CS641A: Assignment 5

Rijndael
Abhishek Jain, Arham Chopra, Ayush Tulsyan
13027, 14130, 14807167

1 The Fall

You are in a passage. There is very little light here. You light a matchstick... the floor is full of rocks. A cold draught hits you from behind and you shiver. You try to walk quickly but stumble. Cursing your luck, you slow down. The passage curves downwards. You put one hand on a side wall for guidance. The downward slope becomes steeper. You walk very slowly to avoid falling down. The slope has become very steep now... You throw away the matchstick to hold both side walls of the passage with your hands. It becomes completely dark! Your foot hits a small rock that rolls down the passage. Stopping, you try to hear the sound of the rock hitting the bottom and after a while you hear a very distant splashing sound. The passage seems to be leading to some deep underground well!

>*go*

You bravely decide to go ahead. With slow and deliberat esteps, you go down the slope. Suddenly, as you put down afoot on the floor, your foot finds thin air! In almost slow motion, you fall forward. You throw your hands about to holda rock but miss everything and suddenly you are in a free fall!! Desperate, you try to grab something, but there is nothing to grab ...

>*wave*

Magically, your fall downwards is arrested! Almost in slow motion, you go downwards till you hit water. It is very cold! You swim in one direction in the hope of finding dry land, but only find vertical rocky wall. Swimming in opposite direction yields the same result!

>*dive*

Diving under the water, you immediately notice mild yellowish light coming out of what appears like a hole in a wall under the water. Swimming quickly, you go through the hole. Now swimming upwards, you quickly hit the surface, and find yourself in a kind-of pool. The pool is circular in shape. You swim to one end of it, and climb out of water... After a while, you look around the place. The pool is a perfect circle! It cannot be naturally formed, you conclude. Then who could have constructed a pool in such a place? There are a number of lighted torches all around, providing ample yellow light. Someone seems to have fitted these torches and lighted them! Another unanswered fact... A well-lit passage is carved in the wall, and seems to be leading somewhere deep inside...

>*go*

You walk down the passage. After quite a long walk, the passage ends in a small door. Entering, you stand transfixed! You are inside a large hall! Someone has constructed this deep inside the mountains!! The floor and the walls are made of marble, there are chandeliers hanging from the ceiling with burning torches inside. You rub your eyes in disbelief. But the scene stays. Recovering a little from the initial shock, you look around the hall. It has two exits, one where you entered from and the other that is closed. It has a glass panel next to it and it is a proper door, made of carved wood... You go around the hall, exploring its details. The marble is is very good quality. (How did someone get all this marble here, you wonder?) It feels cold. The chandeliers are made of crystal glass, so the light is reflecting in beautiful patterns. Who designed all this?

>read

You come up to the closed door and look at the screen ...... there is nothing written on it as earlier. As you wonder if the spirit is around to help you out, you hear it whispering in your ears ... "This is another magical screen. And this one I remember perfectly... Consider a block of size 8 bytes as 8 x 1 vector over F_128 – constructed using the degree 7 irreducible polynomial x^7 + x + 1 over F_2. Define two transformations: first a linear transformation given by invertible 8 x 8 key matrix A with elements from F_128 and second an exponentiation given by 8 x 1 vector E whose elements are numbers between 1 and 126. E is applied on a block by taking the ith element of the block and raising it to the power given by ith element in E. Apply these transformations in the sequence EAEAE on the input block to obtain the output block. Both E and A are part of the key. You can see the coded password by simply whispering 'password' near the screen..."

2 AES Attack

Encoding

Our first task was to figure out the encoding which was being used to convert the string (that we are using as input) into binary blocks. The first guess could obviously be the ASCII values. So we started analyzing the output. We observed that all the outputs ranged from f-u, so it was intuitive that the encoding has something to do with hex base system as f-u consists of 16 characters. We analyzed all the pairs from ff-mu, and we saw that all the pairs encrypted to unique strings. This meant that our assumption of something to do with the ASCII value is somewhat true, as ff-mu comprised of 128 different characters and ff-mu indeed matching those 128 different characters. It was also observed that the input such as **"af"** and **"a"** gave the same output, hence the character **"f"** might have been used as a padding. So we mapped **ff** to **00** in incremental fashion all the way up to **mu** which was mapped to **7f** in hex base (or 128 in decimal) system.

