704	12	303	0.156	0.093
C:\Windows\system32\svchost.exe		1200	4768	25	537	0.140	0.811
C:\Windows\system32\svchost.exe		1316	2832	20	375	1.154	0.296
C:\Users\Heltin\AppData\Local\Googl...		1364	5916	33	768	10.920	3.775
C:\Windows\System32\spoolsv.exe		1416	1124	12	324	0.093	0.124
C:\Program Files\KeyFocus\KFSensor...		1552	788	5	57	0.015	0.031
C:\Program Files\McAfee\Common Fra...		1572	1968	14	271	0.249	0.202
C:\Windows\system32\taskhost.exe		1640	2444	9	217	0.015	0.093
C:\Program Files\McAfee\VirusScan E...		1736	3108	20	230	0.078	0.093
C:\Windows\system32\Dwm.exe		1744	628	3	75	0.000	0.031
C:\Windows\system32\conhost.exe		1780	524	2	57	0.000	0.015

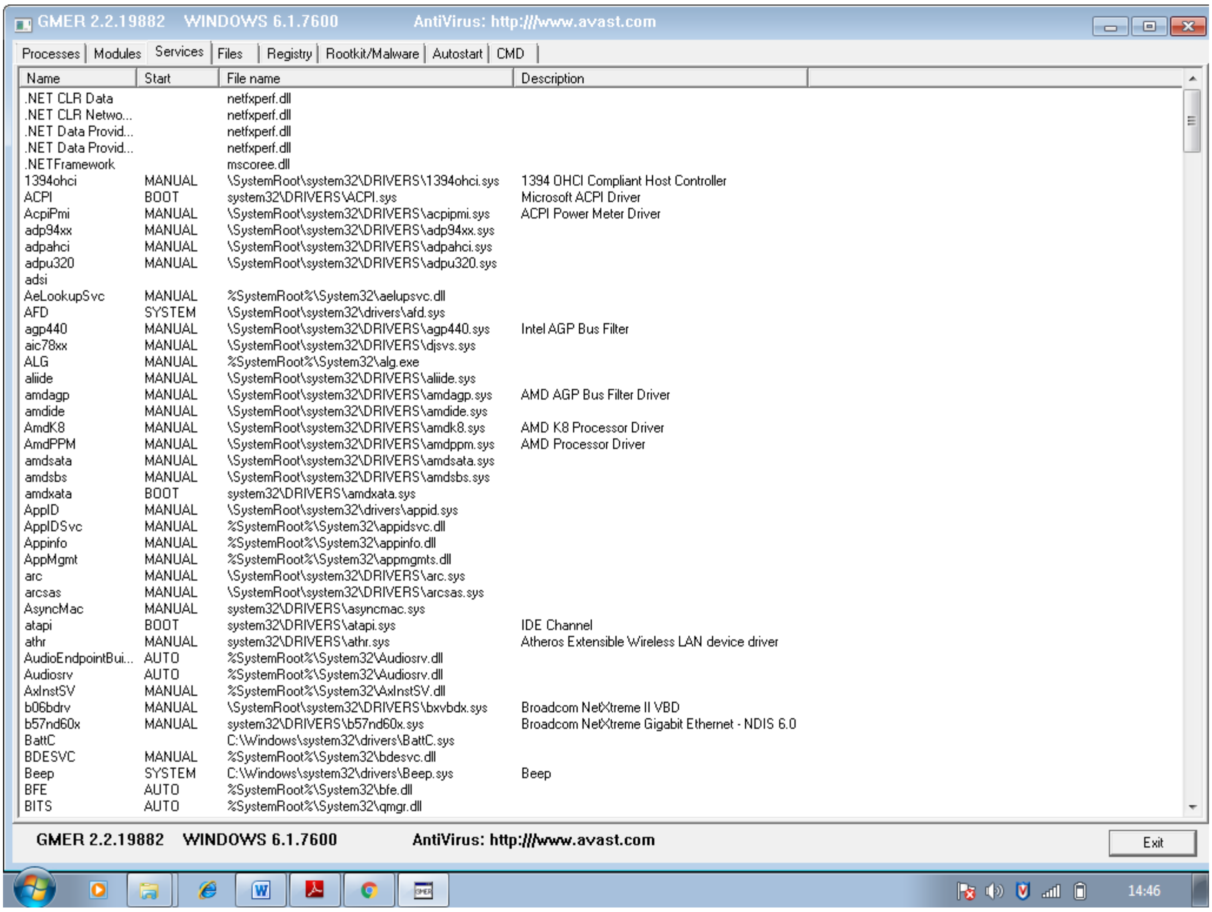
Processes: 58

Command:

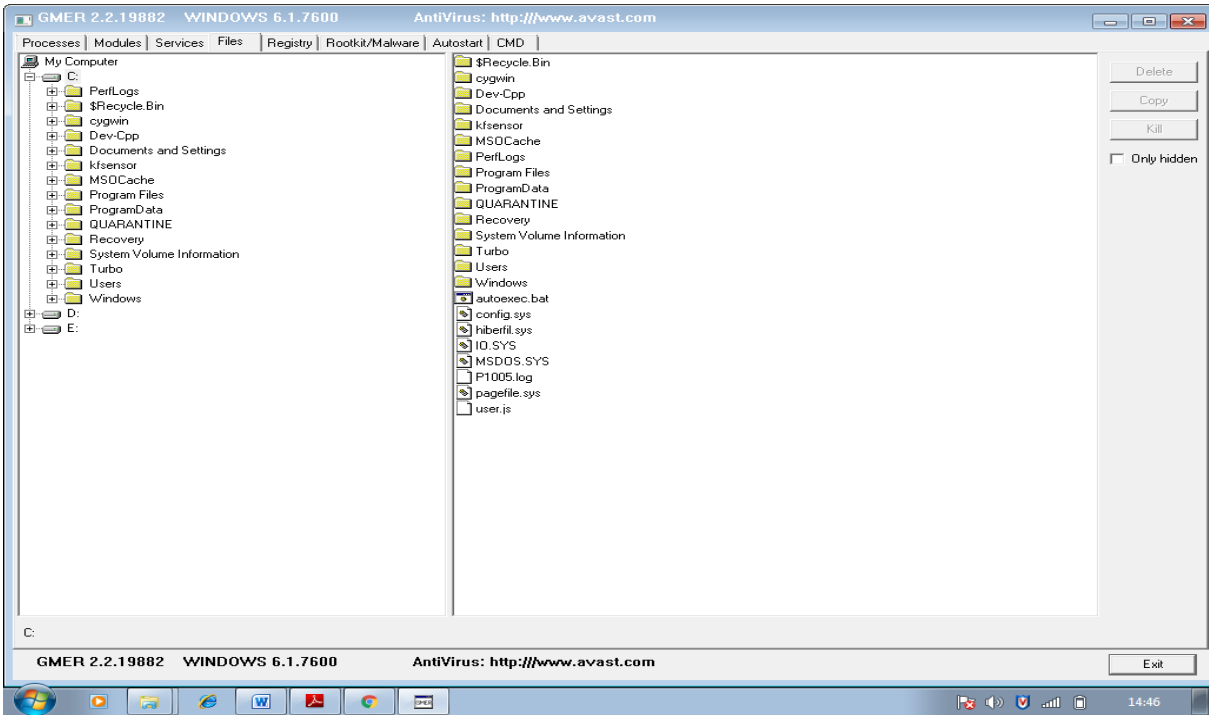
GMER 2.2.19882 WINDOWS 6.1.7600 AntiVirus: http:///www.avast.com



