Experiment 1

Chinmay Parikh A059

Questions

1. What are the goals of cryptography?

2. Explain the following terms with respect to cryptography?

   - Plain Text
   - Cipher Text
   - Encryption
   - Decryption
   - Key
   - Key Space

3. What is brute force attack?

4. Explain Statistical analysis attack.

Answers:

1. - Confidentiality : It means to keep the information secret from everyone, except from those who are authorized, Confidentiality is the protection of transmitted data from attacks.
   - Data integrity : Ensuring that the information has not been altered by unauthorized or by unknown means. One must have the ability to detect data manipulation by unauthorized parties.
   - Authentication : Authentication is a service related to identification. This function applies to both entities and information.
   - Non-repudiation : Non-repudiation prevents either sender or receiver from denying a message. Thus, when a message is sent, the receiver can prove that the message was send by the alleged sender.

2. - Plain Text is normal information , the orignal message to be sent.
   - Cipher Text it's plain text which is processed through a cipher. Normally garbled to read.
   - Cipher is an algorithm that can process plain text to cipher text and vice versa.
   - Encryption is the process of encoding information or message in such a way that only authorized parties can read it.
   - Decryption is the process or decoding information or message which had been encrypted into a secret format. It requires a key or password.

- Key is a byte string used the cipher to encode and decode the message. In some cases the same key is used for both encryption and decryption.
- Key Space is all possible combinations of the characters in the keyspace used to generate a key.

3. Brute force attack is when one tries all the possible combinations of password/key in order to gain illegal access to information.

4. In cryptanalysis, frequency analysis is the study of the frequency of letters or groups of letters in a ciphertext. The method is used as an aid to breaking classical ciphers. Frequency analysis is based on the fact that, in any given stretch of written language, certain letters and combinations of letters occur with varying frequencies. Moreover, there is a characteristic distribution of letters that is roughly the same for almost all samples of that language.In some ciphers, such properties of the natural language plaintext are preserved in the ciphertext, and these patterns have the potential to be exploited in a ciphertext-only attack.

Caeser.py

```
def Caeser(text, key):
    output = ""
    for char in text:
        new = ord(char) + key
        if new > 127:
            new = new - 127
        elif new < 0:
            new = new + 127
        output = output + str(unichr(new))
    return output

def Brutus(text):
    for r in range(1,127):
        print str(r)+" : "+Caeser(text, -r)

def main():
    print "1. Encrypt"
    print "2. Decrypt"
    print "3. BruteForce"
    print "Input Choice 1 | 2 | 3"
    choice = int(raw_input())
    if choice == 1:
        print "Plain Text Please : "
        text = raw_input()
        print "Shift Right By ?(INTEGER PLOX) : "
        key = int(raw_input())
        print Caeser(text, key)
    elif choice == 2:
        print "Cipher Text Please : "
        text = raw_input()
        print "Shift Left By ?(INTEGER PLOX) : "
        key = int(raw_input())
        print Caeser(text, -key)
    elif choice == 3:
        print "Cipher Text Please : "
        text = raw_input()
        Brutus(text)
    else:
        print "Wrong Code"

if __name__ == ’__main__’:
            main()
```

Output:

```
1. Encrypt
2. Decrypt
3. BruteForce
Input Choice 1 | 2 | 3
1
Plain Text Please :
Hello!
Shift Right By ?(INTEGER PLOX) :
3
Khoor$

1. Encrypt
2. Decrypt
3. BruteForce
Input Choice 1 | 2 | 3
2
Cipher Text Please :
Khoor$
Shift Left By ?(INTEGER PLOX) :
3
Hello!

1. Encrypt
2. Decrypt
3. BruteForce
Input Choice 1 | 2 | 3
```

```
3
Cipher Text Please :
Khoor$
1 : Jgnnq#
2 : Ifmmp"
3 : Hello!
4 : Gdkkn
.
.
.
.
.
.
114 : Xu||1
115 : Wt{{~0
116 : Vszz}/
117 : Uryy|.
118 : Tqxx{-
119 : Spwwz,
120 : Rovvy+
121 : Qnuux*
122 : Pmttw)
123 : Olssv(
124 : Nkrru'
125 : Mjqqt&
126 : Lipps%
```