Experiment No. 2

Title: Transposition Cipher

Batch: B2 Roll No.: 16010420117 Experiment No.: 2

Aim: To implement transposition cipher – Row transposition and column transposition cipher.

Resources needed: Windows/Linux

Theory

Pre Lab/ Prior Concepts:

Symmetric-key algorithms are a class of algorithms for cryptography that use the same cryptographic keys for both encryption of plaintext and decryption of cipher text. The keys may be identical or there may be a simple transformation to go between the two keys. The keys, in practice, represent a shared secret between two or more parties that can be used to maintain a private information link. This requirement that both parties have access to the secret key is one of the main drawbacks of symmetric key encryption, in comparison to public-key encryption. Symmetric-key encryption can use either stream ciphers or block ciphers. Transposition Cipher is block cipher. Ancient cryptographic systems are classified as: Substitution and Permutation/Transposition Ciphers.

Transposition Cipher/Permutation Cipher

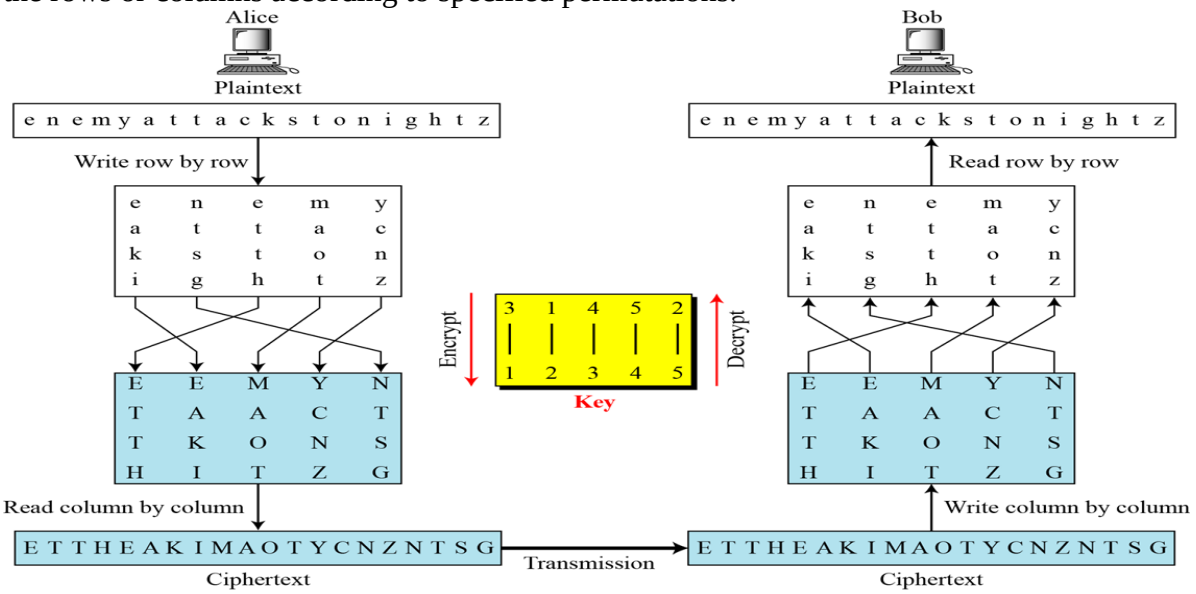
A transposition cipher rearranges (permutes) symbols in a block without altering actual values. It has the same frequency distribution as the original text .So it is easily recognizable.

EXAMPLE :

Plaintext: HELLO MY DEAR
Cipher text: ELHLMDOYAER

There are varieties of transposition ciphers like: keyless and keyed transposition ciphers.

Following figure shows the combination of both keyed and keyless. To encrypt with a transposition cipher, we first write the plaintext into a matrix of a given size and then permute the rows or columns according to specified permutations.



For the transposition, the key consists of the size of the matrix and the row or column permutations. The recipient who knows the key can simply put the cipher text into the appropriate sized matrix and undo the permutations to recover the plaintext.
Unlike a simple substitution, the transposition does nothing to disguise the letters that appear in the message But it does appear to thwart an attack that relies on the statistical information contained in the plaintext, since the plaintext statistics are disbursed throughout the cipher text. The double transposition is not a trivial cipher to break.

Activity :

1) As shown in the figure above, implement row transposition and column transposition ciphers.

Implementation:

The program should have encryption function and decryption function for each cipher. Function should take message and a key as input from the user and display the expected output.

Results: (Program with output as per the format)

COLUMNAR TRANSPOSITION CIPHER

Code:

```
def columnar_encrypt(text,key):
    # Adding _ to the difference place
    incompleteReplacedString = '_'*(len(key)-(len(text)%len(key)))
    updatedString = text + incompleteReplacedString
    sortedKeyWord = sorted(key)

    matrix = []

    while(len(updatedString)!=0):
        matrix.append(updatedString[:len(key)])
        updatedString=updatedString[len(key):]
    keyWord2 = list(key)

    cipher = ""
    for i in sortedKeyWord:
        index = keyWord2.index(i)
        for j in range(len(matrix)):
            cipher += matrix[j][index:index+1]
            matrix[j] = matrix[j][:index]+matrix[j][index+1:]
        keyWord2.remove(i)

    return cipher

def columnar_decrypt(cipher,key):
    sortedKeyWord = sorted(key)

    matrix = []
    noOfRows = len(cipher)//len(key)
    while(len(cipher)!=0):
        matrix.append(cipher[:noOfRows])
        cipher=cipher[noOfRows:]

    decryptedText=""
    newMatrix = []
    for i in key:
        index = sortedKeyWord.index(i)
        newMatrix.append(matrix[index])
    for i in range(noOfRows):
        for j in range(len(newMatrix)):
            decryptedText = decryptedText + newMatrix[j][i]
    return decryptedText.replace("_","")
```

```
text = input("Enter word to be encrypted: ").upper()
key = input("Enter key: ").upper()
cipher = columnar_encrypt(text,key)
print("Encrypted Word is: ",cipher)
print("Decrypted Word is: ",columnar_decrypt(cipher,key))
```

