CS 435: Introduction to Cryptography **Spring 2020**

Homework 1(Solutions)

Professor Somesh Jha **Due:** February 12

1. Use definition 2 of perfect secrecy (Lemma 2.4 of the textbook) to prove that the Vigenère cipher is not perfectly secret.

$$\Pr[\mathsf{Enc}_K(m) = c] = \Pr[\mathsf{Enc}_K(m') = c] \tag{2.1}$$

Lemma 2.4. *An encryption scheme* $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *with message space* $\mathcal{M}$ *is perfectly secret if and only if Equation (2.1) holds for every* $m, m' \in \mathcal{M}$ *and every* $c \in \mathcal{C}$.

Solution:

Since definition 2 of perfect secrecy should hold true for every $m, m' \in \mathcal{M}$ and every $c \in \mathcal{C}$, if we are able to come up with m, m' and c such that Equation 2.1 does not hold true, then we are done.

Let $m = pqrs$, $m' = cdcd$ and $c = dfdf$.

In the Vigenère cipher each letter of plaintext is shifted by a fixed number given by the letter at the corresponding position in the key. Since the key is generally shorter than the text, the key is sequentially applied to the plaintext. Thus for each period-size(period = key-length) section of the plaintext, letters in the same positions (i.e. jth stream) are shifted by the same amount.

Assuming the length of the key as 2, encrypting m to c would require a K which shifts $pq \to df$ and $rs \to df$.

This is not possible. Thus, $\Pr[\mathsf{Enc}_K(m) = c] = 0$ (1).

Now $K == ab$ would encrypt m' as c. Thus, $\Pr[\mathsf{Enc}_K(m') = c] > 0$ (2).

By (1) and (2) we can say that the Vigenère cipher is not perfectly secret.

2. Consider a variant of the Vigenère cipher where instead of a word or short phrase, the key instead consists of a book or some other English-language text that is much longer than the message to be encrypted. Using this cipher, the key is never repeated, so our standard methods for retrieving the key length will fail. Now assume that Bob is a sleeper agent and Alice is his handler. Alice, using this cipher, has sent Bob a ciphertext that reads

```
eplxiuuynlozlrshw
```

The plaintext is known to contain the day of the week that Bob is supposed to receive the dead drop, followed by the day of the week he is supposed to flee the

country. Determine which plaintext Alice sent to Bob, and explain how you reached your answer.

Note. Each ciphertext character c_i is equal to $m_i + k_i \pmod{26}$, where m_i is the i-th character of the plaintext message and k_i is the i-th character of the key. In particular, the alphabet is indexed from 0, so 'a' corresponds to 0, 'b' corresponds to 1, and so on.

Solution:

Notice that the length of the ciphertext is 17. Comparing the length of the days (Monday $\rightarrow$ 6, Tuesday $\rightarrow$ 7, Wednesday $\rightarrow$ 9, Thursday $\rightarrow$ 8, Friday $\rightarrow$ 6, Saturday $\rightarrow$ 8 and Sunday $\rightarrow$ 6), one can figure out that there are two possibilities for the plain text, either "WednesdayThursday" or "WednesdaySaturday".

Both the plaintexts have a common prefix of "Wednesday". Using the plaintext and the ciphertext we can back calculate the first part of the key which is "I like cryp". This is suggestive of the complete key "I like cryptography". One can now use this key to figure out that the plaintext is "WednesdaySaturday".

3. a. Assume an attacker knows that a user's password is either mnop or byce. Say the user encrypts his password using the shift cipher, and the attacker sees the resulting ciphertext. Show how the attacker can determine the user's password, or explain why this is not possible.

   Solution:

   Notice that the letters mnop are all consecutive. This is not the case with byce. Since the shift cipher moves all letters in the password by an equal amount, the resulting cipher text for mnop will also be composed of consecutive letters. If the cipher text is composed of consecutive letters, the password is mnop, if not, then the password is byce.

   b. Repeat part (a) for the Vigenère cipher using period 2, using period 3, and using period 4.