Name	File	Address	Size
ntkrnlpa.exe	\SystemRoot\system32\ntkrnlpa.exe	8321C000	4259840
halmacpi.dll	\SystemRoot\system32\halmacpi.dll	8362C000	225280
kdcom.dll	\SystemRoot\system32\kdcom.dll	80B9D000	32768
mcupdate_GenuineInt...	\SystemRoot\system32\mcupdate_GenuineIntel.dll	8AE0D000	491520
PSHED.dll	\SystemRoot\system32\PSHED.dll	8AE85000	69632
BOOTVID.dll	\SystemRoot\system32\BOOTVID.dll	8AE96000	32768
CLFS.SYS	\SystemRoot\system32\CLFS.SYS	8AE9E000	270336
CI.dll	\SystemRoot\system32\CI.dll	8AEE0000	700416
Wdf01000.sys	\SystemRoot\system32\drivers\Wdf01000.sys	8AF8B000	462848
WDFLDR.SYS	\SystemRoot\system32\drivers\WDFLDR.SYS	8B003000	57344
ACPI.sys	\SystemRoot\system32\DRIVERS\ACPI.sys	8B011000	294912
WMILIB.SYS	\SystemRoot\system32\DRIVERS\WMILIB.SYS	8B059000	36864
msisadrv.sys	\SystemRoot\system32\DRIVERS\msisadrv.sys	8B062000	32768
pci.sys	\SystemRoot\system32\DRIVERS\pci.sys	8B06A000	172032
vdrvroot.sys	\SystemRoot\system32\DRIVERS\vdrvroot.sys	8B094000	45056
partmgr.sys	\SystemRoot\System32\drivers\partmgr.sys	8B09F000	69632
compbatt.sys	\SystemRoot\system32\DRIVERS\compbatt.sys	8B0B0000	32768
BATTC.SYS	\SystemRoot\system32\DRIVERS\BATTC.SYS	8B0B8000	45056
volmgr.sys	\SystemRoot\system32\DRIVERS\volmgr.sys	8B0C3000	65536
volmgrx.sys	\SystemRoot\System32\drivers\volmgrx.sys	8B0D3000	307200
mountmgr.sys	\SystemRoot\System32\drivers\mountmgr.sys	8B11E000	90112
atapi.sys	\SystemRoot\system32\DRIVERS\atapi.sys	8B134000	36864
ataport.SYS	\SystemRoot\system32\DRIVERS\ataport.SYS	8B13D000	143360
msahci.sys	\SystemRoot\system32\DRIVERS\msahci.sys	8B160000	40960
PCIIDEX.SYS	\SystemRoot\system32\DRIVERS\PCIIDEX.SYS	8B16A000	57344
amdxata.sys	\SystemRoot\system32\DRIVERS\amdxata.sys	8B178000	36864
fltmgr.sys	\SystemRoot\system32\drivers\fltmgr.sys	8B181000	212992
fileinfo.sys	\SystemRoot\system32\drivers\fileinfo.sys	8B1B5000	69632
mfehidk.sys	\SystemRoot\system32\drivers\mfehidk.sys	8B23B000	454656
Ntfs.sys	\SystemRoot\System32\Drivers\Ntfs.sys	8B2AA000	1241088
msrpc.sys	\SystemRoot\System32\Drivers\msrpc.sys	8B200000	176128
ksecdd.sys	\SystemRoot\System32\Drivers\ksecdd.sys	8B3D9000	77824
cng.sys	\SystemRoot\System32\Drivers\cng.sys	8B42E000	380928
pcw.sys	\SystemRoot\System32\drivers\pcw.sys	8B48B000	57344
Fs_Rec.sys	\SystemRoot\System32\Drivers\Fs_Rec.sys	8B499000	36864
ndis.sys	\SystemRoot\system32\drivers\ndis.sys	8B4A2000	749568
NETIO.SYS	\SystemRoot\system32\drivers\NETIO.SYS	8B559000	253952
ksecpkg.sys	\SystemRoot\System32\Drivers\ksecpkg.sys	8B597000	151552
tcpip.sys	\SystemRoot\System32\drivers\tcpip.sys	8B627000	1347584
fwpkclnt.sys	\SystemRoot\System32\drivers\fwpkclnt.sys	8B770000	200704
mfewfpk.sys	\SystemRoot\system32\drivers\mfewfpk.sys	8B7A1000	159744
vmstorfl.sys	\SystemRoot\system32\DRIVERS\vmstorfl.sys	8B7C8000	36864
wd.sys	\SystemRoot\system32\DRIVERS\wd.sys	8B7D1000	32768
volsnap.sys	\SystemRoot\system32\DRIVERS\volsnap.sys	8B5BC000	258048



Name	Start	File name	Description
.NET CLR Data		netfxperf.dll	
.NET CLR Netwo...		netfxperf.dll	
.NET Data Provid...		netfxperf.dll	
.NET Data Provid...		netfxperf.dll	
.NETFramework		mscoree.dll	
1394ohci	MANUAL	\SystemRoot\system32\DRIVERS\1394ohci.sys	1394 OHCI Compliant Host Controller
ACPI	BOOT	system32\DRIVERS\ACPI.sys	Microsoft ACPI Driver
AcpiPmi	MANUAL	\SystemRoot\system32\DRIVERS\acpipmi.sys	ACPI Power Meter Driver
adp94xx	MANUAL	\SystemRoot\system32\DRIVERS\adp94xx.sys	
adpahci	MANUAL	\SystemRoot\system32\DRIVERS\adpahci.sys	
adpu320	MANUAL	\SystemRoot\system32\DRIVERS\adpu320.sys	
adsi			
AeLookupSvc	MANUAL	%SystemRoot%\System32\aelupsvc.dll	
AFD	SYSTEM	\SystemRoot\system32\drivers\afd.sys	
agp440	MANUAL	\SystemRoot\system32\DRIVERS\agp440.sys	Intel AGP Bus Filter
aic78xx	MANUAL	\SystemRoot\system32\DRIVERS\djsvs.sys	
ALG	MANUAL	%SystemRoot%\System32\alg.exe	
aliide	MANUAL	\SystemRoot\system32\DRIVERS\aliide.sys	
amdagp	MANUAL	\SystemRoot\system32\DRIVERS\amdagp.sys	AMD AGP Bus Filter Driver
amdide	MANUAL	\SystemRoot\system32\DRIVERS\amdide.sys	
AmdK8	MANUAL	\SystemRoot\system32\DRIVERS\amdk8.sys	AMD K8 Processor Driver
AmdPPM	MANUAL	\SystemRoot\system32\DRIVERS\amdppm.sys	AMD Processor Driver
amdsata	MANUAL	\SystemRoot\system32\DRIVERS\amdsata.sys	
amdsbs	MANUAL	\SystemRoot\system32\DRIVERS\amdsbs.sys	
amdxata	BOOT	system32\DRIVERS\amdxata.sys	
AppID	MANUAL	\SystemRoot\system32\drivers\appid.sys	
AppIDSvc	MANUAL	%SystemRoot%\System32\appidsvc.dll	
Appinfo	MANUAL	%SystemRoot%\System32\appinfo.dll	
AppMgmt	MANUAL	%SystemRoot%\System32\appmgmts.dll	
arc	MANUAL	\SystemRoot\system32\DRIVERS\arc.sys	
arcsas	MANUAL	\SystemRoot\system32\DRIVERS\arcsas.sys	
AsyncMac	MANUAL	system32\DRIVERS\asyncmac.sys	
atapi	BOOT	system32\DRIVERS\atapi.sys	IDE Channel
athr	MANUAL	system32\DRIVERS\athr.sys	Atheros Extensible Wireless LAN device driver
AudioEndpointBui...	AUTO	%SystemRoot%\System32\Audiosrv.dll	
Audiosrv	AUTO	%SystemRoot%\System32\Audiosrv.dll	
AxInstSV	MANUAL	%SystemRoot%\System32\AxInstSV.dll	
b06bdrv	MANUAL	\SystemRoot\system32\DRIVERS\bxvbdx.sys	Broadcom NetXtreme II VBD
b57nd60x	MANUAL	system32\DRIVERS\b57nd60x.sys	Broadcom NetXtreme Gigabit Ethernet - NDIS 6.0
BattC		C:\Windows\system32\drivers\BattC.sys	
BDESVC	MANUAL	%SystemRoot%\System32\bdesvc.dll	
Beep	SYSTEM	C:\Windows\system32\drivers\Beep.sys	Beep
BFE	AUTO	%SystemRoot%\System32\bfe.dll	
BITS	AUTO	%SystemRoot%\System32\qmgr.dll	



