

1. Write a Python script to encrypt columnar transposition using keyword.

Sol:

```
kali@kali: ~  
File Actions Edit View Help  
def encrypt_columnar_transposition(message, keyword):  
    message = message.replace(" ", "")  
    key_length = len(keyword)  
    columns = [''] * key_length  
    sorted_key = sorted(list(keyword))  
    for index, char in enumerate(message):  
        columns[index % key_length] += char  
    encrypted_message = ''  
    for char in sorted_key:  
        column_index = keyword.index(char)  
        encrypted_message += columns[column_index]  
    return encrypted_message  
message = "this is a secret message"  
keyword = "keyword"  
encrypted_message = encrypt_columnar_transposition(message, keyword)  
print("Encrypted Message:", encrypted_message)
```

Output:

```
(kali@kali)-[~]  
$ vi lab7.py  
  
(kali@kali)-[~]  
$ python lab7.py  
Encrypted Message: amhestseiegstesraics
```

2. Write a Python script to encrypt double columnar transposition.

Sol:

```
kali@kali: ~  
File Actions Edit View Help  
def columnar_transposition_encrypt(plaintext, key1, key2):  
    def create_grid(text, key):  
        grid = [''] * len(key)  
        for i, char in enumerate(text):  
            grid[i % len(key)] += char  
        return grid  
    grid1 = create_grid(plaintext, key1)  
    intermediate_text = ''.join(grid1[i] for i in sorted(range(len(key1)), key=lambda k: key1[k]))  
    grid2 = create_grid(intermediate_text, key2)  
    cipher_text = ''.join(grid2[i] for i in sorted(range(len(key2)), key=lambda k: key2[k]))  
    return cipher_text  
plaintext = "thisisasecretmessage"  
key1 = "4312"  
key2 = "3214"  
cipher_text = columnar_transposition_encrypt(plaintext, key1, key2)  
print("Encrypted Text:", cipher_text)
```

Output:

```
(kali@kali)-[~]  
$ vi lab7.py  
  
(kali@kali)-[~]  
$ python lab7.py  
Encrypted Text: rshatasemeigscieests
```

3. Write a Python script to encrypt the message "She is listening" using the 6-character keyword "PASCAL" with Vigenere cipher.

Sol:

```
File Actions Edit View Help
def vigenere_encrypt(plaintext, keyword):
    keyword_repeated = (keyword * (len(plaintext) // len(keyword))) + keyword[:len(plaintext) % len(keyword)]
    ciphertext = ""
    for p, k in zip(plaintext, keyword_repeated):
        if p.isalpha():
            shift = ord(k.lower()) - ord('a')
            encrypted_char = chr((ord(p.lower()) - ord('a') + shift) % 26 + ord('a'))
            if p.isupper():
                encrypted_char = encrypted_char.upper()
            ciphertext += encrypted_char
        else:
            ciphertext += p
    return ciphertext
plaintext = "She is listening"
keyword = "PASCAL"
ciphertext = vigenere_encrypt(plaintext, keyword)
print("Encrypted Text:", ciphertext)
```

Output:

```
(kali@kali)-[~]
$ vi lab7.py

(kali@kali)-[~]
$ python lab7.py
Encrypted Text: Hhw id lautpcifi
```

4. Write a Python script to encrypt and decrypt Hill cipher Bonus Point
5. Write a Python script to perform Kasiski test.