   Solution:

   Period 2

   A period 2 key implies that the 1st and 3rd characters of the password will be shifted by the same amount. Similarly, the 2nd and 4th characters will be shifted by the same amount.
   In mnop the difference between the 1st and 3rd character is 2 and the difference between the 2nd and 4th character is 2.
   In **byce** the difference between the 1st and 3rd character is 1 and the difference between the 2nd and 4th character is 6.
   Thus if the difference between the 1st and 3rd character of ciphertext is 2, the attacker will know that the password is mnop else if the difference is 1, the password is byce. Similarly, the difference between 2nd and 4th characters can be exploited.

   Period 3

   A period 3 key implies that the 1st and 4th character of the plain text will be shifted by the same amount.

In mnop the difference between the 1st and 4th character is 3. In byce the difference between the 1st and 4th character is 3. Since there is no difference which can be exploited, the attacker will not be able to tell whether the password is mnop or byce.

Period 4

With period 4, for any ciphertext c, there exists keys K and K' such that $\mathsf{Enc}_K(\mathsf{mnop}) = c$ and $\mathsf{Enc}_{K'}(\mathsf{byce}) = c$. Thus the attacker will not be able to tell whether the password was mnop or byce.

4. Describe the improved attack on shift cipher using statistical test (refer to pages 12 and 13 in the book). What happens if you replace I_j by $\sum_{i=0}^{25} q_{i+j}^2$? Does the attack work?

Solution:

The brute force attack of checking all 26 possibilities of shift and choosing the text which "makes sense" is not convenient and difficult to automate. This is why we have the improved attack which does not have any of these drawbacks. Let p_i with $0 \leq p_i \leq 1$ denote the frequency of the ith letter in normal English text (ignoring spaces, punctuation, etc.). Calculation using Figure 1.3 of the book gives:

$$\sum_{i=0}^{25} p_i^2 \approx 0.065$$

Now, say we are given some ciphertext and let q_i denote the frequency of the ith letter of the alphabet in this ciphertext; i.e., q_i is simply the number of occurrences of the ith letter of the alphabet in the ciphertext divided by the length of the ciphertext. If the key is k, then p_i should be roughly equal to q_{i+k} for all i, because the ith letter is mapped to the $(i+k)$th letter. (We use i+k instead of the more cumbersome [i+k mod 26].) Thus, if we compute

$$I_j \stackrel{\text{def}}{=} \sum_{i=0}^{25} p_i \cdot q_{i+j}$$

for each value of $j \in \{0, \ldots, 25\}$, then we expect to find that $I_k \approx 0.065$ (where k is the actual key), whereas I_j for $j \neq k$ will be different from 0.065. This leads to a key-recovery attack that is easy to automate: compute I_j for all j, and then output the value k for which I_k is closest to 0.065.

If we replace I_j by $\sum_{i=0}^{25} q_{i+j}^2$, this attack will not work because we are summing the frequencies of the shifted letters and it does not matter what the shift is because the letters wrap around after **z** and thus the sum remains constant. Mathematically, since the sum in the subscript of q is modulo 26, thus for any integer j

$$\sum_{i=0}^{25} q_{i+j}^2 = \sum_{i=0}^{25} q_i^2 \approx \sum_{i=0}^{25} p_i^2 \approx 0.065$$

For example:

$$\sum_{i=0}^{25} q_{i+j}^2 = q_j^2 + q_{j+1}^2 + q_{j+2}^2 + \cdots + q_{j+25}^2$$

and because of modulo 26 addition,

$$\sum_{i=0}^{25} q_{i+j+2}^2 = q_{j+2}^2 + q_{j+3}^2 + \cdots + q_{j+26}^2 + q_{j+27}^2 = q_{j+2}^2 + q_{j+3}^2 + \cdots + q_j^2 + q_{j+1}^2$$

which is the same as before. As we cannot differentiate between two shift amounts, the attack fails.