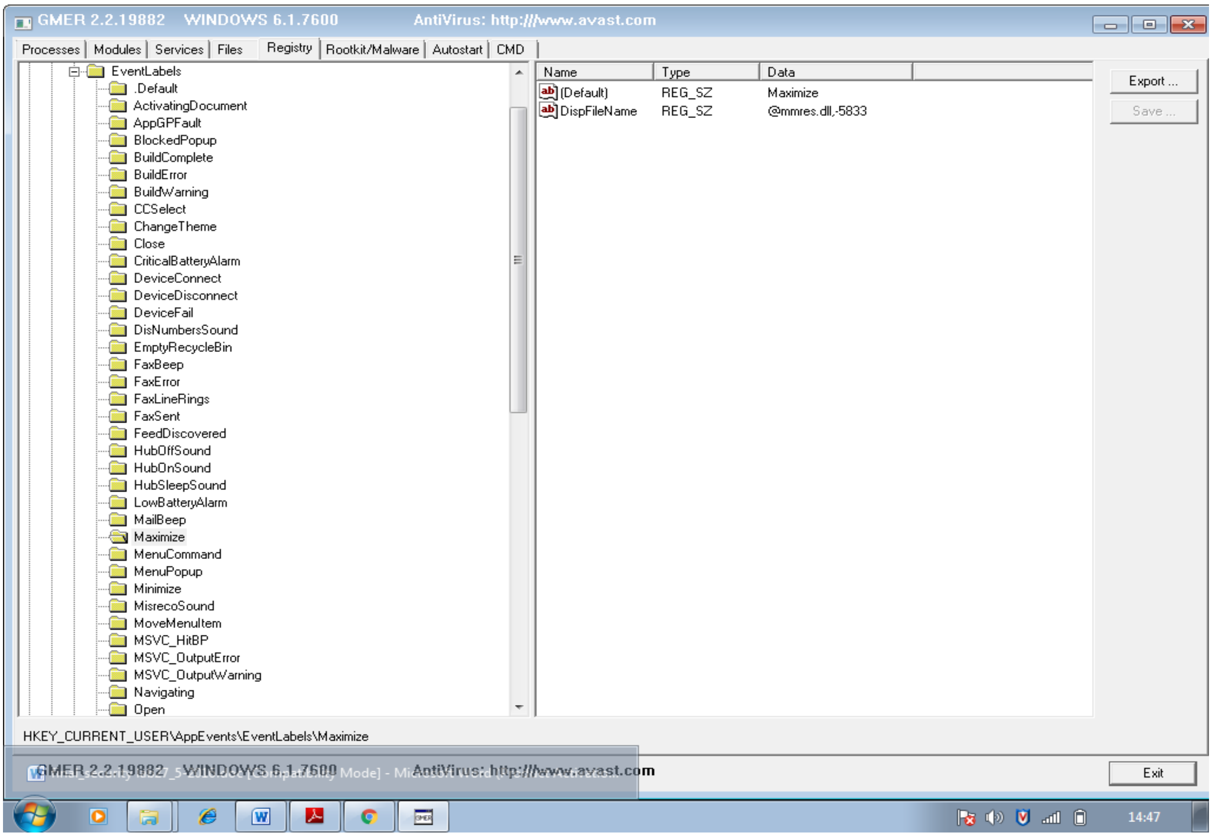
```
My Computer
  C:
    PerfLogs
    $Recycle.Bin
    cygwin
    Dev-Cpp
    Documents and Settings
    kfsensor
    MSOCache
    Program Files
    ProgramData
    QUARANTINE
    Recovery
    System Volume Information
    Turbo
    Users
    Windows
  D:
  E:
```

```
$Recycle.Bin
cygwin
Dev-Cpp
Documents and Settings
kfsensor
MSOCache
PerfLogs
Program Files
ProgramData
QUARANTINE
Recovery
System Volume Information
Turbo
Users
Windows
autoexec.bat
config.sys
hiberfil.sys
IO.SYS
MSDOS.SYS
P1005.log
pagefile.sys
user.js
```

□ Only hidden

C:

GMER 2.2.19882 WINDOWS 6.1.7600 AntiVirus: http://www.avast.com

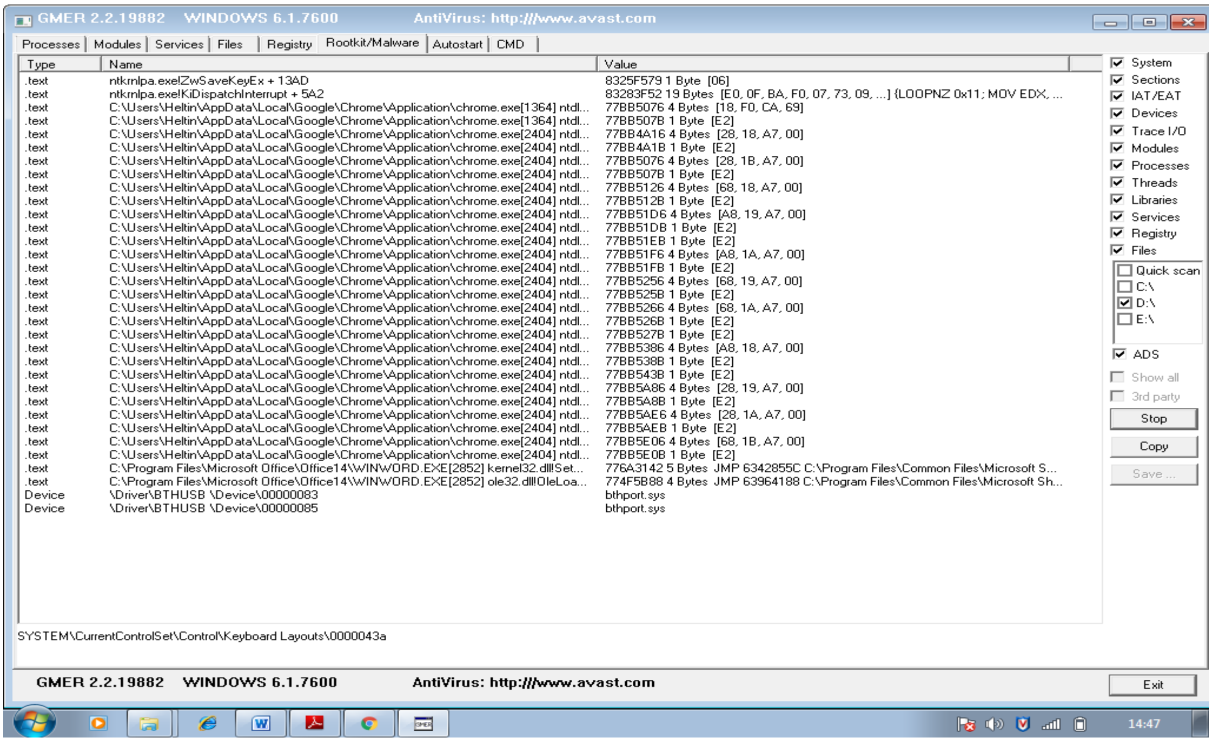


```
EventLabels
  .Default
  ActivatingDocument
  AppGPFault
  BlockedPopup
  BuildComplete
  BuildError
  BuildWarning
  CCSelect
  ChangeTheme
  Close
  CriticalBatteryAlarm
  DeviceConnect
  DeviceDisconnect
  DeviceFail
  DisNumbersSound
  EmptyRecycleBin
  FaxBeep
  FaxError
  FaxLineRings
  FaxSent
  FeedDiscovered
  HubOffSound
  HubOnSound
  HubSleepSound
  LowBatteryAlarm
  MailBeep
  Maximize
  MenuCommand
  MenuPopup
  Minimize
  MisrecoSound
  MoveMenuItem
  MSVC_HitBP
  MSVC_OutputError
  MSVC_OutputWarning
  Navigating
  Open
```

Name	Type	Data
(Default)	REG_SZ	Maximize
DispFileName	REG_SZ	@mmres.dll,-5833

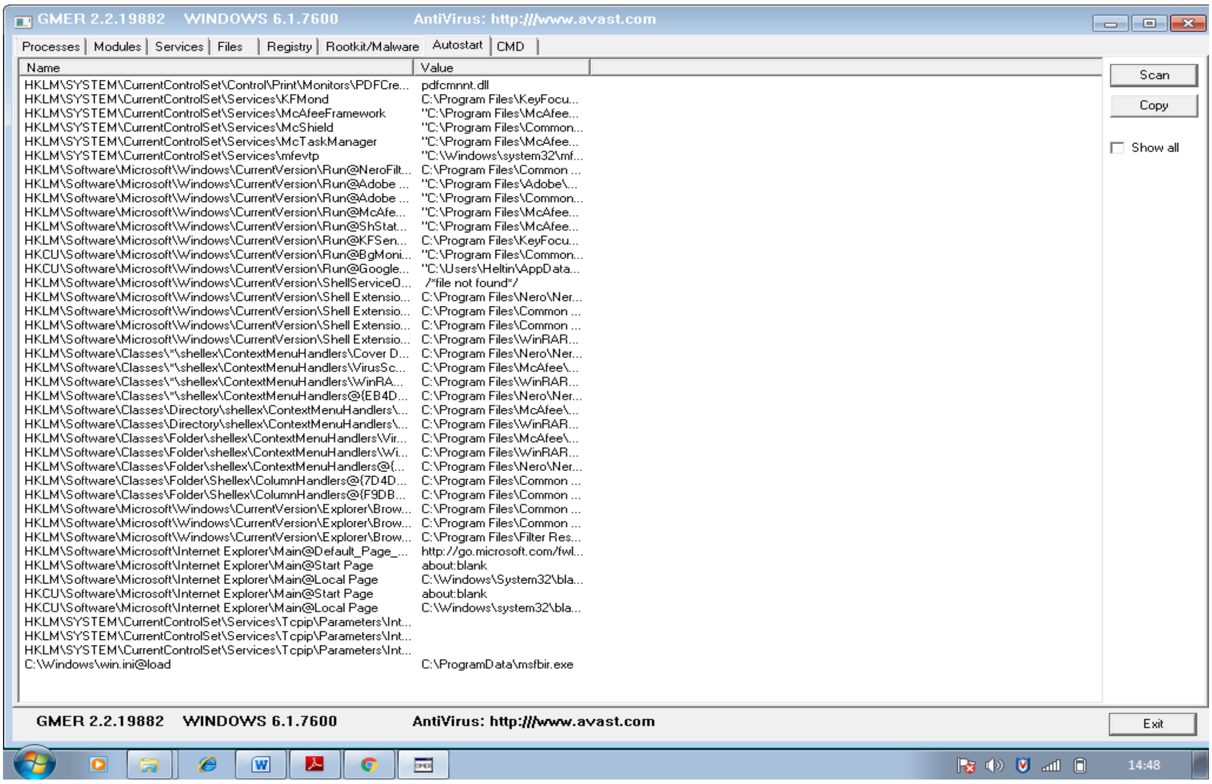
HKEY_CURRENT_USER\AppEvents\EventLabels\Maximize

