1. Given the plaintext to AES {0405060708090A0B000102030C0D0E0F}

Do the following:

1. Show the original contents of state, displayed as 4x4 matrix.
2. Show the value of state after ShiftRows.
3. Show the value of state after MixColumn, Using the following matrix:

02 03 01 01

01 02 03 01

01 01 02 03

03 01 01 02

Solutions:

a.

|  |  |  |  |
| --- | --- | --- | --- |
| 04 | 08 | 00 | 0C |
| 05 | 09 | 01 | 0D |
| 06 | 0A | 02 | 0E |
| 07 | 0B | 03 | 0F |

b.

|  |  |  |  |
| --- | --- | --- | --- |
| 04 | 08 | 00 | 0C |
| 09 | 01 | 0D | 05 |
| 02 | 0E | 06 | 0A |
| 0F | 07 | 0B | 03 |

c.

|  |  |  |  |
| --- | --- | --- | --- |
| 02 | 03 | 01 | 01 |
| 01 | 02 | 03 | 01 |
| 01 | 01 | 02 | 03 |
| 03 | 01 | 01 | 02 |

|  |  |  |  |
| --- | --- | --- | --- |
| 04 | 08 | 00 | 0C |
| 05 | 09 | 01 | 0D |
| 06 | 0A | 02 | 0E |
| 07 | 0B | 03 | 0F |

First row

1. 02 \* 04 ⊕ 03 \* 05 ⊕ 01 \* 06 ⊕ 01 \* 07
2. 02 \* 08 ⊕ 03 \* 09 ⊕ 01 \* 0A⊕ 01 \* 0B
3. 02 \* 00 ⊕ 03 \* 01 ⊕ 01 \* 02 ⊕ 01 \* 03
4. 02 \* 0C ⊕ 03 \* 0D ⊕ 01 \* 0E ⊕ 01 \* 0F

Convert to binary form

1. 0000 0100 \* 10 ⊕ (0000 0101 \* 10 ⊕ 0000 0101 \* 01)⊕ 0000 0110 \* 01 ⊕ 0000 0111 \* 01
2. 0000 1000 \* 10 ⊕ (0000 1001 \* 10 ⊕ 0000 1001 \* 01)⊕ 0000 1010 \* 01 ⊕ 0000 1011 \* 01
3. 0000 0000 \* 10 ⊕ (0000 0001 \* 10 ⊕ 0000 0001 \* 01) ⊕ 0000 0010 \* 01 ⊕ 0000 0011 \* 01
4. 0000 1100 \* 10 ⊕ (0000 1101 \* 10 ⊕ 0000 1101 \* 01) ⊕ 0000 1110 \* 01 ⊕ 0000 1111 \* 01

Shift and XOR

1. 0000 1000 ⊕ 0000 1111 ⊕ 0000 0110 ⊕ 0000 0111 = 0000 0110 = 06 =6
2. 0001 0000 ⊕ 0001 1011 ⊕ 0000 1010 ⊕ 0000 1011 = 0000 1010 = 0A =10
3. 0000 0000 ⊕ 0000 0011 ⊕ 0000 0010 ⊕ 0000 0011 = 0000 0010 = 02 = 2
4. 0001 1000 ⊕ 0001 0111 ⊕ 0000 1110 ⊕ 0000 1111 = 0000 1110 = 0E =14

Second row

1. 01 \* 04 ⊕ 02 \* 05 ⊕ 03 \* 06 ⊕ 01 \* 07
2. 01 \* 08 ⊕ 02 \* 09 ⊕ 03 \* 0A⊕ 01 \* 0B
3. 01 \* 00 ⊕ 02 \* 01 ⊕ 03 \* 02 ⊕ 01 \* 03
4. 01 \* 0C ⊕ 02 \* 0D ⊕ 03 \* 0E ⊕ 01 \* 0F

Convert to binary form

1. 0000 0100 \* 01 ⊕ 0000 0101 \* 10 ⊕ (0000 0110 \* 10 ⊕ 0000 0110 \* 01) ⊕ 0000 0111 \* 01
2. 0000 1000 \* 01 ⊕ 0000 1001 \* 10 ⊕ (0000 0110 \* 10 ⊕ 0000 1010 \* 01) ⊕ 0000 1011 \* 01
3. 0000 0000 \* 01 ⊕ 0000 0001 \* 10 ⊕ (0000 0010 \* 10 ⊕ 0000 0010 \* 01) ⊕ 0000 0011 \* 01
4. 0000 1100 \* 01 ⊕ 0000 1101 \* 10 ⊕ (0000 1110 \* 10 ⊕ 0000 1110 \* 01) ⊕ 0000 1111 \* 01

