# Siddaganga Institute of Technology, Tumkur-572103

Department of Computer Science and Engineering

**CRYPTOGRAPHY AND NETWORK SECURITY LAB (7RCSL01)**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Student Name: A N V S Anudeep | | | USN: 1SI19CS017 | Batch No: A1 | | Date: 16/01/2023 | |
| **Evaluation:** | | | | | | | |
| **Write Up (10 marks)** | **Clarity in concepts (10 marks)** | **Implementation and execution of the algorithms (10 marks)** | | | **Viva (05 marks)** | | **Total (35 marks)** |
|  |  |  | | |  | |  |
|  | | | | | | | |
| Sl.No | Name of the Faculty In-Charge | | | | | | Signature |
| 1. | Dr. AS Poornima | | | | | |  |
| 2. | Ravi V | | | | | |  |
| **Question No: 9** Implement the following with respect to RC4:   1. Print first *n* key bytes generated by key generation process. 2. Illustrate encryption/decryption by accepting one-byte data as input on the above generated keys. | | | | | | | |
| Algorithm: i) To Generate the key, Its three-step process,  **Initialization of S:**    **Initial Permutation of S:**    **Stream Generation:** | | | | | | | |

#include <bits/stdc++.h>  
using namespace std;  
  
int main()

{  
    int S[256], T[256], keyStream[256];  
    string ptString, keyString, dtString = "";  
    int pt[256], ct[256], dt[256], j;  
  
    cout << "Enter message : "; cin >> ptString;  
    cout << "Enter key     : "; cin >> keyString;  
    int n = ptString.length();  
  
    cout << "\nPlain text \t: " ;  
    for(int i=0; i<n; i++)  
    {  
        pt[i] = ptString[i];  
        cout << pt[i] << " ";  
    }  
  
     
    for(int i=0; i<256; i++)  
    {  
        S[i] = i;  
        T[i] = (int)keyString[i%keyString.length()];  
    }  
  
     
    for(int i=0; i<256; i++)  
    {  
        j = (j + S[i] + T[i]) % 256;  
        swap(S[i], S[j]);  
    }  
  
     
    cout << "\nKey Stream \t: ";  
    j=0;  
    for(int i=0; i<n; i++)  
    {  
        j = (j + S[i]) % 256;  
        swap(S[i], S[j]);  
        int t = (S[i] + S[j]) % 256;  
        keyStream[i] = S[t];  
        cout << keyStream[i] << " ";  
    }  
  
     
    cout << "\nCipher Text \t: ";  
    for(int i=0; i<n; i++)

{

ct[i] = pt[i] ^ keyStream[i];  
        cout << ct[i] << " ";  
}  
  
     
    cout << "\nDecrypted text \t: " ;  
    for(int i=0; i<n; i++)  
    {  
        dt[i] = ct[i] ^ keyStream[i];  
        cout << dt[i] << " ";  
        dtString += (char)dt[i];  
    }  
    cout << "\nDecrypted text \t: " << dtString << endl;  
}