GMER 2.2.19882 WINDOWS 6.1.7600 AntiVirus: http://www.avast.com



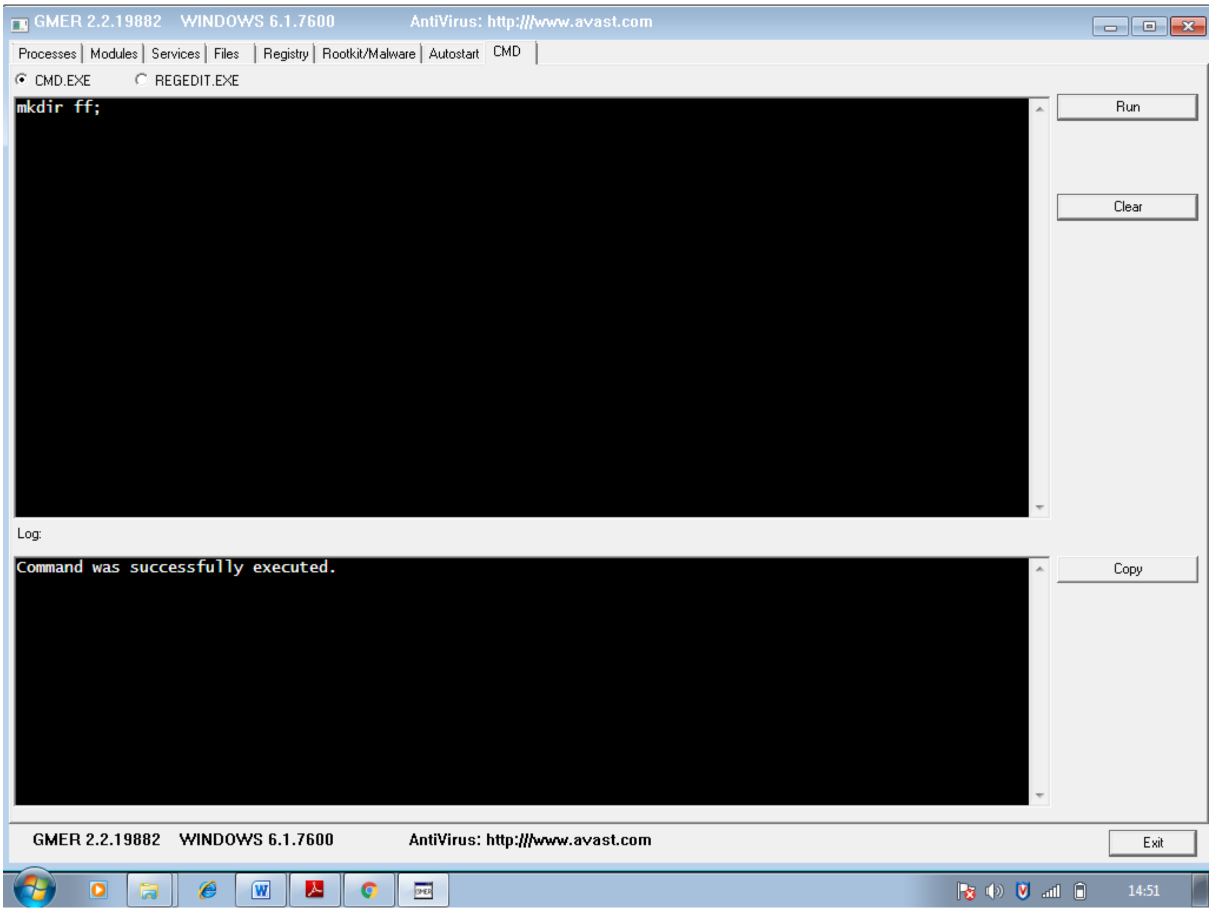
```
Type      Name                                                                         Value
.text     ntkrnlpa.exe!ZwSaveKeyEx + 13AD                                              8325F579 1 Byte  [06]
.text     ntkrnlpa.exe!KiDispatchInterrupt + 5A2                                       83283F52 19 Bytes  [E0, 0F, BA, F0, 07, 73, 09, ...] {LOOPNZ 0x11; MOV EDX, ...
.text     C:\Users\Heltin\AppData\Local\Google\Chrome\Application\chrome.exe[1364] ntdl...  77BB5076 4 Bytes  [18, F0, CA, 69]
.text     C:\Users\Heltin\AppData\Local\Google\Chrome\Application\chrome.exe[1364] ntdl...  77BB507B 1 Byte  [E2]
.text     C:\Users\Heltin\AppData\Local\Google\Chrome\Application\chrome.exe[2404] ntdl...  77BB4A16 4 Bytes  [28, 18, A7, 00]
.text     C:\Users\Heltin\AppData\Local\Google\Chrome\Application\chrome.exe[2404] ntdl...  77BB4A1B 1 Byte  [E2]
.text     C:\Users\Heltin\AppData\Local\Google\Chrome\Application\chrome.exe[2404] ntdl...  77BB5076 4 Bytes  [28, 1B, A7, 00]
.text     C:\Users\Heltin\AppData\Local\Google\Chrome\Application\chrome.exe[2404] ntdl...  77BB507B 1 Byte  [E2]
.text     C:\Users\Heltin\AppData\Local\Google\Chrome\Application\chrome.exe[2404] ntdl...  77BB5126 4 Bytes  [68, 18, A7, 00]
.text     C:\Users\Heltin\AppData\Local\Google\Chrome\Application\chrome.exe[2404] ntdl...  77BB512B 1 Byte  [E2]
.text     C:\Users\Heltin\AppData\Local\Google\Chrome\Application\chrome.exe[2404] ntdl...  77BB51D6 4 Bytes  [A8, 19, A7, 00]
.text     C:\Users\Heltin\AppData\Local\Google\Chrome\Application\chrome.exe[2404] ntdl...  77BB51DB 1 Byte  [E2]
.text     C:\Users\Heltin\AppData\Local\Google\Chrome\Application\chrome.exe[2404] ntdl...  77BB51EB 1 Byte  [E2]
.text     C:\Users\Heltin\AppData\Local\Google\Chrome\Application\chrome.exe[2404] ntdl...  77BB51F6 4 Bytes  [A8, 1A, A7, 00]
.text     C:\Users\Heltin\AppData\Local\Google\Chrome\Application\chrome.exe[2404] ntdl...  77BB51FB 1 Byte  [E2]
.text     C:\Users\Heltin\AppData\Local\Google\Chrome\Application\chrome.exe[2404] ntdl...  77BB5256 4 Bytes  [68, 19, A7, 00]
.text     C:\Users\Heltin\AppData\Local\Google\Chrome\Application\chrome.exe[2404] ntdl...  77BB525B 1 Byte  [E2]
.text     C:\Users\Heltin\AppData\Local\Google\Chrome\Application\chrome.exe[2404] ntdl...  77BB5266 4 Bytes  [68, 1A, A7, 00]
.text     C:\Users\Heltin\AppData\Local\Google\Chrome\Application\chrome.exe[2404] ntdl...  77BB526B 1 Byte  [E2]
.text     C:\Users\Heltin\AppData\Local\Google\Chrome\Application\chrome.exe[2404] ntdl...  77BB527B 1 Byte  [E2]
.text     C:\Users\Heltin\AppData\Local\Google\Chrome\Application\chrome.exe[2404] ntdl...  77BB5386 4 Bytes  [A8, 18, A7, 00]
.text     C:\Users\Heltin\AppData\Local\Google\Chrome\Application\chrome.exe[2404] ntdl...  77BB538B 1 Byte  [E2]
.text     C:\Users\Heltin\AppData\Local\Google\Chrome\Application\chrome.exe[2404] ntdl...  77BB543B 1 Byte  [E2]
.text     C:\Users\Heltin\AppData\Local\Google\Chrome\Application\chrome.exe[2404] ntdl...  77BB5A86 4 Bytes  [28, 19, A7, 00]
.text     C:\Users\Heltin\AppData\Local\Google\Chrome\Application\chrome.exe[2404] ntdl...  77BB5A8B 1 Byte  [E2]
.text     C:\Users\Heltin\AppData\Local\Google\Chrome\Application\chrome.exe[2404] ntdl...  77BB5AE6 4 Bytes  [28, 1A, A7, 00]
.text     C:\Users\Heltin\AppData\Local\Google\Chrome\Application\chrome.exe[2404] ntdl...  77BB5AEB 1 Byte  [E2]
.text     C:\Users\Heltin\AppData\Local\Google\Chrome\Application\chrome.exe[2404] ntdl...  77BB5E06 4 Bytes  [68, 1B, A7, 00]
.text     C:\Users\Heltin\AppData\Local\Google\Chrome\Application\chrome.exe[2404] ntdl...  77BB5E0B 1 Byte  [E2]
.text     C:\Program Files\Microsoft Office\Office14\WINWORD.EXE[2852] kernel32.dll!Set...  776A3142 5 Bytes  JMP 6342855C C:\Program Files\Common Files\Microsoft S...
.text     C:\Program Files\Microsoft Office\Office14\WINWORD.EXE[2852] ole32.dll!OleLoa...  774F5B88 4 Bytes  JMP 63964188 C:\Program Files\Common Files\Microsoft Sh...
Device    \Driver\BTHUSB \Device\00000083                                              bthport.sys
Device    \Driver\BTHUSB \Device\00000085                                              bthport.sys

SYSTEM\CurrentControlSet\Control\Keyboard Layouts\0000043a
```

