Fall 2026 CIS 3362 Homework #4: DES, AES
Due: Check WebCourses for the due date.

1) (10 pts) Show the result of applying the IP to the following input block represented in hexadecimal: 6A3B 05C9 D841 FE27. Express the final result in HEX. Show your 64 bits written out in order in your work and indicate the output 64 bits and their conversion to HEX.

Solution

For this question, I listed the hexadecimal digits as binary, and created an 8x8 square of bits (2 hexadecimal digits per row * 8 rows). Then, to create each individual row, I read the columns, from bottom to top, even columns first, then odd columns, creating the order 2, 4, 6, 8, 1, 3, 5, 7. The 2nd column read bottom to top becomes the first row, the 4th column read bottom to top becomes the 2nd row, so on and so forth until you've completed the transposition.

```
0110 1010 -> 0111 1001
0011 1011 -> 0101 0010
0000 0101 -> 1100 0100
1100 1001 -> 1010 1110
1101 1000 -> 0101 1000
0100 0001 -> 1100 0011
1111 1110 -> 0101 1011
0010 0111 -> 1100 0011
```

79 52 C4 AE 58 C3 5B C3

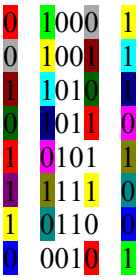
2) (10 pts) The input to the expansion matrix E in DES, expressed in hexadecimal is 89AB5F62. What is the output? Please express your answer in **hexadecimal** and put a little space between each group of **2** hex characters.

Solution

89AB5F62 converted to binary:

```
1000
1001
1010
1011
0101
1111
0110
0010
```

Following the Expansion permutation matrix, we list the bits 4x8, one hex digit per row. We add 2 bits to each row corresponding with the matrix, to find the left bit to add for each row, you take the rightmost input bit from the row above, taking the bottom right input bit when on the first row. To find the bit you add to the right side of each row for each, you take the leftmost input bit from the row below, taking the top left bit when on the bottom row.



As you can see, the inner bits are the input bits, and I have color coded which added bits come from which original input bits

```
0100 0101 -> 45
0011 1101 -> 3D
0101 0110 -> 56
1010 1111 -> AF
1110 1011 -> EB
0000 0101 -> 05
```

45 3D 56 AF EB 05

3) (10 pts) Consider a portion of a single DES round where the input (expressed in HEX) to S boxes S_1, S_2, S_3, S_4 S_5, S_6, S_7, and S_8 is 2345 9876 CDEF. What are the 32 bits of output from the S-boxes? Express your result in **binary (so 32 separate bits)**. (Note: Partial credit on this question will be limited for obvious reasons, so double check your answers.)

Solution

First, convert to binary:
0010 0011 0100 0101 1001 1000 0111 0110 1100 1101 1110 1111

Change to sets of 6 for input into S boxes:
001000 110100 010110 011000 011101 101100 110111 101111

Each of our 6 digit long binary numbers will be input into our S boxes as follows, the outer area, the blue, will be the row of the s box, while the inner number is the column.

```
S1 Row 0 Col  4 ->  2 -> 0010
S2 Row 2 Col 10 -> 12 -> 1100
S3 Row 0 Col 11 ->  7 -> 0111
S4 Row 0 Col 12 -> 11 -> 1011
S5 Row 1 Col 14 ->  8 -> 1000
S6 Row 2 Col  6 -> 12 -> 1100
S7 Row 3 Col 11 -> 15 -> 1111
S8 Row 3 Col  7 -> 13 -> 1101
```

0010 1100 0111 1011 1000 1100 1111 1101

4) (20 pts) Use the posted code, DES.java and the corresponding text file destables.txt, to encrypt the following text:

This is
a test,
for the
2026 CIS
 3362 DE
S Assign
ment.(^)

using the following key in HEX: BC19 C7AC 23F1 9E70

Each line above has exactly 8 characters. To encode, encode each character into 8 bits according to its Ascii value. Your ciphertext will be binary, when converted back to Ascii many of the characters will not printable, so express your output in hexadecimal (16 HEX characters per block.)

