

Cryptography and Security — Midterm Exam

Solution

Serge Vaudenay

24.11.2011

- duration: 1h45
- no documents is allowed
- a pocket calculator is allowed
- communication devices are not allowed
- exam proctors will not answer any technical question during the exam
- answers to every exercise must be provided on separate sheets
- readability and style of writing will be part of the grade
- do not forget to put your name on every sheet!

1 A Weird Mode of Operation

In this exercise, we assume that we have a block cipher C and we use it in the following mode of operation: to encrypt a sequence of blocks x_1, \dots, x_n , we initialize a counter t to some IV value, then we compute

$$y_i = t_i \oplus C_K(x_i)$$

for every i where K is the encryption key and $t_i = IV + i$. The ciphertext is

$$IV, y_1, \dots, y_n$$

Namely, IV is sent in clear.

Q.1 Is this mode of operation equivalent to something that you already know? Say why?

It is equivalent to the ECB mode. Namely, a passive adversary can compute t_i and then $y_i \oplus t_i$ for every i . This gives the ECB encryption of x_1, \dots, x_n .

Q.2 Does the IV need to be unique?

No.

Q.3 What kind of security problem does this mode of operation suffer from?

Like the ECB mode, if the entropy of a block x_i is low, then $y_i \oplus t_i$ repeats. For instance, $x_i = x_j$ is equivalent to $y_i \oplus t_i = y_j \oplus t_j$ which can be observed with values which are sent over the insecure channel.

2 RSA Modulo 1 000 001

Given $a_1, a_2, \dots, a_n \in \{0, 1, \dots, 9\}$, we denote by $\overline{a_1 a_2 \dots a_n}$ the decimal number equal to $10(10(\dots 10a_1 + a_2 \dots) + a_{n-1}) + a_n$.

Q.1 Consider a decimal number $\overline{abc def}$. Show that

$$\overline{abc def} \equiv \overline{ab} - \overline{cd} + \overline{ef} \pmod{101}$$

As an application, compute $336\,634 \pmod{101}$ and $663\,368 \pmod{101}$.

We have

$$\begin{aligned} \overline{abc def} &= 10(10(10(10(10a + b) + c) + d) + e) + f \\ &= 100^2(10a + b) + 100(10c + d) + (10e + f) \end{aligned}$$

Since $100 \equiv -1 \pmod{101}$, this writes

$$\overline{abc def} \equiv (10a + b) - (10c + d) + (10e + f) \pmod{101}$$

which is what we had to prove. So,

$$336\,634 \equiv 33 - 66 + 34 = 1 \pmod{101}$$

and

$$663\,368 \equiv 66 - 33 + 68 = 101 \equiv 0 \pmod{101}$$

which yields $336\,634 \pmod{101} = 1$ and $663\,368 \pmod{101} = 0$.

Q.2 Compute the inverse of $x = 1\,000$ modulo $p = 101$.

A general method consists of applying the extended Euclid algorithm. We have

$$\begin{aligned} \mathbf{x}_1 &= (1\,000, 1, 0) & \mathbf{x}_2 &= (101, 0, 1) \\ \mathbf{x}_2 &= (101, 0, 1) & \mathbf{x}_3 &= (91, 1, -9) & \mathbf{x}_3 &= \mathbf{x}_1 - 9\mathbf{x}_2 \\ \mathbf{x}_3 &= (91, 1, -9) & \mathbf{x}_4 &= (10, -1, 10) & \mathbf{x}_4 &= \mathbf{x}_2 - \mathbf{x}_3 \\ \mathbf{x}_4 &= (10, -1, 10) & \mathbf{x}_5 &= (1, 10, -99) & \mathbf{x}_5 &= \mathbf{x}_3 - 9\mathbf{x}_4 \\ \mathbf{x}_5 &= (1, 10, -99) & \mathbf{x}_6 &= (0, -101, 1\,000) & \mathbf{x}_6 &= \mathbf{x}_4 - 10\mathbf{x}_5 \end{aligned}$$

so $1 = 1\,000 \times 10 - 101 \times 99$. Therefore, $x^{-1} \pmod{p} = 10$.

Q.3 Consider a decimal number $\overline{abc def}$. Show that

$$\overline{abc def} \equiv \overline{ab00} - \overline{ab} + \overline{cdef} \pmod{9\,901}$$

As an application, compute $336\,634 \pmod{9\,901}$ and $663\,368 \pmod{9\,901}$.

Just like before, we have

$$\begin{aligned}\overline{abcdef} &= 10(10(10(10(10a + b) + c) + d) + e) + f \\ &= 10^4(10a + b) + \overline{cdef}\end{aligned}$$

Since $10^4 \equiv 100 - 1 \pmod{9901}$, this writes

$$\overline{abcdef} \equiv 100(10a + b) - (10a + b) + \overline{cdef} \pmod{101}$$

which is what we had to prove. So,

$$336\,634 \equiv 3300 - 33 + 6634 = 9901 \equiv 0 \pmod{9901}$$

and

$$663\,368 \equiv 6600 - 66 + 3368 = 9902 \equiv 1 \pmod{9901}$$

which yields $336\,634 \bmod 101 = 0$ and $663\,368 \bmod 101 = 1$.