☑ System
☑ Sections
☑ IAT/EAT
☑ Devices
☑ Trace I/O
☑ Modules
☑ Processes
☑ Threads
☑ Libraries
☑ Services
☑ Registry
☑ Files
☐ Quick scan
☐ C:\
☑ D:\
☐ E:\
☑ ADS
☐ Show all
☐ 3rd party



```
Name                                                                   Value
HKLM\SYSTEM\CurrentControlSet\Control\Print\Monitors\PDFCre...         pdfcmnnt.dll
HKLM\SYSTEM\CurrentControlSet\Services\KFMond                          C:\Program Files\KeyFocu...
HKLM\SYSTEM\CurrentControlSet\Services\McAfeeFramework                 "C:\Program Files\McAfee...
HKLM\SYSTEM\CurrentControlSet\Services\McShield                        "C:\Program Files\Common...
HKLM\SYSTEM\CurrentControlSet\Services\McTaskManager                   "C:\Program Files\McAfee...
HKLM\SYSTEM\CurrentControlSet\Services\mfevtp                          "C:\Windows\system32\mf...
HKLM\Software\Microsoft\Windows\CurrentVersion\Run@NeroFilt...         C:\Program Files\Common ...
HKLM\Software\Microsoft\Windows\CurrentVersion\Run@Adobe ...           "C:\Program Files\Adobe\...
HKLM\Software\Microsoft\Windows\CurrentVersion\Run@Adobe ...           "C:\Program Files\Common...
HKLM\Software\Microsoft\Windows\CurrentVersion\Run@McAfe...            "C:\Program Files\McAfee...
HKLM\Software\Microsoft\Windows\CurrentVersion\Run@ShStat...           "C:\Program Files\McAfee...
HKLM\Software\Microsoft\Windows\CurrentVersion\Run@KFSen...            C:\Program Files\KeyFocu...
HKCU\Software\Microsoft\Windows\CurrentVersion\Run@BgMoni...           "C:\Program Files\Common...
HKCU\Software\Microsoft\Windows\CurrentVersion\Run@Google...           "C:\Users\Heltin\AppData...
HKLM\Software\Microsoft\Windows\CurrentVersion\ShellServiceO...         /*file not found*/
HKLM\Software\Microsoft\Windows\CurrentVersion\Shell Extensio...       C:\Program Files\Nero\Ner...
HKLM\Software\Microsoft\Windows\CurrentVersion\Shell Extensio...       C:\Program Files\Common ...
HKLM\Software\Microsoft\Windows\CurrentVersion\Shell Extensio...       C:\Program Files\Common ...
HKLM\Software\Microsoft\Windows\CurrentVersion\Shell Extensio...       C:\Program Files\WinRAR...
HKLM\Software\Classes\*\shellex\ContextMenuHandlers\Cover D...         C:\Program Files\Nero\Ner...
HKLM\Software\Classes\*\shellex\ContextMenuHandlers\VirusSc...         C:\Program Files\McAfee\...
HKLM\Software\Classes\*\shellex\ContextMenuHandlers\WinRA...           C:\Program Files\WinRAR...
HKLM\Software\Classes\*\shellex\ContextMenuHandlers@{EB4D...           C:\Program Files\Nero\Ner...
HKLM\Software\Classes\Directory\shellex\ContextMenuHandlers\...        C:\Program Files\McAfee\...
HKLM\Software\Classes\Directory\shellex\ContextMenuHandlers\...        C:\Program Files\WinRAR...
HKLM\Software\Classes\Folder\shellex\ContextMenuHandlers\Vir...        C:\Program Files\McAfee\...
HKLM\Software\Classes\Folder\shellex\ContextMenuHandlers\Wi...         C:\Program Files\WinRAR...
HKLM\Software\Classes\Folder\shellex\ContextMenuHandlers@{...          C:\Program Files\Nero\Ner...
HKLM\Software\Classes\Folder\Shellex\ColumnHandlers@{7D4D...           C:\Program Files\Common ...
HKLM\Software\Classes\Folder\Shellex\ColumnHandlers@{F9DB...           C:\Program Files\Common ...
HKLM\Software\Microsoft\Windows\CurrentVersion\Explorer\Brow...        C:\Program Files\Common ...
HKLM\Software\Microsoft\Windows\CurrentVersion\Explorer\Brow...        C:\Program Files\Common ...
HKLM\Software\Microsoft\Windows\CurrentVersion\Explorer\Brow...        C:\Program Files\Filter Res...
HKLM\Software\Microsoft\Internet Explorer\Main@Default_Page_...        http://go.microsoft.com/fwl...
HKLM\Software\Microsoft\Internet Explorer\Main@Start Page              about:blank
HKLM\Software\Microsoft\Internet Explorer\Main@Local Page              C:\Windows\System32\bla...
HKCU\Software\Microsoft\Internet Explorer\Main@Start Page              about:blank
HKCU\Software\Microsoft\Internet Explorer\Main@Local Page              C:\Windows\system32\bla...
HKLM\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\Int...
HKLM\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\Int...
HKLM\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\Int...
C:\Windows\win.ini@load                                                C:\ProgramData\msfbir.exe
```

☐ Show all


GMER 2.2.19882 WINDOWS 6.1.7600 AntiVirus: http:///www.avast.com
Processes
Modules
Services
Files
Registry
Rootkit/Malware
Autostart
CMD
● CMD.EXE
○ REGEDIT.EXE
Run
Clear
Log:
Copy
GMER 2.2.19882 WINDOWS 6.1.7600 AntiVirus: http:///www.avast.com
Exit
14:51

```
mkdir ff;
```

```
Command was successfully executed.
```

RESULT:

Thus the study of installation of Rkhunter software and its variety of options were developed successfully.

DATE:	**SETUP A HONEY POT AND MONITOR THE HONEYPOT ON NETWORK**
EXPT NO: 14	

AIM

To setup a honey pot and monitor the honey pot on network.

PROCEDURE:

Step 1: Honey Pot is a device placed on Computer Network specifically designed to capture malicious network traffic.

Step 2: KF Sensor is the tool to setup as honeypot when KF Sensor is running it places a siren icon in the windows system tray in the bottom right of the screen. If there are no alerts then green icon is displayed.

Step 3: Download KF Sensor Evaluation Setu File from KF Sensor Website.

Step 4: Install with License Agreement and appropriate directory path.

Step 5: Reboot the Computer now.

Step 6: The KF Sensor automatically starts during windows boot Click Next to setup wizard.

Step 7: Select all port classes to include and Click Next.