Output:

```
PS C:\Users\infin\OneDrive\Desktop\CODE\CollegeSubmissions\TY\INS\Experiements\Experiment 2\code> python .\columnarTrans
position.py
Enter word to be encrypted: My name is Neckrozma Antheros
Enter key: 43561
Encrypted Word is:  ASKAH_YENOARMM R E  EZNONICMTS
Decrypted Word is:  MY NAME IS NECKROZMA ANTHEROS
```

ROW TRANSPOSITION CIPHER

CODE:

```
def row_encrypt(text,key):
    incompleteReplacedString = '_'*(len(key)-(len(text)%len(key)))
    updatedString = text + incompleteReplacedString

    sortedKeyWord = sorted(key)

    matrix = []

    while(len(updatedString)!=0):
        matrix.append(updatedString[:len(key)])
        updatedString=updatedString[len(key):]

    key2 = list(key)

    encryptMatrix = []
    for i in matrix:
        encryptMatrix.append("")

    for i in range(len(sortedKeyWord)):
        index = key2.index(sortedKeyWord[i])
        for j in range(len(matrix)):
            letter = matrix[j][index:index+1]
            encryptMatrix[j] = encryptMatrix[j]+letter

    return "".join(encryptMatrix)

def row_decrypt(cipher,key):
```

```
    matrix = []
    while(len(cipher)!=0):
        matrix.append(cipher[:len(key)])
        cipher=cipher[len(key):]

    cipher = []
    for i in matrix:
        cipher.append("")

    key2 = list(key)
    sortedKeyWord = sorted(key)

    for i in range(len(key2)):
        index = sortedKeyWord.index(key2[i])
        for j in range(len(matrix)):
            letter = matrix[j][index:index+1]
            cipher[j] = cipher[j]+letter

    return ("".join(cipher)).replace("_","")

text = input("Enter word to be encrypted: ")
key = input("Enter key: ")
cipher = row_encrypt(text,key)
print("Encrypted Word is: ",cipher)
print("Decrypted Word is: ",row_decrypt(cipher,key))
```

Output:

```
PS C:\Users\infin\OneDrive\Desktop\CODE\CollegeSubmissions\TY\INS\Experiements\Experiment 2\code> python .\rowTransposit
ion.py
Enter word to be encrypted: mynameisrintronglade
Enter key: 432671
Encrypted Word is:  enymamtrsiinanorgl__ed__
Decrypted Word is:  mynameisrintronglade
```

Questions:

1) Compare substitution ciphers and transposition/permutation ciphers. comment on confusion and diffusion properties of both.

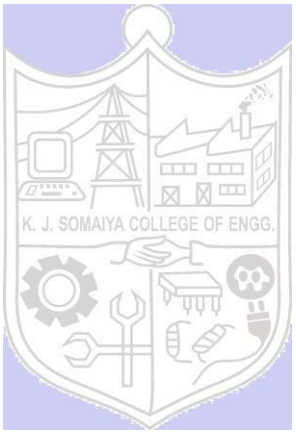
Sr No.	Substitution Cipher Technique	Transposition Cipher Technique
1	In substitution Cipher Technique, plain text characters are replaced with other characters, numbers and symbols.	In transposition Cipher Technique, plain text characters are rearranged with respect to the position
2	Substitution Cipher's forms are: Mono alphabetic substitution cipher and poly alphabetic substitution cipher.	Transposition Cipher's forms are: Key-less transposition cipher and keyed transposition cipher.
3	In substitution Cipher Technique, character's identity is changed while its position remains unchanged.	While in transposition Cipher Technique, The position of the character is changed but character's identity is not changed.
4	In substitution Cipher Technique, The letter with low frequency can detect plain text.	While in transposition Cipher Technique, The Keys which are nearer to correct key can disclose plain text.
5	The example of substitution Cipher is Caesar Cipher.	The example of transposition Cipher is Rail Fence Cipher

Substitution Cipher Technique:

In Substitution Cipher Technique plain text characters are replaced with other characters, numbers and symbols as well as in substitution Cipher Technique, character's identity is changed while its position remains unchanged.

Transposition Cipher Technique:

Transposition Cipher Technique rearranges the position of the plain text's characters. In transposition Cipher Technique, The position of the character is changed but character's identity is not changed.



Outcomes: CO1: Describe the basics of information security.

Conclusion: We learnt about transposition ciphers and implemented the code and saw the working of simple columnar and row transposition ciphers.

Grade: AA / AB / BB / BC / CC / CD /DD

Signature of faculty in-charge with date

References: Books/ Journals/ Websites:

1. Behrouz A. Forouzan, “Cryptography and Network Security”, Tata McGraw Hill
2. Mark Stamp, “Information Security Principles and Practice”, Wiley.
3. William Stalling, “Cryptography and Network Security”, Prentice Hall