UC Computer Science and Software Engineering

COSC362 Data and Network Security
Semester Spring, 2021

Lab 4: CrypTool Part 2

Exercises from Lecture 5 to Lecture 9.

Background

This exercise sheet has been written for use with CrypTool v2.1.

You can download the open-source CrypTool program at `https://www.cryptool.org/en/ct2-downloads` from the Windows Virtual Machine. Please download the stable version of CrypTool v2.1. You can get more information about the CrypTool program at `https://www.cryptool.org/en/`.

Exercise 1: Simple Substitution

The aim of this exercise is as much to get used to using CrypTool as it is about implementing the Simple Substitution Cipher.

You are in the CrypTool Startcenter and follow: Cryptography → Classical → Substitution Cipher.

On the top left, there is your plaintext, on the bottom left, there is the plaintext ("source") alphabet. At the bottom centre, there is your key (labelled "destination alphabet").

QUESTION 1

(a) How does the plaintext alphabet differ from the example of the Simple Substitution Cipher that we looked at in class?

(b) In this example, how has the key been formed?

(c) If we replaced this key with a random permutation of the letters of the plaintext alphabet, how would the size of the resulting keyspace compare with the example of the Simple Substitution Cipher that we looked at in class?

The "Substitution" box that is top centre left allows you to see a pictorial representation of the encryption process.

Select Maximize (rectangular icon) on this box to see this. You can close it by then selecting Minimize (minus sign).

The "Substitution" box that is top centre right allows you to see a pictorial representation of the decryption process.

cont/...

Select Maximize (rectangular icon) on this box to see this. You can close it by then selecting Minimize (minus sign).

The ciphertext is displayed in the box on the top right of the window. The plaintext recovered by decrypting the ciphertext is displayed in the bottom right of the window. You can now modify aspects of this process by editing the appropriate window. In particular, you can modify the plaintext (top left) or the key (bottom centre). We look at several modifications of the key.

QUESTION 2

(a) In the key (destination alphabet) window, replace "bc" with "cb". You should notice that the ciphertext changes, but why does the plaintext not change?

(b) In the key (destination alphabet) window, replace "[ ]B" with "B[ ]" (there is a space between the square brackets). Look at the resulting ciphertext. Do you think this change has made it harder to conduct frequency analysis?

(c) In the key (destination alphabet) window, replace "c" with "z". Look at the plaintext that results from decrypting the ciphertext. What has gone wrong?

You can continue to play with this as long as you like.

Once you are finished, press Stop and reselect the Start Center window. It is recommended that for now you keep open the Substitution Cipher window. Later you can close it down, with or without saving.

Exercise 2: Simple Substitution using a Password

In this exercise, we look at key generation for the Simple Substitution Cipher. Even though we know this cryptosystem is not fit for modern use, there are some interesting points.

Cryptography → Classical → Substitution Cipher using a password.

The screen should look familiar, with the exception of three new additions in the bottom centre.

QUESTION 3

(a) What are the three new additions in the bottom centre?

(b) The idea behind this example is that generating a "random permutation" of the symbols of the plaintext (source) alphabet is not straightforward. This example shows how to "shortcut" this process and make it easier to implement.

Why is generating a random permutation not straightforward?

(c) Closely examine the key that has been generated (displayed in the destination alphabet box).

How was this key generated?

cont/...

Change the password to "secred".

Examine the effect that it has on the key and the ciphertext (you may want to toggle back and forth between password "secret" and "secred" to study this).

QUESTION 4

The Substitution Cipher is displaying a bad property here. What good design principle for the effect on a cryptosystem of slightly changing an encryption key would you suggest from observing this behaviour?

Change the password to "ABRAcadabra", the offset to 5, and the order to Ascending (for the latter, you will need to first select Maximize in the permutator box).

You can continue to generate some keys of your own by altering the data in the key generation boxes.

QUESTION 5

(a) What is the advantage of generating Substitution Cipher keys in this manner?

(b) How secure would you say this method of Substitution Cipher key generation was?

(c) (optional) If we restricted our plaintext and ciphertext alphabet to consist only of capital letters, used an offset of 3, fixed the order, and restricted passwords to 5-letter words where each character was different, what is the size of the keyspace that would arise from generating keys using this technique?

Exercise 3: Properties of Modern Symmetric Encryption

This exercise looks at two fundamental properties of a modern symmetric encryption algorithm.

Cryptography → Modern → Symmetric → Triple DES Cipher.

The box on the top left should contain some text that represents the plaintext and the box on the bottom left contains the key.

Before doing anything, count the number of characters in the box representing the key.

QUESTION 6

(a) Does the answer strike you as a bit strange? Consider how many bits long you are expecting a Triple DES key to be.

(b) Consider now the possibility that the characters in the box representing the key are perhaps not bits, but in fact hex characters. Does your last answer now seem more sensible?

cont/...

The ciphertext appears in the box on the top right.

Make a very small change to the input to the plaintext by editing the box on the top left. Then, observe the resulting changes to the ciphertext in the box on the top right.

QUESTION 7

What do you notice, and is this what you would expect to happen?

Stop.

We will now look at what happens when we make tiny changes to the key.

Replace the first character of the key by changing it to the letter "E". Play then Stop once the execution has finished.

Take a good look at the resulting ciphertext in the box on the top right.

Replace the first character of the key by changing it to the letter "F". Play then Stop once the execution has finished.