Observation

We had the input data in the ciphertexts.txt file, where we observed that if we changed the i^{th} pair bit of the data, the output from corresponding i^{th} pair bit onwards started changing. This gave us the hint that the matrix is of the *lower triangular* form. We also had the constraint that the ASCII values of the elements ranged from 0 to 127 only.

Cracking the encryption boxes

With the above observation in hand, it becomes fairly simple to crack the Exponential transformation box E and diagonal elements of the Linear Transform A.

We denote the elements of A by $a_{i,j}$, i being the row of element and j being the column. Also, e_i denotes the i^{th} element of vector E

$$\begin{bmatrix} a_{0,0} & 0 & 0 & \dots & 0 \\ a_{1,0} & a_{1,1} & 0 & \dots & 0 \\ a_{2,0} & a_{2,1} & a_{2,2} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{7,0} & a_{7,1} & a_{7,2} & \dots & a_{7,7} \end{bmatrix}$$

We begin by fetching cipher-texts for the plain-texts of the form where only one block is non-zero. For every choice of block, there are 127 possible plain-texts 1 to 127. Using these 127 plain-text cipher-text pairs, one can iterate over the choice of diagonal elements and the exponentiation corresponding to the non-zero block.

One basic observation is, given the linear transform is lower triangular matrix, any block of output, let's say i^{th} block, is dependent on j^{th} block of input iff $i \geq j$.

In our chosen pattern for input, if x is the value of non-zero input block(say i), then the corresponding block of output has the value $(a_{i,i}(a_{i,i} * x^{e_i})^{e_i})^{e_i}$. Iterating over values of $a_{i,i}$ (from 0 to 127) and e_i (from 1 to 126), one gets 3 tuples of values for each block.

Block number	Pairs for $a_{i,i}$ and e_i
0	[(11, 110), (34, 125), (82, 13)]
1	[(8, 121), (25, 14), (94, 76)]
2	[(72, 118), (84, 80), (98, 32)]
3	[(15, 10), (31, 57), (81, 38)]
4	[(40, 12), (89, 86), (125, 5)]
5	[(38, 35), (87, 65)]
6	[(8, 59), (25, 92), (94, 81)]
7	[(53, 89), (83, 77), (118, 81)]

Now, these above values have to be used to get the other elements of matrix and eliminate some pairs among these.

Any element $a_{i,j}$ can be obtained by looking at the i^{th} block of output when the j^{th} block of input is non-zero. To calculate this, besides the diagonal elements, some other elements of A are also needed. To discover $a_{i,j}$, all elements in the following set $S_{i,j}$ should be known

$$S_{i,j} = \{a_{n,m} | n > m, j \leq n, m \leq i\} \cap \{a_{n,n} | j \leq n \leq i\}$$

To visualize, these elements form a right angled triangle, with corners $\{a_{j,j}, a_{i,j}, a_{i,i}\}$

Search for non-diagonal is executed iteratively. First on the elements of the form $a_{i+1,i}$, followed by $a_{i+2,i}$ and so on. While searching for the $a_{i+1,i}$, we could eliminate the extra pairs from the above matrix.

Overall, it took us 127*8 chosen plain-texts to execute the attack. We haven't explored the possibility of doing so with lesser inputs to encryption machine, but this seems plausible. The number of operations required are 128*128*36 ($\approx 2^{19}$) (36 non-zero elements in matrix and maximum number of operations required to find any element is 128*128)

For implementing, alternate Multiply and Exponentiate functions have been used. The addition in F_{128} is similar to XOR operation in integers. The above two were modified accordingly and the values were stored into 128*128 matrices. Using these we then defined our Encrypt function, this was used during the brute force attack on each of the $a_{i,j}, e_k$ to check whether out encrypted output matched with actual encrypted output.