This question is testing your ability to take the posted code an modify/use it to solve the task at hand.

Explain the modifications you made to the code in your write up and provide the final answer. It's okay to do some portion of the work by hand. (Say the program produces output in binary, then you may either manually convert it to hex, or write code to do so. Either is fine, just state in your write up what you did and if you did write code, include it.)

Solution

I examined the inputs and outputs of Professor Guha's code. I found that we would need to change the input and output of the code slightly to make it match our use case.

First, the original code required the input to be in bits, so we would first have to change Ascii to bits. I figure the easiest way to do this is in blocks of 8, matching the lines.

Finally, I notice that the output of the program is also in bits, and our required answer is in Hexadecimal, so we need to make the conversion from binary to hex afterwards.

I complete this task in these steps:

1. I created a method textToBits() that takes 8 ASCII chars and converts to bits, to fit into the given function setBlock(), written by Professor Guha

```
// Turns 8 ASCII chars into a 64 bit string for setBlock.
public static String textToBits(String text) {
        String bits = "";
        for (int i=0; i<8; i++) {
                // Pad to 8 bits so leading zeros aren't lost.
                String b = Integer.toBinaryString(text.charAt(i));
```

```
                while (b.length() < 8)
                        b = "0" + b;
                bits += b;
        }
        return bits;
}
```

2. I separate the given plaintext into 8 char segments as such:

```
String[] lines = {"This is ", "a test, ", "for the ", "2026 CIS",
        " 3362 DE", "S Assign", "ment.(^)"};
```

Easy to manually set blocks instead of coding a generalized version when the input is this small. Allows little room for error.

3. Since the original print() writes binary, I write a function that instead prints the Hex values for each block of 64 bit ciphertext from the unedited portion of the code.

```
// Prints the block as 16 hex chars, 4 bits at a time.
public void printHex(FileWriter fout) throws Exception {
        String hex = "0123456789ABCDEF";
        for (int i=0; i<64; i+=4) {
                int val = 8*block[i] + 4*block[i+1] + 2*block[i+2] + block[i+3];
                fout.write(hex.charAt(val));
        }
        fout.write("\n");
}
```

4. Cleaned up main to match with our new functions.

```
// Encrypt each 8 char line as its own block.
    String key = "BC19C7AC23F19E70";  //Added hardcoded key
    FileWriter fout = new FileWriter(new File("hexoutput.txt"));

// Set up the keys.
int[] mykey = getKey(key);
DESModify code = new DESModify(mykey);        //Class is changed to DESModify to show this
code.setKeys();                                // is the file that contained the modified code.

for (int i=0; i<lines.length; i++) {
   code.setBlock(textToBits(lines[i])); //Changed to add our new textToBits
   code.encrypt();
   code.printHex(fout);                //changed to our new print
}
```

ANSWER:

```
2B9BF03AC53913D3
0393C9B832D3C2D8
1E9DF58459BB50DD
CB20F039DE0BA4B4
AE52805571E99B98
9F8D0B6C6318E1C5
D82C52DFA9620E1C
```

5) (8 pts) Let the state matrix to AES right before the SubBytes step be the matrix shown below. Show the state of the matrix right AFTER the SubBytes step:

63	9A	45	27
C8	02	E9	F6
24	68	AC	E0
31	75	B9	FD

Solution

FB	B8	6E	CC
E8	77	1E	42
36	45	91	E1
C7	9D	56	54

Enter the bytes into the AES S box to get the substituted result.

6) (4 pts) Let the state matrix to AES right before the ShiftRows step be your answer from problem 5. Show the state of the matrix right AFTER the ShiftRows step. (This will be graded solely based on what your answer to problem 5 was. You can get this one correct even if you got problem 5 incorrect.)

