MSIN-4212 Deciphering Bitcoin

Cryptography Assignment

Theoretical Exercise

Daniel Felipe Carrillo Vanegas – Cod. 201215750
Javier Peniche – Cod. 201716772

JUNE 2022

Question 1: The message below was encrypted using an old encryption algorithm. Use frequency analysis to recover the original message.

Encrypted text:
hjlnjzdrnuqrmjupxodnacnzdnjcjungcanvxuunpxmnejurnwcnzdnbnjmernacnzdnujvdnacnwx
cardwoxmnbdermjlxwbdvdnacncdexjcxmxnuvdwmxnwyxlxodnnunbyjwcjsxhnulxlxmnuvd
wmxnwcjulxhdwcdajzdnjlanmrcxbdjenwcdajvxaraldnamxhereauxlx

Analysis

For the resolution of the exercise, the free software tool CrypTool was used. With the tool the analysis option was used for ciphertexts with the Cesar cipher (which meets the characteristic of being an old encryption algorithm). There, an overlap analysis was performed considering two files, cipherTest.txt, which contains the encrypted file, and genesis-es.txt, which has the percentages of the appearance of the different letters in the Spanish language.

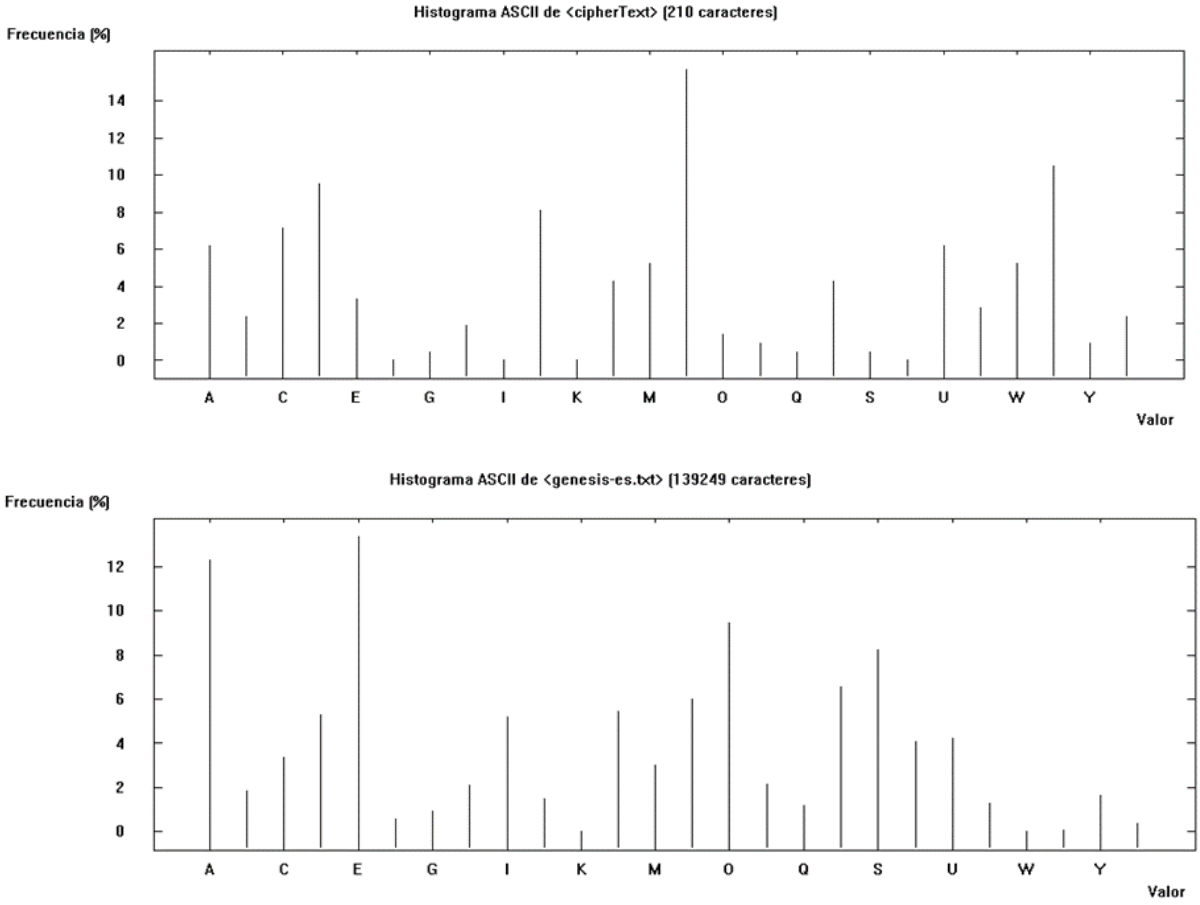
Since key offsetting was used during decryption, there were two possible keys: For a key offset of 1 the key is "I", for a key offset of 0 the key is "J". After the text with both keys had been decrypted, the only key that resulted in a text with intelligible words in Spanish was the key "I". It is so then, that the decrypted text was obtained.