QUESTION 8

(a) What do you notice, and is this what you would expect to happen?

(b) How big a change to the key did you make between these two experiments?

(c) Which mode of operation and which padding scheme are being used in this example?

Change the mode of operation in both the DES Encrypt and DES Decrypt boxes (Maximise by selecting the rectangular icon).

Make a very small change to the start of the plaintext by editing the box on the top left. Then observe the resulting changes to the ciphertext in the box on the top right.

QUESTION 9

What do you notice, how did this compare with ECB mode, and is this what you would expect to happen?

Exercise 4: Stream Ciphers

It is worth seeing an example of a real stream cipher in action, so that you can appreciate its properties. Our stream cipher example is **HC-128**, developed by Hongjun Wu.

Cryptography → Modern → Symmetric → HC128 Cipher.

cont/...

QUESTION 10

(a) What is the key length of HC-128?

Hints: Look this up online, e.g.

`https://www.cryptopp.com/wiki/HC-128`

`https://www.ecrypt.eu.org/stream/p3ciphers/hc/hc128_p3.pdf`

(b) HC-128 features an initial variable (called initialisation value IV). What is its purpose?

You should discover that the ciphertext (shown in the Text Output box in the top right) is all zeroes. This would not normally happen. The example has been carefully chosen so that this does happen. It will help us with the next part of the exercise.

Change one of the hex entries in the Plaintext box (top left) so that a single bit is flipped (you could change an "E" into an "F", for example).

QUESTION 11

(a) What is the impact on the ciphertext?

(b) What kinds of application might this property be beneficial for?

Now delete one of the hex entries in the Plaintext box (top left).

QUESTION 12

(a) What is the impact on the ciphertext this time?

(b) What problem does this raise for deployment of a stream cipher?

Now change one of the hex entries in the IV box (bottom left).

QUESTION 13

(a) What is the impact on the ciphertext this time?

(b) Is this a good property for a stream cipher to have?

Exercise 5: Exhaustive Key Search

This exercise allows us to do some partial exhaustive key searches.

Cryptanalysis → Modern → DES Analysis using Entropy.

cont/...

You will see some target ciphertext in the left box. We want to find the plaintext using an exhaustive key search. A complete exhaustive key search will take us too long, but we can demonstrate how an exhaustive key search works by giving CrypTool some information about the key.

Select Settings (flower icon) in the Key Searcher box (top right). Replace all the "" symbols in the Key by "1", except for the last one on the right (leave this as a "*"). Select Presentation (left most icon) in the Key Searcher box. Play then Stop once the execution has finished.*

You should now see a list of the top ten "most likely" keys and their resulting plaintexts according to CrypTool.

QUESTION 14

(a) Did the search recover a plausible plaintext?

(b) How long did the exhaustive key search take to run?

(c) How many keys did CrypTool test during this search?

(d) (optional) How did CrypTool work out which plaintexts were more "plausible"?

(e) (optional) Can you explain why there appear to be two different keys that decrypt this ciphertext to the correct plaintext?

Now repeat this process by replacing all the "" symbols in the Key by "1", except for the last 4 on the right (leave these four as "*").*

QUESTION 15

(a) How long did the exhaustive key search take to run?

(b) How many keys did CrypTool test during this search?

Now repeat this process by replacing all the "" symbols in the Key by "1", except for the last 6 on the right (leave these six as "*").*

QUESTION 16

(a) How long did the exhaustive key search take to run?

(b) How many keys did CrypTool test during this search?

Now repeat this process by only inserting "1" in the first six positions of the key.

cont/...

QUESTION 17

When does CrypTool estimate this search is going to end?

Feel free to experiment further with this example. CrypTool will probably have something interesting to say about any searches that you attempt that are even bigger than the one we have just tried.

Exercise 6: Error Propagation of Block Ciphers

This exercise backs up the theory of error propagation with practice. We will simulate a single bit transmission error in a ciphertext in CBC mode and see what damage it causes the plaintext. Our block cipher example is **Twofish**. It was one of the five finalists of the AES (Advanced Encryption Standard) contest, but it was not selected for standardization. Twofish was slightly slower than Rijndael (the chosen algorithm for AES) for 128-bit keys, but somewhat faster for 256-bit keys.

QUESTION 18

What is the block size of the block cipher Twofish?

Hints: Look this up online, e.g.

`https://en.wikipedia.org/wiki/Twofish`

Cryptography → Modern → Symmetric → Twofish Cipher.

In the red Twofish box, change chaining mode from ECB to CBC. Play then Stop once the execution has finished.

We now want to grab the ciphertext and turn this Twofish simulator into a decryption tool rather than encryption.

Copy the hex ciphertext in the Text Output box (the furthest right box) and paste it into the Text Input box (the furthest left box), replacing the original plaintext. In the box labelled "string decoder", change the Input format from text to hexadecimal. In the Twofish box, change the action to decrypt. In the box labelled "string decoder", change the Presentation Format from hexadecimal to text. Play then Stop once execution has finished.

Did you successfully decrypt the ciphertext back to the original plaintext? If not, you must have forgotten one of the above settings, so try again!

Make a single bit change to the ciphertext in the Text Input box (any bit will do, but changing an "E" to an "F" is one way you can do this). Play then Stop once the execution has finished.

cont/...

QUESTION 19

(a) What is the impact of this change on the resulting plaintext that you can see in the Text Output box?

(b) Is this what you expected to happen?