$$A^T = \begin{bmatrix} 3 & 118 & 82 & 92 & 126 & 38 & 19 & 2 \\ 0 & 121 & 10 & 118 & 32 & 103 & 54 & 66 \\ 0 & 0 & 118 & 99 & 22 & 36 & 125 & 33 \\ 0 & 0 & 0 & 57 & 85 & 11 & 3 & 62 \\ 0 & 0 & 0 & 0 & 86 & 59 & 31 & 116 \\ 0 & 0 & 0 & 0 & 0 & 35 & 100 & 61 \\ 0 & 0 & 0 & 0 & 0 & 0 & 81 & 34 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 77 \end{bmatrix}$$

and

$$E = [82, 8, 72, 31, 89, 38, 94, 83]$$

This was also used to decrypt the actual password. To find the actual password we did a similar thing, we assumed the input blocks to be unknown. Then we brute forces each of the individuals blocks from left to right. This is because the ith leftmost block of the output only depend on the i leftmost blocks of the input.

The plain-text password is: `mvcpldwwlmjabzfm`

Reason for Lower Triangular Matrix

The terminology followed here is :

- $p_0, p_1, p_2, p_3, p_4, p_5, p_6, p_7$ is the input 8 byte plaintext.
- $c_0, c_1, c_2, c_3, c_4, c_5, c_6, c_7$ is the output 8 byte ciphertext.
- 0 represents 'ff'
- The linear transform matrix is represented as

$$\begin{bmatrix} a_{0,0} & a_{0,1} & a_{0,2} & \cdots & a_{0,7} \\ a_{1,0} & a_{1,1} & a_{1,2} & \cdots & a_{1,7} \\ a_{2,0} & a_{2,1} & a_{2,2} & \cdots & a_{2,7} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{7,0} & a_{7,1} & a_{7,2} & \cdots & a_{7,7} \end{bmatrix}$$

By passing input where p_k is varied keeping other $p_i = 0$, we observed that the ciphertexts have the form $c_i = 0$ for $i < k$.

If the input to the last A layer is the vector $[x_0 x_1 x_2 \dots x_7]$, the output will be a vector will be

$$\begin{bmatrix} y_0 = a_{0,0} * x_0 + a_{0,1} * x_1 + a_{0,2} * x_2 + \cdots + a_{0,7} * x_7 \\ y_1 = a_{1,0} * x_0 + a_{1,1} * x_1 + a_{1,2} * x_2 + \cdots + a_{1,7} * x_7 \\ \vdots \\ y_7 = a_{7,0} * x_0 + a_{7,1} * x_1 + a_{7,2} * x_2 + \cdots + a_{7,7} * x_7 \end{bmatrix}$$

Since the last E layer is just a bijective mapping any unique output corresponds to a unique input. Also since the operation is exponentiation, 0 at output will map to 0 at the input.

On passing input as $00 \dots p_k p_{k+1} \dots p_7$ we obtained output as $00 \dots c_k c_{k+1} \dots c_7$. In other words, whenever the input plaintext has $p_0, .., p_i = 0$ the output will have $c_0, .., c_i = 0$. This implies that input to the last E layer will also have been of the same format $00 \dots c'_k c'_{k+1} \dots c'_7$. Thus the equation for $y_i = 0$ had 2^{49-7*i} solutions. This is only possible if all but i of $a_{i,j}$ is nonzero and the rest are 0. Thus the first row will have 7 0 elements, 2nd row 6 0 elements, 3rd 5 o elements and so on.

We also observed that passing $p_0 p_1 \ldots p_k p_{k+1} \ldots p_7$ and $p_0 p_1 \ldots p_k p'_{k+1} \ldots p_7$, the output only changed after k, this implies that all the 0 present in each row have to be at the end of the row. Thus we get a lower triangular matrix.

References

[1] Biryukov, Alex, and Adi Shamir. "Structural cryptanalysis of SASAS." International Conference on the Theory and Applications of Cryptographic Techniques. Springer, Berlin, Heidelberg, 2001.