Cody Murrer

11/12/2021

Cosc-472: Network Security

Dr. Lu

Project 2 Report

The algorithms that we were tasked to implement had varied in difficulty with the easiest one being the first algorithm we had to implement which was the Caesar cipher. Increasing in level difficulty with the next algorithm that had to be implemented was the Mono Alphabetical cipher. After the Caesar and the Mono Alphabetical cipher was implement the next algorithm that was to be implemented was the Poly Alphabetical cipher. Then lastly the hardest algorithm we were tasked with the implement was the Toy Tetragraph hashing algorithm.

The Caesar cipher is rudimentary encryption algorithm where you take an alphabetical dictionary then choose a numerical key zero to twenty-five and that key will serve as the number of letters to map that original alphabetical dictionary to. For example, if using a key of 3 then the letter A in the dictionary would be mapped to the letter D. The starting point of the key would be D and ABC would be wrapped to the end of the alphabet. The possible problem with this algorithm is that it is very insecure and can be easily cracked or brute forced. The way it can be cracked is by testing each different variation that can be possible with the alphabetical key as well as finding the frequency of letters occurring and work from there. I would solve this issue by possibly using multiple keys on the plaintext to encrypt it multiple times.

The second algorithm used was the Mono Alphabetical cipher which takes a phrase for the key which gets appended onto the front of the alphabet. With the key appended to the front of the alphabet the letters used in the appended key need to be removed from the original alphabet so there are no longer any more repeating occurrences of those letters. This type of encryption method is a substitution method where the dictionary gets mapped to the key. After that key is then substituted as the dictionary which is ran with the plain text to map the letters to the key encrypting the data. For example, with the test cases given with a key of zebras the letter Z would be mapped with A, E with B, and B with C and continues like that down through the rest of the alphabet. The problems with this type of algorithm are that it can be easily broken using letter frequency to determine which letter is mapped to what within the dictionary. This can be solved with using larger phrases and strings appending to the plaintext or to rewrite the whole alphabet into the key to replace it.

With the third algorithm implemented it uses a combination of the Caesar cipher and the Mono Alphabetical cipher where it uses a key that repeats itself to the length of the plaintext to substitute the alphabet with. After the key replaces the alphabet the Caesar cipher comes into play where it is used to shift the letters. For example, with a plaintext of ABCD and a key of AD it will shift the text based on their positions in the alphabet so A would be no shift and D would produce a shift of three letters to look like AECG as an encrypted output. The problems found within this algorithm are that when using long strings finding frequent letters and strings and then determine the amount of gap between the two letters to work backwards. The way to solve this is to do in a similar manner of the Mono and Caesar ciphers where multiple passes with the key and more complex keys to be used.

The hardest hashing algorithm that was implemented was the Toy Tetragraph Hashing algorithm. This is a version of the SHA hashing algorithms where it uses letters instead binary data. This function creates four by four two dimensional arrays to hold the data in which is then encrypted with a dictionary of letters mapped to numbers and numbers mapped to letters. This allows to produce two-dimensional arrays that use numbers so the use of modulus is possible. After a few passes on the plaintext, it is then summed up and mapped to the dictionary that is being mapped to numbers to produce a hashing value of four characters. This is the most secure algorithm that is being used within this assignment where the only output is a four-letter hash making it hard to determine how to decrypt it.