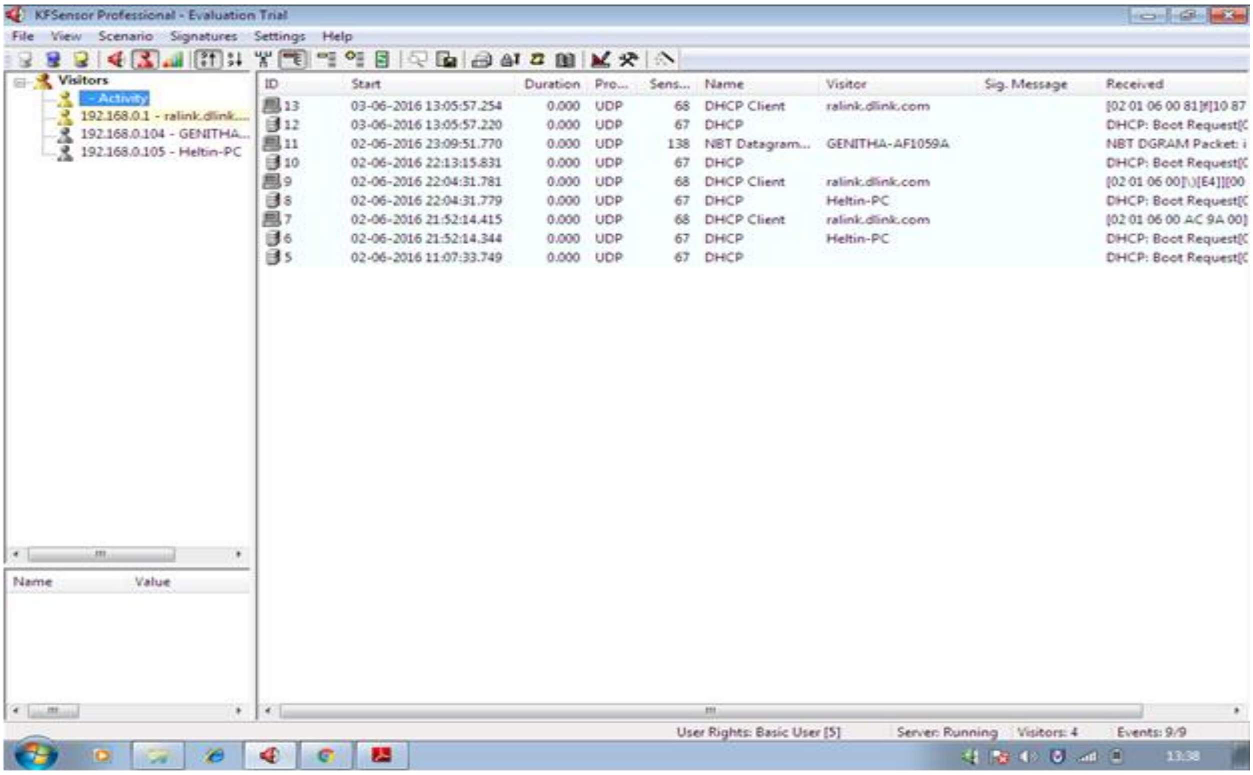
Step 8: Send the email and Send from email enter the ID and Click Next.

Step 9: Select the options such as Denial of Service [DOS], Port Activity, Proxy Emulsion, Network Port Analyzer, Click Next.

Step 10: Select Install as System service and Click Next.

Step 11: Click finish.

OUTPUT:

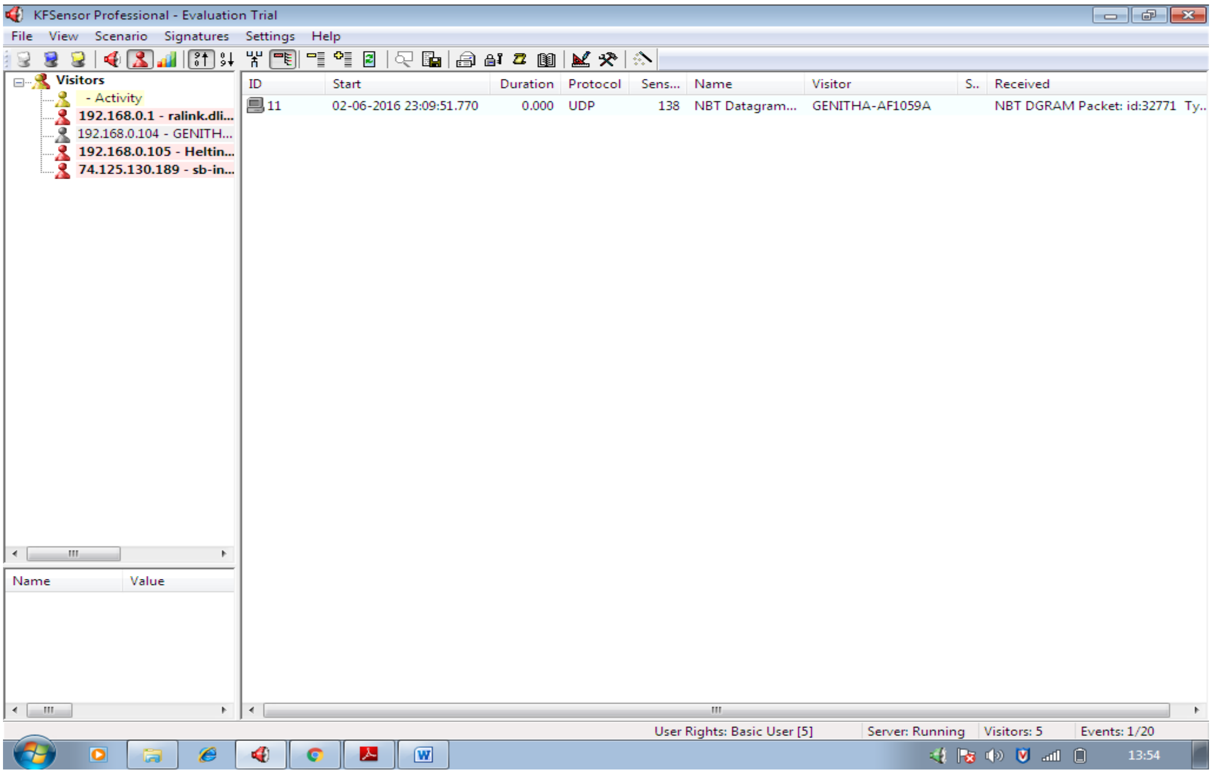


Visitors
- Activity
192.168.0.1 - ralink.dlink....
192.168.0.104 - GENITHA...
192.168.0.105 - Heltin-PC

ID	Start	Duration	Pro...	Sens...	Name	Visitor	Sig. Message	Received
13	03-06-2016 13:05:57.254	0.000	UDP	68	DHCP Client	ralink.dlink.com		[02 01 06 00 81]f][10 87
12	03-06-2016 13:05:57.220	0.000	UDP	67	DHCP			DHCP: Boot Request[C
11	02-06-2016 23:09:51.770	0.000	UDP	138	NBT Datagram...	GENITHA-AF1059A		NBT DGRAM Packet: i
10	02-06-2016 22:13:15.831	0.000	UDP	67	DHCP			DHCP: Boot Request[C
9	02-06-2016 22:04:31.781	0.000	UDP	68	DHCP Client	ralink.dlink.com		[02 01 06 00][\][E4]][00
8	02-06-2016 22:04:31.779	0.000	UDP	67	DHCP	Heltin-PC		DHCP: Boot Request[C
7	02-06-2016 21:52:14.415	0.000	UDP	68	DHCP Client	ralink.dlink.com		[02 01 06 00 AC 9A 00]
6	02-06-2016 21:52:14.344	0.000	UDP	67	DHCP	Heltin-PC		DHCP: Boot Request[C
5	02-06-2016 11:07:33.749	0.000	UDP	67	DHCP			DHCP: Boot Request[C

