The Space Attack challenge report

June 26, 2022

1 Introduction

The challenge was about deciphering multiple messages with various encoding/ciphering methods of increasing difficulty.

We used the Python programming language to help us decipher the messages, and to check the ideas we had.

2 Common data

The first objective we completed was about the decryption of the common data, two base64 encoded strings encrypted with AES128 in ECB mode, with the encryption key: '`aesEncryptionKey`'.

The first part was `JaAbDk1QlerxhNo8pLqS2Q==`, that after decryption resulted in the string `start`.
The last part was `nij8GNMQUx06N++TLehaxw==`, that after decryption resulted in the string `stop`.

In the next sections the messages are considered without this common data.

3 General approach

To decipher the hardest messages, we followed two main steps aimed at identifying the cipher used:

- Frequency analysis, to distinguish between monoalphabetic and polyalphabetic
- Autocorrelation, to identify if the cipher involved some transposition of the characters

4 Message 1

The first message was fairly easy, as it was only a base64 encoded message.

Original: `d2VsY29tZSBldmVyeW9uZSB0byBjeWJlciwgdGhpcyBpcyB0aGUgZmlyc3Qgc3RlcCB0byBsZWFkIHlvdXIgdGVhbSB0byB3aW4=`
Decrypted: welcome everyone to cyber, this is the first step to lead your team to win

5 Message 2

The second message was a bit more complicated, we originally thought about some kind of Casear ciphering, but no english word matched the final word with an apostrophe. Then we had the idea to read it backwards and it worked!

Original: `tlovrednu suB B niam a dah ev'ew,melborp a dah ev'ew`
Decrypted: we've had a problem,we've had a main B Bus undervolt

6 Message 3

The third message was one of the hardest ones. We started with an histogram of the letters (Figure 1)

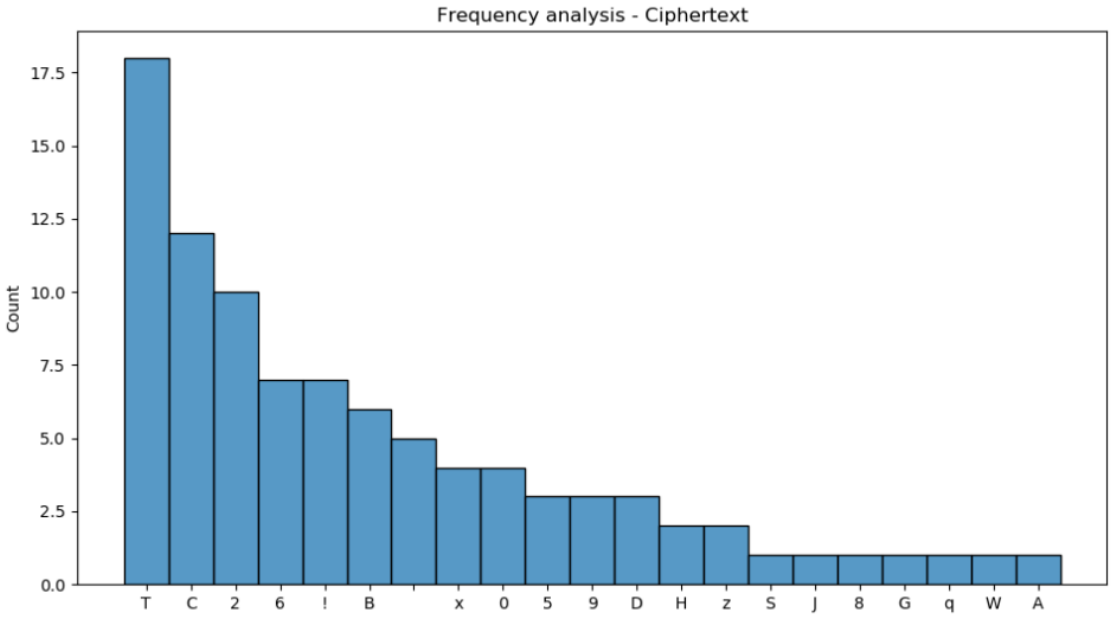

Figure 1: The histogram of letters in message 3

and we compared it with the histogram of a reference english sentence (Figure 3).

