ECE599 - Assignment Homework 4

Opeyemi Ajibuwa

Question 1

a. The primes for when RSA-CRT are used (Bellcore) are given as follows.

p = f9555b790d60dcb3fdcdf464b88ab7bb629bfce037f4154927df19fcdb1b4c7327d41b17d848455cffbda7e8080c08600be3af126df6c481ab25da70bec471c0fb
q = 9f44ddf28f05904455669a629df988adf203812f56aa8047c7db9bb7b4e61dd67b027e80d8700a77471943cc76370ced07056ef808a12b2a467c159e586c33

b. The primes for when RSA-CRT are used (Lenstra) are given as follows.

p = d4693216ca3210f1491477d556e709141f6b5ea57e8b64a51011190d607b6b92a601857e4ad26e2b45123804ebdd08ccd15b0e50edcdc8754d5b2bb99dc8286087
q = eacd987fce4c2815b8e1f6557a4120cd822763baa732e6fbd2d35d61b85f8278263ce068cddf6099ba885cda0b4ed1c2374de5d34b265fec3358611905ae81
d =
6f7345e591342f162230a8b392814b0f4f80268e7008b129cf0c4c009ad4cce91e6a3f1d2a5eb72ed86e55b079a8a2963248640819b1121f0411d0ba1b1647445bf2438288738d9ecaef90ed
7b3a4d1170f22f28cc60396854b9508df0cc39397bbc4eae35428edb25416b6370bda32bc8d58ac6fceef0b3d94a0d65c7ef1501

Question 2

a. How many glitches are successful? Does that seem like a high success rate?

The number of successful glitches in the data is 402. A success rate of 402/10000 = 4.02% is obtained. If we consider the difficulty of the attack and the complexity involved, this seems like a high success rate. That means that on average, every 25th glitch of AES-128 is successful.

b. How many total pairs will we need? By parsing the provided files, find enough pairs to complete the attack and output them here in hex format. Can anything happen that might cause you to need more pairs?

For the full attack, I suppose I wouldn't need more than 8 pairs total to extract all the 16 key bytes since my D include the possibilities that any of the 4 bytes in the column can be glitched. However, if I assume that location of the glitch happens strictly in the first row, then I may have to look more many more pairs to be able to correctly extract the four key bytes corresponding to each column.

The 8 pairs I used to fully extract all 16 key bytes are output below:

```
---ciphertext/ftext pairs (in hex) to extract the first 4 bytes of the 10th round key corresponding to
the first column are---
Ctext: ['0x23', '0xdb', '0x99', '0xde', '0xfc', '0x61', '0x74', '0x7b', '0x8c', '0x5f', '0x36', '0x65',
'0x1b', '0x7f', '0x26', '0x1e']
 Ftext: ['0x33', '0xdb', '0x99', '0xde', '0xfc', '0x61', '0x74', '0xf6', '0x8c', '0x5f', '0xa3',
'0x65', '0x1b', '0x2e', '0x26', '0x1e']

Ctext: ['0xe5', '0x99', '0x2a', '0xab', '0xc2', '0xd1', '0x22', '0xa6', '0xde', '0x2', '0xb', '0xcd',
'0x53', '0x7e', '0xad', '0xf']
 Ftext: ['0x95', '0x99', '0x2a', '0xab', '0xc2', '0xd1', '0x22', '0x9a', '0xde', '0x2', '0x22', '0xcd',
'0x53', '0x56', '0xad', '0xf']

---ciphertext/ftext pairs (in hex) to extract the next 4 bytes of the 10th round key corresponding to
the second column are---
Ctext: ['0x72', '0x90', '0x2', '0x6', '0x1e', '0x13', '0x83', '0xe8', '0xfc', '0xa6', '0x51', '0x42',
'0xe9', '0x2', '0xca', '0xb4']
 Ftext: ['0x72', '0xd7', '0x2', '0x6', '0x44', '0x13', '0x83', '0xe8', '0xfc', '0xa6', '0x51', '0xe9',
'0xe9', '0x2', '0x43', '0xb4']

Ctext: ['0x5c', '0xc1', '0xb5', '0x61', '0xe7', '0xb', '0xd5', '0x38', '0xb1', '0x76', '0x1b', '0xed',
'0x86', '0xb4', '0xdc', '0x26']
 Ftext: ['0x5c', '0x89', '0xb5', '0x61', '0xf0', '0xb', '0xd5', '0x38', '0xb1', '0x76', '0x1b', '0xf1',
'0x86', '0xb4', '0x91', '0x26']

---ciphertext/ftext pairs (in hex) to extract the next 4 bytes of the 10th round key corresponding to
the third column are---
Ctext: ['0xbb', '0x4f', '0x38', '0xf6', '0xca', '0xe5', '0xb6', '0x3b', '0xb1', '0x30', '0xe8', '0xdb',
'0x5d', '0xb0', '0x63', '0x42']
 Ftext: ['0xbb', '0x4f', '0xca', '0xf6', '0xca', '0xdd', '0xb6', '0x3b', '0x75', '0x30', '0xe8',
'0xdb', '0x5d', '0xb0', '0x63', '0x55']

Ctext: ['0xec', '0xb4', '0x9e', '0x91', '0x4c', '0xac', '0x9e', '0x83', '0xf1', '0xe1', '0xf9', '0x5b',
'0xe0', '0x87', '0x4d', '0x10']
 Ftext: ['0xec', '0xb4', '0xd4', '0x91', '0x4c', '0x4c', '0x9e', '0x83', '0xed', '0xe1', '0xf9',
'0x5b', '0xe0', '0x87', '0x4d', '0x48']

---ciphertext/ftext pairs (in hex) to extract the next 4 bytes of the 10th round key corresponding to
the fourth column are---
Ctext: ['0xb6', '0xdc', '0x67', '0xdb', '0xc6', '0xf7', '0x63', '0x2c', '0x1a', '0x16', '0xe', '0xa4',
'0xc7', '0x44', '0x85', '0xc7']
 Ftext: ['0xb6', '0xdc', '0x67', '0x15', '0xc6', '0xf7', '0x60', '0x2c', '0x1a', '0x24', '0xe', '0xa4',
'0xaa', '0x44', '0x85', '0xc7']

Ctext: ['0x3b', '0xcd', '0xd2', '0xb4', '0xdc', '0x96', '0x67', '0x6a', '0xfb', '0xd8', '0x71', '0x8d',
'0x4e', '0xa9', '0xbf', '0xd']
 Ftext: ['0x3b', '0xcd', '0xd2', '0x19', '0xdc', '0x96', '0x19', '0x6a', '0xfb', '0xae', '0x71',
'0x8d', '0x71', '0xa9', '0xbf', '0xd']
```

c. **The four extracted keys for the simple attack.**
The key bytes for the simple attack are: [**168**, **138**, **164**, **45**] corresponding to **[k0, k13, k10, k7]**

d. **Provide the entire round 10 keys extracted with the full attack**
All keys correctly shifted according to the state are: [**168**, **73**, **55**, **172**, **53**, **213**, **50**, **45**, **93**, **35**, **164**, **0**, **170**, **138**, **46**, **198**]

e. **Recover the original (first round) key and use it to decrypt the following (hex encoded) secret message: 2a92fc6ad8006b658f49062c2843ad99**
The secret message correctly decrypted is DFAIsAFunAttack!