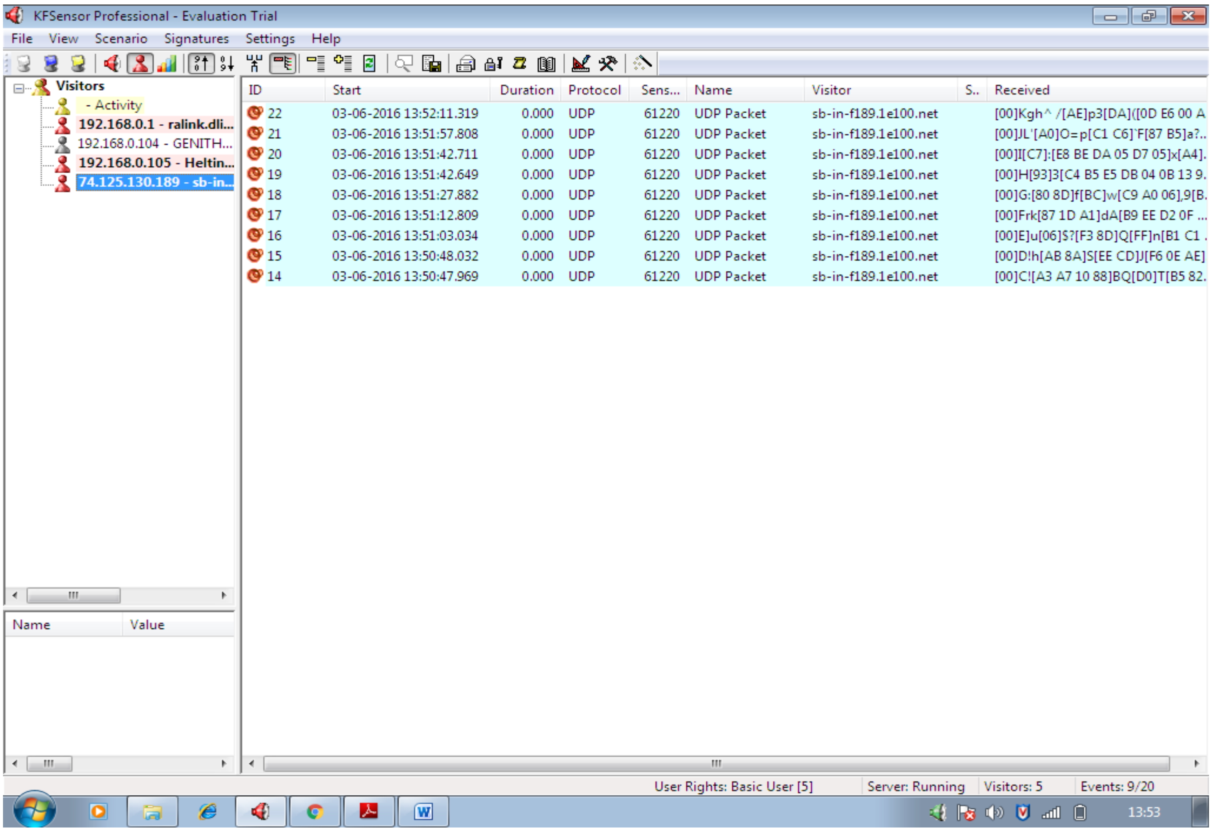
User Rights: Basic User [5] Server: Running Visitors: 4 Events: 9/9

13:38


KFSensor Professional - Evaluation Trial
File View Scenario Signatures Settings Help
Visitors
- Activity
192.168.0.1 - ralink.dli...
192.168.0.104 - GENITH...
192.168.0.105 - Heltin...
74.125.130.189 - sb-in...
Name Value
13:54

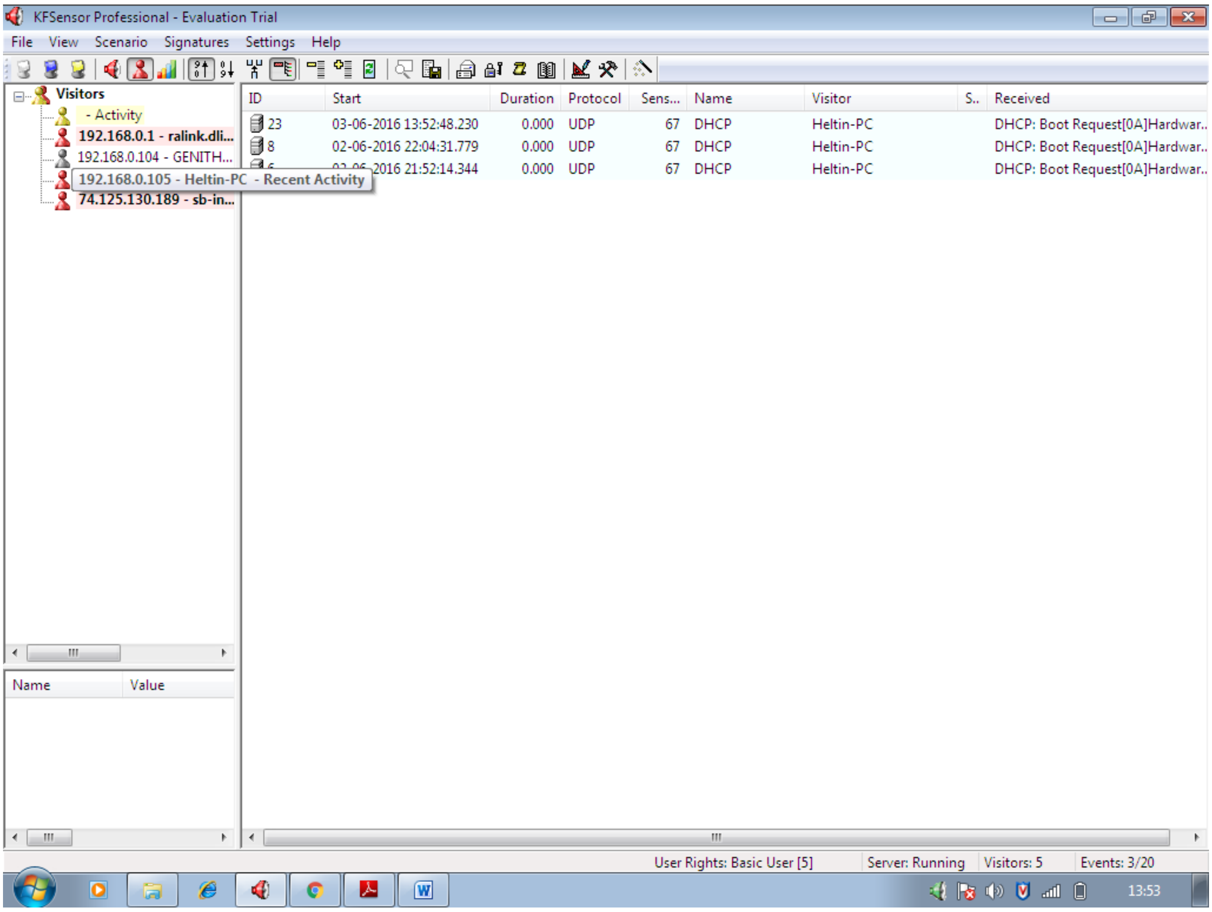
ID	Start	Duration	Protocol	Sens...	Name	Visitor	S..	Received
11	02-06-2016 23:09:51.770	0.000	UDP	138	NBT Datagram...	GENITHA-AF1059A		NBT DGRAM Packet: id:32771 Ty..

User Rights: Basic User [5] Server: Running Visitors: 5 Events: 1/20


KFSensor Professional - Evaluation Trial
File View Scenario Signatures Settings Help
Visitors
- Activity
192.168.0.1 - ralink.dli...
192.168.0.104 - GENITH...
192.168.0.105 - Heltin...
74.125.130.189 - sb-in...
Name Value
13:53

ID	Start	Duration	Protocol	Sens...	Name	Visitor	S..	Received
22	03-06-2016 13:52:11.319	0.000	UDP	61220	UDP Packet	sb-in-f189.1e100.net		[00]Kgh^ /[AE]p3[DA]([0D E6 00 A
21	03-06-2016 13:51:57.808	0.000	UDP	61220	UDP Packet	sb-in-f189.1e100.net		[00]JL'[A0]O=p[C1 C6]`F[87 B5]a?..
20	03-06-2016 13:51:42.711	0.000	UDP	61220	UDP Packet	sb-in-f189.1e100.net		[00]I[C7]:[E8 BE DA 05 D7 05]x[A4].
19	03-06-2016 13:51:42.649	0.000	UDP	61220	UDP Packet	sb-in-f189.1e100.net		[00]H[93]3[C4 B5 E5 DB 04 0B 13 9.
18	03-06-2016 13:51:27.882	0.000	UDP	61220	UDP Packet	sb-in-f189.1e100.net		[00]G:[80 8D]f[BC]w[C9 A0 06],9[B.
17	03-06-2016 13:51:12.809	0.000	UDP	61220	UDP Packet	sb-in-f189.1e100.net		[00]Frk[87 1D A1]dA[B9 EE D2 0F ...
16	03-06-2016 13:51:03.034	0.000	UDP	61220	UDP Packet	sb-in-f189.1e100.net		[00]E]u[06]$?[F3 8D]Q[FF]n[B1 C1 .
15	03-06-2016 13:50:48.032	0.000	UDP	61220	UDP Packet	sb-in-f189.1e100.net		[00]D!h[AB 8A]S[EE CD]J[F6 0E AE]
14	03-06-2016 13:50:47.969	0.000	UDP	61220	UDP Packet	sb-in-f189.1e100.net		[00]C![A3 A7 10 88]BQ[D0]T[B5 82.

User Rights: Basic User [5] Server: Running Visitors: 5 Events: 9/20



KFSensor Professional - Evaluation Trial

Visitors
- Activity
192.168.0.1 - ralink.dli...
192.168.0.104 - GENITH...
192.168.0.105 - Heltin-PC - Recent Activity
74.125.130.189 - sb-in...