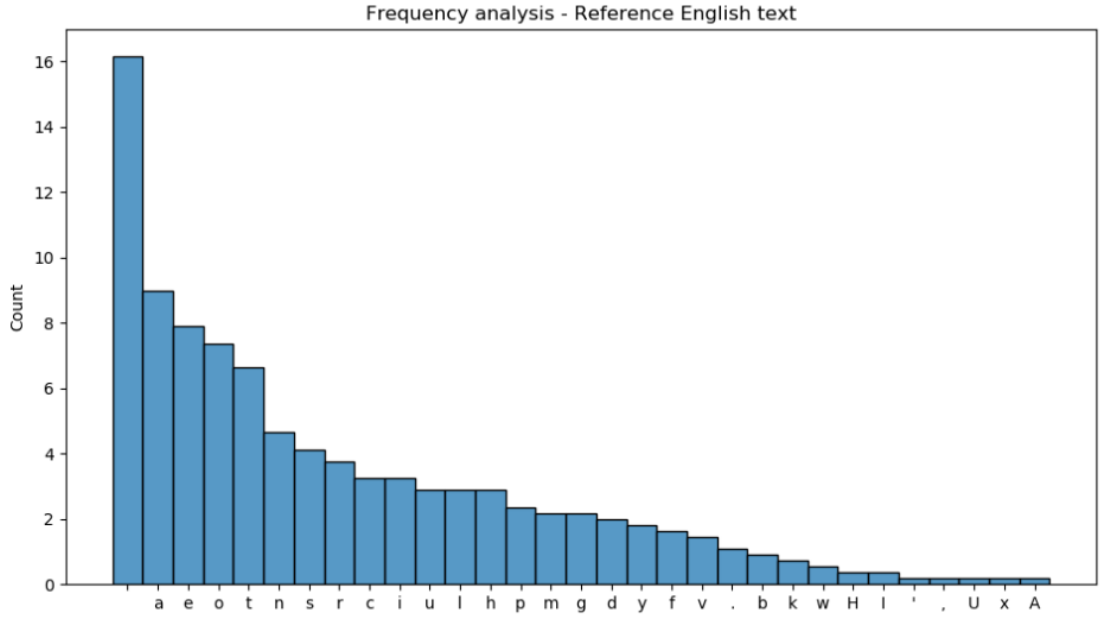

Figure 2: The histogram of letters of a reference english sentence

From this result, we excluded the possibility of a polyalphabetic cipher, since the message frequencies matched well the frequencies of the english sentence, even if the characters themselves didn't match. From that result, we tried some kind of shift-based ciphering using the ASCII encoding, but without success. After some trial and error, we came up with a partial mapping of the characters, apparently without a logic. But we had to find some kind of pattern to decode the two digit characters of the key, since they didn't appear anywhere in the text. The cipher we finally found was a simple rotational cipher (with a rotation of +43) on the alphabet:

```
ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz1234567890 !.
```

The '#' character was an unknown character that was needed to obtain the correct alignment of all the known letters.

Original: `C52TB!9DC6! T6BTC!TDB2Tx T6B9TC!Tz!00D 6zxC2TC29202CAHTC!TC52T0x6 TBCxC6! WTq52T 2GCT82HT6BTJS`
Decrypted: the solution is to use an isl to communicate telemetry to the main station. The next key is 10

7 Message 4

This message was a bit simpler in our opinion, since the numbers from 1 to 8 helped us a lot to find the correct way to read the message. It seemed a lot like an anagram of an English phrase, since the letters frequency matched well the typical English letters, and the autocorrelation of the message was low, we opted for a transposition ciphering.
Aligning the numbers from 1 to 8 on the same column allowed us to correctly read a part of the message, and putting them in the correct column was the key point to decrypt the full message.

Eventually we found that the ciphering algorithm is a columnar transposition cipher, using the hint from with a column size of 10.

T	h	e		p	r	i	o	r	i
t	y		i	s		t	o		d
o	w	n	l	o	a	d		a	l
l		t	h	e		i	m	a	g
e	s		o	f		N	a	p	l
e	s		t	h	a	t		c	o
n	t	a	i	n		s	h	i	p
s	.		T	h	e		l	a	s
t		k	e	y		A	E	S	
i	s		p	a	s	s	w	o	r
d	1	2	3	4	5	6	7	8	

Figure 3: The table used to decipher the message 4

Original: `Ttoleenstidhyw sst. s1e nt a k 2 ilhotiTep3psoefhnhya4r a a e s5itdiNts As6oo ma hlEw7r aapciaSo8idlglops r`
Decrypted: The priority is to download all the images of Naples that contain ships. The last key AES is password12345678

8 Message 5

After decrypting message 4, the message 5 was the easiest to solve, since it used the same encryption algorithm as the first common data.

Original: `gCOQ6/UzLUyeLlGHRUYLqUlnGr32YknbLdVHDW5adg4AIfqpGYNTW7z78bcqRy2q+60ZwpJC2aGA3ya2XY66cw==`
Decrypted: Congratulations! This is the last level of encryption