# Homework #1 for CS 419— Computer Security

## 1 Known Plaintext Attack (10 pt)

In our slides, we illustrate how to break “Mono-alphabetic substitution cipher” by frequency analysis. As a result, we need a lot of ciphertext to build the frequency histogram. In this question, let’s try an alternative and powerful attack—Known Plaintext Attack. Below, we are given a sentence of ciphertext and its corresponding plaintext. Please recover the secret key as good as you can and use your corrupted secret key to attack another sentence of ciphertext.

Known Ciphertext: NBZQARAV KTAH ETQ CJII STL WLHQ MZCAH STL HQVZEOAV.

Corresponding Plaintext: Whatever does not kill you just makes you stranger.

The ciphertext you want to corrupt: IAQ LH MZCA VLQOAVH OVAZQ ZOZJE

## 2 Chosen Ciphertext Attack (10 pt)

In the case of chosen ciphertext attack, we are considering a more powerful adversary, where the adversary can make a query to an oracle $\mathcal{O}$ which takes a ciphertext as input and outputs the corresponding plaintext. For instance, let $\mathsf{Enc}(\cdot,\cdot)$ be an encryption scheme, and for any message $\mathsf{m}$, we write $\mathsf{c}_{\mathsf{m}} = \mathsf{Enc}(\mathsf{sk},\mathsf{m})$ to be the corresponding ciphertext. Then the adversary can upload $\mathsf{c}_{\mathsf{m}}$ to the oracle $\mathcal{O}$ and the oracle will give back $\mathsf{m}$ to the adversary.

Please design a chosen ciphertext attack for the following scheme:

- $\mathsf{Enc}(\cdot,\cdot)$ is an encryption scheme;
- there exists an efficient and publicly known algorithm $S$ such that, for any message $\mathsf{m}$

$$S(\mathsf{Enc}(\mathsf{sk},\mathsf{m})) = \mathsf{Enc}(\mathsf{sk},\mathsf{m}+1).$$

The adversary wants to corrupt a ciphertext $\mathsf{c}^*$, but it is not allowed to upload $\mathsf{c}^*$ to the oracle directly.

## 3 Kasisky Test (20 pt)

In Kasisky test, we start by looking for identical strings with length at least 3. 1) Please calculate the probability of two 3-letter strings having different plaintexts for a given Vigenere cipher. (Only need to consider 26 letters). To simplify the problem, we assume the key is a fixed word, `LUCK`. 2) Demonstrate why “m divides gcd($\Delta_1$, $\Delta_2$, )” (Slides 20, Lecture 2).

## 4 Independent Variables (12 pt)

Answer the questions in Slide 8, Lecture 3:

- What is the joint probability of $D1$ and $S1$?
- What is the joint probability of $D2$ and $S2$?
- What is the conditional probability $Pr[S1 = s|D1 = i]$ for $0 \le i \le 5$ and $0 \le s \le 10$?
- What is the conditional probability $Pr[D1 = i|S2 = s]$ for $0 \le i \le 5$ and $0 \le s \le 5$?
- Are $D1$ and $S1$ independent?
- Are $D1$ and $S2$ independent?

## 5 Perfect Secrecy (10 pt)

Perfect secrecy implies that key-length is at least as long as message length. Why?

## 6 Caesar's cipher (8 pt)

Implement the Caesar's cipher (a shift cipher where the shift is 3) using Python. Your program(s) should work on the iLab environment. You cannot use existing crypto libraries. You should submit two files: "`en.py`" and "`de.py`". Each of them takes a string as input. "**en.py**" implements the encryption algorithm and "**de.py**" implements the decryption algorithm.