ID	Start	Duration	Protocol	Sens...	Name	Visitor	S..	Received
23	03-06-2016 13:52:48.230	0.000	UDP	67	DHCP	Heltin-PC		DHCP: Boot Request[0A]Hardwar..
8	02-06-2016 22:04:31.779	0.000	UDP	67	DHCP	Heltin-PC		DHCP: Boot Request[0A]Hardwar..
6	02-06-2016 21:52:14.344	0.000	UDP	67	DHCP	Heltin-PC		DHCP: Boot Request[0A]Hardwar..

Name	Value

User Rights: Basic User [5] Server: Running Visitors: 5 Events: 3/20

RESULT:

Thus the study of setup a hotspot and monitor the hotspot on network has been developed successfully..

DATE:	**SIGN/ENCRYPT AND DECRYPT USING KLEOPATRA**
EXPT NO: 15	

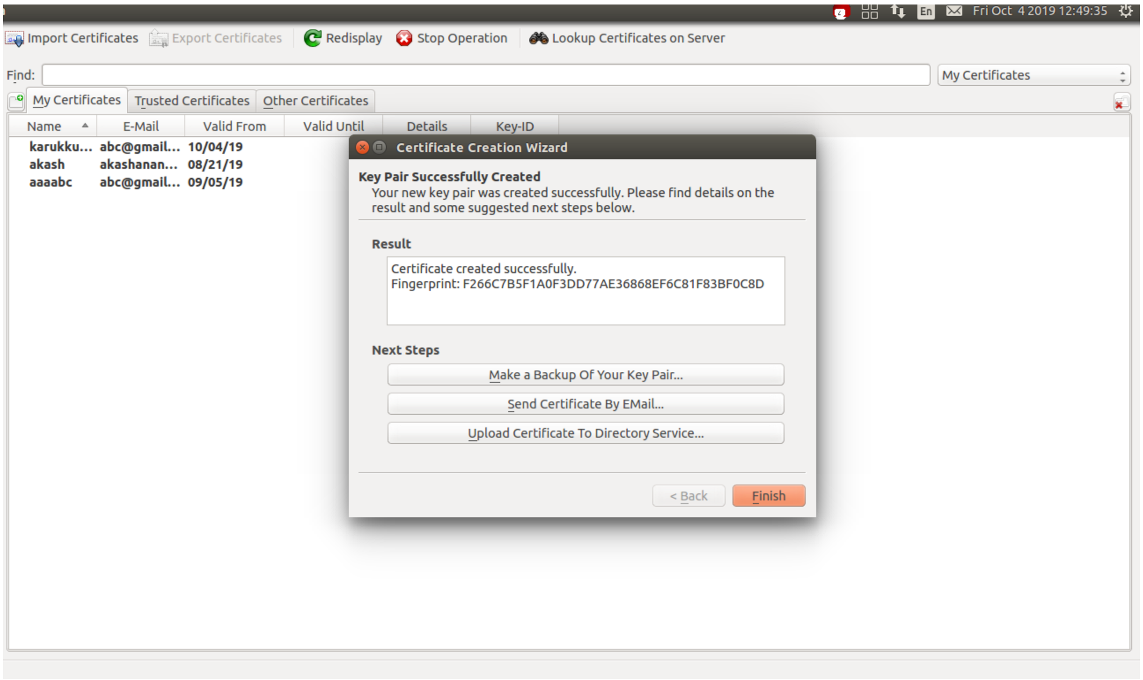
AIM:
To demonstrate the Sign/Encrypt and Decrypt the file using Kleopatra.

PROCEDURE:
Step 1: Start Kleopatra and Select File.
Step 2: Select Sign/encrypt files.
Step 3: Browse to a folder and select the file to sign/encrypt > select Open.

Signing:
Step 4: To sign the file select the radio button next to sign > select Next.
Step 5: Untick the option sign with OpenPGP
Step 6: Select the Certificate from the S/MIME Sign certificate. If it is not already selected.
Step 7: Click on Sign
Step 8: Enter the password for the secret key originally entered during certificate enrolment.

Encrypt:
Step 9: To encrypt the file select the radio button next to Encrypt > select Next
Step 10: Select recipient’s certificate > select Add/
Step 11: Select Encrypt to encrypt the file.
Step 12: Click on continue when warned that the file cannot be decrypted.

Execution:



Name	E-Mail	Valid From	Valid Until	Details	Key-ID
karukku...	abc@gmail...	10/04/19			
akash	akashanan...	08/21/19			
aaaabc	abc@gmail...	09/05/19			

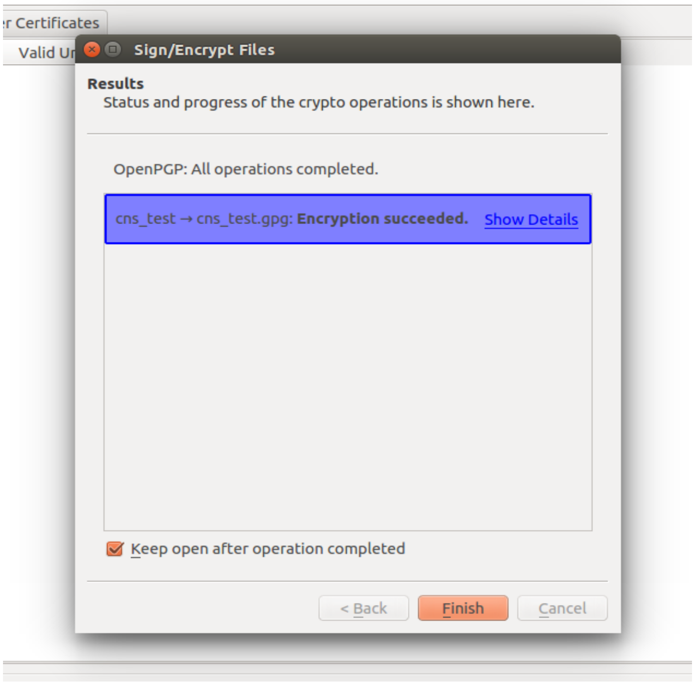
Certificate Creation Wizard

Key Pair Successfully Created
Your new key pair was created successfully. Please find details on the result and some suggested next steps below.

Result

Certificate created successfully.
Fingerprint: F266C7B5F1A0F3DD77AE36868EF6C81F83BF0C8D

Next Steps



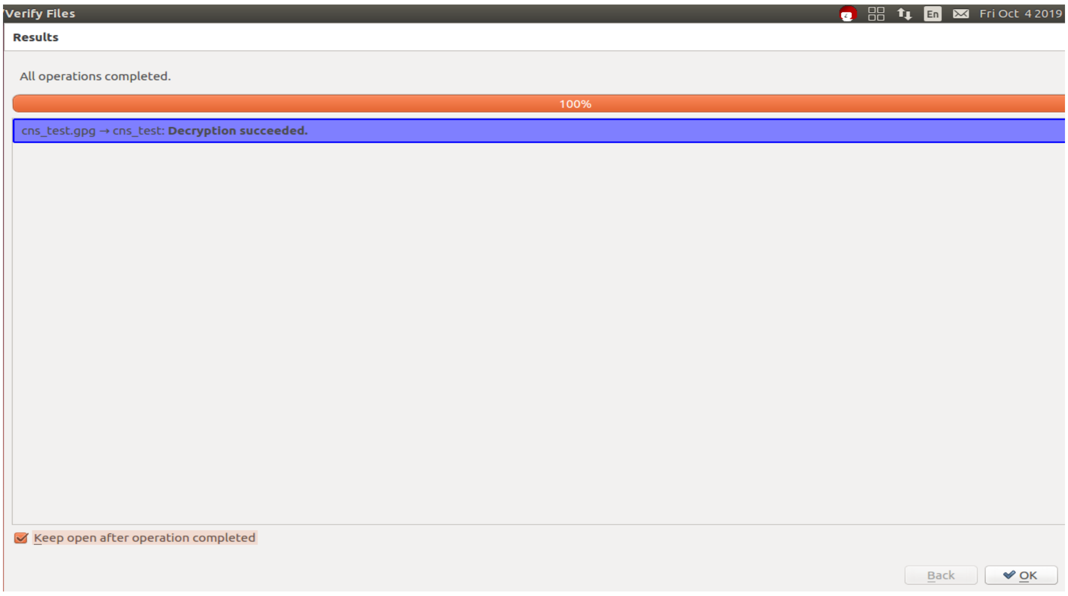
Results

Status and progress of the crypto operations is shown here.

OpenPGP: All operations completed.

cns_test → cns_test.gpg: **Encryption succeeded.** Show Details

☑ Keep open after operation completed



Results

All operations completed.

100%

cns_test.gpg → cns_test: **Decryption succeeded.**

☑ Keep open after operation completed

RESULT:

Thus the Sign/Encryption and Decryption of the file was generated successfully using kleopatra.