Shift and XOR

1. 0000 0100 ⊕ 0000 1010 ⊕ 0001 1010 ⊕ 0000 0111 = 0001 0011 = 13 = 38
2. 0000 1000 ⊕ 0001 0010 ⊕ 0000 0110 ⊕ 0000 1011 = 0001 0111 = 17 = 46
3. 0000 0000 ⊕ 0000 0010 ⊕ 0000 0110 ⊕ 0000 0011 = 0000 0111 = 07 = 7
4. 0000 1100 ⊕ 0001 1010 ⊕ 0001 0010 ⊕ 0000 1111 = 0000 1011 = 0B = 11

Third row

1. 01 \* 04 ⊕ 01 \* 05 ⊕ 02 \* 06 ⊕ 03 \* 07
2. 01 \* 08 ⊕ 01 \* 09 ⊕ 02 \* 0A⊕ 03 \* 0B
3. 01 \* 00 ⊕ 01 \* 01 ⊕ 02 \* 02 ⊕ 03 \* 03
4. 01 \* 0C ⊕ 01 \* 0D ⊕ 02 \* 0E ⊕ 03 \* 0F

Convert to binary form

1. 0000 0100 \* 01 ⊕ 0000 0101 \* 01 ⊕ 0000 0110 \* 10 ⊕ (0000 0111 \* 10 ⊕ 0000 0111 \* 01)
2. 0000 1000 \* 01 ⊕ 0000 1001 \* 01 ⊕ 0000 1010 \* 10 ⊕ (0000 1011 \* 10 ⊕ 0000 1011 \* 01)
3. 0000 0000 \* 01 ⊕ 0000 0001 \* 01 ⊕ 0000 0010 \* 10 ⊕ (0000 0011 \* 10 ⊕ 0000 0011 \* 01)
4. 0000 1100 \* 01 ⊕ 0000 1101 \* 01 ⊕ 0000 1110 \* 10 ⊕ (0000 1111 \* 10 ⊕ 0000 1111 \* 01)

Shift and XOR

1. 0000 0100 ⊕ 0000 0101 ⊕ 0000 1100 ⊕ 0000 1001 = 0000 0110 = 06 = 6
2. 0000 1000 ⊕ 0000 1001 ⊕ 0001 0100 ⊕ 0001 1101 = 0000 1000 = 08 = 8
3. 0000 0000 ⊕ 0000 0001 ⊕ 0000 0100 ⊕ 0000 0101 = 0000 0000 = 00 = 0
4. 0000 1100 ⊕ 0000 1101 ⊕ 0000 1110 ⊕ 0001 0001 = 0001 1110 = 1E = 60

Fourth row

1. 03 \* 04 ⊕ 01 \* 05 ⊕ 01 \* 06 ⊕ 02 \* 07
2. 03 \* 08 ⊕ 01 \* 09 ⊕ 01 \* 0A⊕ 02 \* 0B
3. 03 \* 00 ⊕ 01 \* 01 ⊕ 01 \* 02 ⊕ 02 \* 03
4. 03 \* 0C ⊕ 01 \* 0D ⊕ 01 \* 0E ⊕ 02 \* 0F

Convert to binary

1. (0000 0100 \* 10 ⊕ 0000 0100\* 01) ⊕ 0000 0101 \* 01 ⊕ 0000 0110 \* 01 ⊕ 0000 0111 \* 10
2. (0000 1000 \* 10 ⊕ 0000 1000 \* 01) ⊕ 0000 1001 \* 01 ⊕ 0000 1010 \* 01 ⊕ 0000 1011 \* 10
3. (0000 0000 \* 10 ⊕ 0000 0000 \* 01) ⊕ 0000 0001 \* 01 ⊕ 0000 0010 \* 01 ⊕ 0000 0011 \* 10
4. (0000 1100 \* 10 ⊕ 0000 1100 \* 01) ⊕ 0000 1101 \* 01 ⊕ 0000 1110 \* 01 ⊕ 0000 1111 \* 10

Shift and XOR

1. 0000 1100 ⊕ 0000 0101 ⊕ 0000 0110 ⊕ 0000 1110 = 0000 0001 = 01 = 1
2. 0001 1000 ⊕ 0000 1001 ⊕ 0000 1010 ⊕ 0001 0110 = 0000 1101 = 0D = 13
3. 0000 0000 ⊕ 0000 0001 ⊕ 0000 0010 ⊕ 0000 1100 = 0000 1111 = 0F = 15
4. 0001 0100 ⊕ 0000 1101 ⊕ 0000 1110 ⊕ 0001 1110 = 0000 1001 = 09 = 9