Solution

FB	B8	6E	CC	0 Shift
77	1E	42	E8	1 Shift
91	E1	36	45	2 Shifts
54	C7	9D	56	3Shifts

For the ith row, cyclic shift i bytes to the left (0 indexed)

7) (10 pts) Consider the process of AES Key Expansion. Imagine that we have:

w[36] = 49 C0 E7 58 (in hex)
w[39] = 23 16 AB DF (in hex)

Calculate w[40], showing each of the following intermediate results: RotWord(temp), SubWord(RotWord(temp)), Rcon[i/4], and the result of the XOR with Rcon[i/4]

Solution

RotWord	SubWord	Rcon[i/4]	XOR	FinalResult
16 AB DF 23	47 62 9E 26	36 00 00 00	71 62 9E 26	38 A2 79 7E

First, rotate the word by 1 byte
23 16 AB DF -> 16 AB DF 23
Then plug into AES s box
16 AB DF 23 ->47 62 9e 26
XOR with Rcon[i/4]
47 62 9e 26
36 00 00 00 XOR

71 62 9e 26

71 62 9e 26
49 C0 E7 58 XOR

38 a2 79 7E

8) (12 pts) In class we discussed multiplication in the AES field GF(2^8) with the irreducible polynomial $x^8 + x^4 + x^3 + x + 1$. Based on this discussion, derive the answer for the calculation of 08 x 9E. Display your final result with two hexadecimal characters.

Solution
0000 1000 x 1001 1110

$x^3 * (x^7+x^4 +x^3 + x^2+ x)$
$x^{10}+x^7 +x^6 + x^5+ x^4$
$x^{10} = x^8(x^2) \equiv (x^4 +x^3 + x+ 1)(x^2) = (x^6 +x^5 + x^3+ x^2)$

It follows that the product is

$= (x^6 +x^5 + x^3+ x^2) + (x^7 +x^6 + x^5+ x^4)$
$= x^7 + x^4 + x^3+ x^2$
= 1001 1100 (in binary)
= **9C** in HEX

9) (16 pts) Let the input to the MixCols (during AES encryption) be $\begin{bmatrix} B5 & 34 & 29 & A6 \\ A6 & 19 & 79 & 97 \\ 63 & F5 & 4C & C2 \\ DB & D2 & FD & A3 \end{bmatrix}$.
Working by hand, show the work and output in both row 1, column 4 AND row 2, column 3? (So 8 points for each.)

(The matrix by which to "multiply" is $\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix}$.)

Solution
Row 1 Col 4:

$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix}$.

x

$\begin{bmatrix} B5 & 34 & 29 & A6 \\ A6 & 19 & 79 & 97 \\ 63 & F5 & 4C & C2 \\ DB & D2 & FD & A3 \end{bmatrix}$.

02 x A6 + 03 x 97 + 01 x C2 + 01 x A3

02 x A6
= 0010 x 1010 0110
= 1 0100 1100
0100 1100
0001 1011 XOR
0101 0111
57

03 x 97
= 0011 x 1001 0111
= 0001 x 1001 0111 + 0010 x 1001 0111
0001 x 1001 0111 = 1001 0111
0010 x 1001 0111 = 0010 1110 XOR
0001 1011
0011 0101
1001 0111
0011 0101 XOR
1010 0010= **A2**

01 x C2 = **C2**
01 x A3 = **A3**

57 + A2 + C2 + A3

0101 0111
1010 0010
1100 0010
1010 0011 XOR
1001 0100

94 FINAL ANSWER

Row 2 Col 3:

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix}.$$

x

$$\begin{bmatrix} B5 & 34 & 29 & A6 \\ A6 & 19 & 79 & 97 \\ 63 & F5 & 4C & C2 \\ DB & D2 & FD & A3 \end{bmatrix}.$$

01 x 29 + 02 x 79 + 03 x 4C + 01 x FD

01 x 29 = **29**
02 x 79
= 0000 0010 x 0111 1001
= 1111 0010
F2

03 x 4C
= 0000 0001 x 0100 1100 + 0000 0010 x 0100 1100
0000 0001 x 0100 1100 = 0100 1100
0000 0010 x 0100 1100
= 1001 1000
1001 1000
0100 1100 XOR
=1101 0100
D4

01 x FD = **FD**
29 + F2 + D4 + FD

0010 1001
1111 0010
1101 0100
1111 1101 XOR
1111 0010
F2 FINAL ANSWER