To obtain the replacement map, a program was created where the ciphertext and the decrypted text were compared position by position, assigning in a HashMap as a key the character of the ciphertext, and as a value the character of the corresponding decrypted text.

```
public class CompareCharByPosition {

   private static int min = 97, max = 123;

   static Map<String,String> hashM= new HashMap<>();

   static String
cipherText="hjlnjzdrnuqrmjupxodnacnzdnjcjungcanvxuunpxmnejurnwcnzdnbnjmernacnzdnujvdnacnwxcardwox
mnbdermjlxwbdvdnacncdexjcxmxnuvdwmxnwyxlxodnnunbyjwcjsxhnulxlxmnuvdwmxnwcjulxhdwcdajzdnjlanmr
cxbdjenwcdajvxaraldnamxhereauxlx";

   static String
decryptedText="yaceaquielhidalgofuertequeatalextremollegodevalientequeseadviertequelamuertenotriunfodes
uvidaconsumuertetuvoatodoelmundoenpocofueelespantajoyelcocodelmundoentalcoyunturaqueacreditosuaven
turamorircuerdoyvivrloco";

   public static void main(String[] args) {

      compareCharsAt(cipherText, decryptedText);

      for (int i = min; i < max; i++) {
         char ch = (char) i;
         String str= String.valueOf(ch);
         System.out.println((char) i + "->" + hashM.get(str));
      }
   }


   public static void compareCharsAt(String a, String b) {

      for (int i = 0; i < a.length(); i++) {
         hashM.put(String.valueOf(a.charAt(i)) , String.valueOf(b.charAt(i)));
      }
   }
}
```

Ciphertext:

Hjlnjzdrnuqrmjupxodnacnzdnjcjungcanvxuunpxmnejurnwcnzdnbnjmernacnzdnujvdnacnwxcardwo
xmnbdermjlxwbdvdnacncdexjcxmxnuvdwmxnwyxlxodnnunbyjwcjsxhnulxlxmnuvdwmxnwcjulxhdw
cdajzdnjlanmrcxbdjenwcdajvxaraldnamxhereauxlx



Decrypted text:
Yaceaquielhidalgofuertequeatalextremollegodevalientequeseadviertequelamuertenotriunfodesuvidaconsumuertetuvoatodoelmundoenpocofueelespantajoyelcocodelmundoentalcoyunturaqueacreditosuaventuramorircuerdoyvivrloco

Original Text (suggested):
"Yace aquí el hidalgo fuerte que a tal extremo llego de valiente que se advierte que la muerte no triunfo de su vida con su muerte tuvo a todo el mundo en poco fue el espantajo y el coco del mundo en tal coyuntura que a crédito su aventura morir cuerdo y vivir loco".

Reference found on the internet:
"Yace aquí el hidalgo fuerte que a tanto extremo llegó de valiente, que se advierte que la muerte no triunfó de su vida con su muerte. Tuvo a todo el mundo en poco, fue el espantajo y el coco del mundo, en tal coyuntura, que acreditó su ventura morir cuerdo y vivir loco"

Se reconoce entonces que el texto hace parte del libro Don Quijote de la Mancha.

Substitution map:

Ciphertext character-> Plaintext character

a->r

b->s

c->t

d->u

e->v

f->null

g->x

h->y

i->null

j->a

k->null

l->c

m->d

n->e

o->f

p->g

q->h

r->i

s->j

t->null

u->l

v->m

w->n

x->o

y->p

z->q

Question 2: Suppose a user wants to store a file on a hard disk with the goal of keeping information confidential across time. Describe a scheme (step by step) to store the file, guaranteeing that only the owner of the file can read the information and validate that it has not been tampered with.

Solution:

Step 1

The user generates a key (with the symmetric encryption scheme), which is only known to him.

Step 2

The user generates a cryptographic Hash code with the file, H(M). This in order to ensure the integrity of the message.

Step 3

The user saves the encrypted file (M) with his key. Represented by C(K, M). This in order to ensure the confidentiality of the message.

Step 4

The user stores separately H(M) and C(K, M)

Question 3: Suppose Alice sends to Beto the following message:

$$C(K\_B+, C\ (K\_A-, K)) \mid C(K, M) \mid C(K\_A-, H(M))$$

Where:
Alice has her public and private keys: K_A+, K_A-
Beto has his public and private keys: K_B+, K_B-
K is a symmetric key
| means concatenated

Describe the procedure (step by step) that Beto must follow to recover information, identify what security guarantees we have (confidentiality, integrity, authentication, non-repudiation) and justify your answer.

Solution

Step 1

Beto, with his private key deciphers C(K_B+, C (K_A-, K)), leaving C (K_A-, K)

Attribute that is guaranteed:

Confidentiality: Since only Beto can decrypt the message with his private key (K_B-).

Step 2

Beto, with Alice's public key, deciphers C (K_A-, K), resulting in K, thus obtaining the symmetrical key.

Attributes that are guaranteed:

Authentication and non-repudiation: Since the message was encrypted with Alice's private key, Beto can be sure that Alice was the one who sent the message.

Step 3

Beto, with the symmetric key (K), can decipher C(K, M). Thus, obtaining the message M

Step 4

Beto, with the public key of can decipher C(K_A-, H(M)), resulting in H(M). Beto can then recalculate the H(M) itself and thus verify that the message was not altered.

Attributes that are guaranteed:

Integrity: Since Beto can recalculate the Hash, he can make sure that the message has not been altered.

Authentication: Since the message was encrypted with Alice's private key, Beto can be sure that Alice was the one who sent the message.