**FALL 2021-2022 CS411-507 HOMEWORK 4**

**Elif Cemre Durgut – 26493**

You can find my codes in the folder separately. Only for the first two questions, I modified the my\_id variable in the given client py files as it is written in the file as a comment.

**Question 1)**

Since we can test as many ciphertexts as we want, this is a chosen ciphertext attack. I found the following attack method on the Internet. It works because. So, the answer is

Bravo! You find it. Your secret code is 39851

A screenshot of a computer

Description automatically generated with medium confidence

**Question 2)**

We are given that the PIN is a 4-digit number. So, we can apply brute force and try all the possible PINs and check if it is true or not since it is a short range [1000,9999]. I checked the message coming from the server to see if the PIN is correct or not.

The correct PIN is 4981.

**Question 3)**

As we saw in the lecture, t = (beta\*\*k) \* m (mod p). For this question, beta is given as h.

We are already given t, h, and m. We do not know k and m. We can again apply brute-force on the k value. It should be in the range of [0, 2\*\*16-1] as it is calculated in the Enc function in ElGamal.py.

K value is 64278. After finding the k value, the only unknown is m (message). If we leave m alone in the above formula, m = t \* ((h\*\*k)\*\*-1) mod p.

The answer is I am gonna make him an offer he cannot refuse

Graphical user interface, text, application

Description automatically generatedA screenshot of a computer

Description automatically generated

**Question 4)**

I realized that r1 is equal to r2. So we can use what we’ve learned in the lecture as I added below. We first convert m1 to integer from byte and then apply the formula using the modinv function.

The answer is Well, it was more like a command, no was not an option!

Graphical user interface, application, Word

Description automatically generated

**Question 5)**

Again, I realized that r1 = r2. So, we can apply what we learned in the lecture with the help of modinv function. The answer is 18011493590957919843196654272530256451916130571913898417508651137437

To make sure I found the correct key, I double-checked it using this equality: beta = g\*\*a mod p. The given beta value and the value that I found match so it is correct.

Graphical user interface, text

Description automatically generated

**Question 6)**

According to hint, if we ran out of random keys, that means we either used the same key (like the previous question) or a multiple of a previously used key. r1 is not equal to r2 so the first option is not true. I first tried over the range of [0, 50) but it did not work so I increased the range and I found the x as 63. Then, I applied what we saw in the lecture. The answer is 66568624500090235129890566130399211243633217014

Graphical user interface, text, application

Description automatically generated