The result:

|  |  |  |  |
| --- | --- | --- | --- |
| 6 | 10 | 2 | 14 |
| 38 | 46 | 7 | 11 |
| 6 | 8 | 0 | 60 |
| 1 | 13 | 15 | 9 |

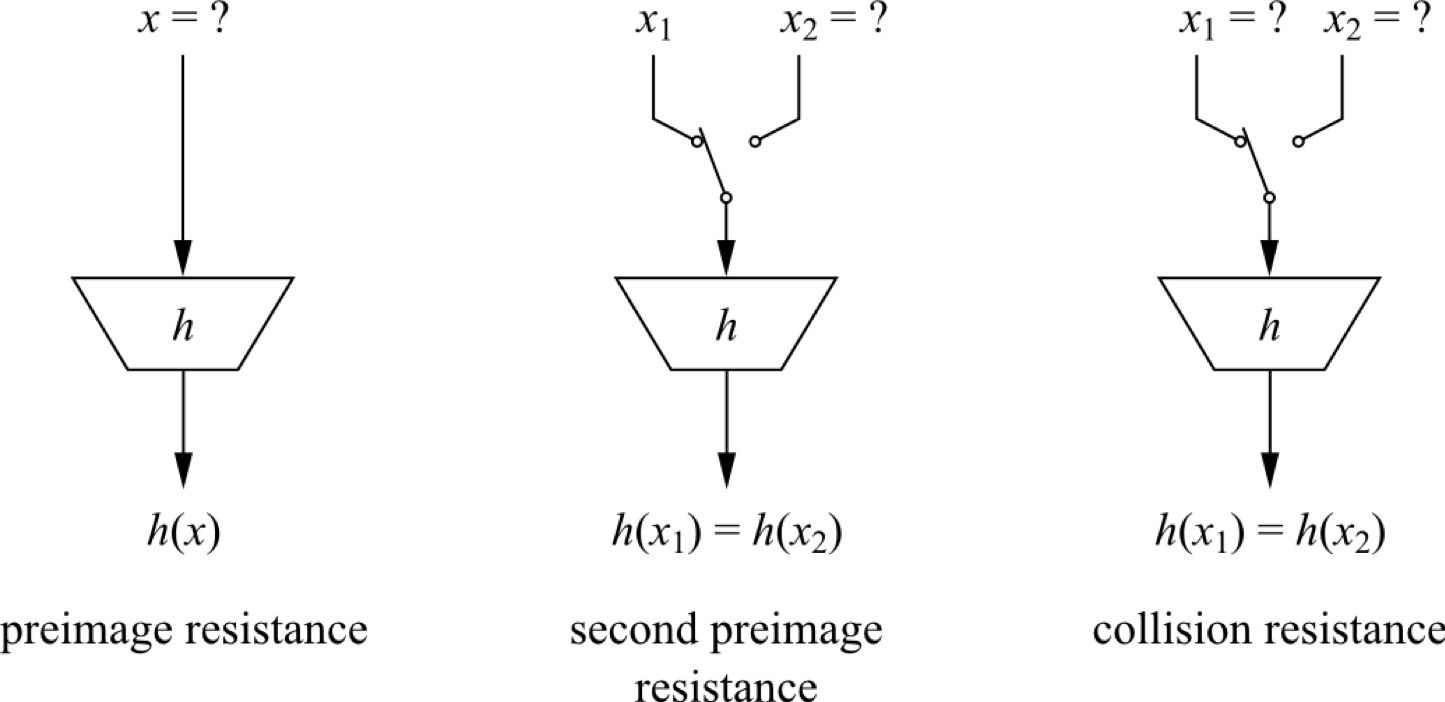
2.

1. How many bytes in **State** are affected by ShiftRows operation?
2. Explain the three security properties of Hash Functions (with diagrams).

Solutions:

1. 6-bytes.
2. pre-image resistance, second pre-image resistance, and collision resistance.

* pre-image resistance: given an output z it’s impossible to find any input x such that h(x) = z , h(x) is one way function.
* second pre-image resistance: given x, h(x) it’s computationally infeasible to find y such that h(x) = h(y).
* collision resistance: It is computationally infeasible to ﬁnd any pairs x != y such that h(x) = h(y).



1. Decrypt the cipher C= 87 using RSA with the following parameters:

e = 1127, n = 41 X 37.

1. Apply an attack to send a valid signature to Alice pretending you are Bob, provided, you know that Bob and Alice are using RSA digital signature, Public Key of Bob is 11, and the public Modula n=221.

Show that your attack is successful.

1. Convert the superincreasing knapsack (1, 4, 9, 17, 38, 79) to a general one. Then encrypt the message (101011010100011).