Q.4 Compute $x^{199} \bmod q$ for $x = 1\,000$ and $q = 9901$.

Then, $x^{199} \equiv x^3 \times (x^4)^{49} \pmod{q}$. We have

$$x^2 = 1\,000^2 = 1\,000\,000 \equiv 10000 - 100 + 0000 = 9900 \equiv -1 \pmod{q}$$

so $x^4 \bmod q = 1$ and $x^3 \bmod q = -x \bmod q = 8901$. Thus, $b = 8901$.

Applying the square-and-multiply algorithm would have led to $x^4 \bmod q = 1$ as well.

Q.5 Given a and b , show that $x = 336\,634a + 663\,368b$ is such that $x \bmod 101 = a$ and $x \bmod 9901 = b$.

We have $336\,634 \bmod 101 = 1$ and $663\,368 \bmod 101 = 0$ so, by linearity, we have $x \equiv a \pmod{101}$. We have $336\,634 \bmod 9901 = 0$ and $663\,368 \bmod 9901 = 1$ so, by linearity, we have $x \equiv b \pmod{9901}$. This expression for x is actually the inverse formula for the Chinese remainder theorem using moduli 101 and 9901 (note that they are coprime).

Q.6 Given $p = 101$ and $q = 9901$, we let $N = pq$. Compute $\varphi(N)$ and factor it into a product of prime numbers.

Since p and q are prime, we have

$$\varphi(N) = (p - 1)(q - 1) = 100 \times 9900 = 990\,000 = 10^4 \times 9 \times 11 = 2^4 \times 3^2 \times 5^2 \times 11$$

Q.7 Let e be an integer. Show that e is a valid RSA exponent for modulus N if and only if there is no prime factor of $\varphi(N)$ dividing e .

e is a valid RSA exponent if and only if $\gcd(e, \varphi(N)) = 1$ which is if and only if none of the prime factors of $\varphi(N)$ divide e . Since the list of prime factors of $\varphi(N)$ is $\{2, 3, 5, 11\}$, we obtain the result.

Q.8 Show that $e = 199$ is a valid RSA exponent for modulus N and compute the encryption of $x = 1\,000$ for this public key.

199 has no prime factor in $\{2, 3, 5, 11\}$ so it is a valid exponent. To compute $x^e \bmod N$, we use the Chinese remainder theorem. We compute $a = x^e \bmod p$ and $b = x^e \bmod q$.

We have $a = x^{199} \bmod 101 = x^{199 \bmod 100} \bmod 101 = x^{-1} \bmod 101 = 10$ due to Q.2. Similarly, we have $b = x^{199} \bmod 9\,901 = 8\,901$ due to Q.4. Finally,

$x^e \bmod N = (336\,634 \times 10 + 663\,368 \times 8\,901) \bmod N = 5\,908\,004\,908 \bmod N = 999\,001$

So, the encryption of x is 999 001.

3 AES Galois Field and AES Decryption

We briefly recall the AES block cipher here. It encrypts a block specified as a 4×4 matrix of bytes s and using a sequence W_0, \dots, W_n of matrices which are derived from a secret key. For convenience the row and columns indices range from 0 to 3. For instance, $s_{1,3}$ means the term of s in the second row and last column. The main AES encryption function is defined by the following pseudocode:

```
AESencryption( $s, W$ )
1: AddRoundKey( $s, W_0$ )
2: for  $r = 1$  to  $n - 1$  do
3:   SubBytes( $s$ )
4:   ShiftRows( $s$ )
5:   MixColumns( $s$ )
6:   AddRoundKey( $s, W_r$ )
7: end for
8: SubBytes( $s$ )
9: ShiftRows( $s$ )
10: AddRoundKey( $s, W_n$ )
```

AddRoundKey(s, W_r) is replacing s by $s \oplus W_r$, the component-wise XOR of matrices s and W_r . **SubBytes**(s) is replacing s by a new matrix in which the term at position i, j is $S(s_{i,j})$, where S is a fixed permutation of the set of all byte values. **ShiftRows**(s) is replacing s by a new matrix in which the term at position i, j is $s_{i, i+j \bmod 4}$. **MixColumns**(s) is replacing s by a new matrix in which the column at position j is $M \times s_{:,j}$, where $s_{:,j}$ denotes the column at position j of s and M is a fixed matrix defined by

$$M = \begin{pmatrix} 0x02 & 0x03 & 0x01 & 0x01 \\ 0x01 & 0x02 & 0x03 & 0x01 \\ 0x01 & 0x01 & 0x02 & 0x03 \\ 0x03 & 0x01 & 0x01 & 0x02 \end{pmatrix}$$

The matrix product inherits from the algebraic structure $\text{GF}(256)$ on the set of all byte values. Namely, each byte represents a polynomial on variable x of degree at most 7 and coefficients in \mathbf{Z}_2 . Polynomials are added and multiplied modulo 2 and modulo $P(x) = x^8 + x^4 + x^3 + x + 1$. The correspondence between bytes and polynomial works as follows: each byte a is a sequence of 8 bits a_7, \dots, a_0 which is represented in hexadecimal $0xuv$ where u and v are two hexadecimal digits (i.e. between 0 and f), u encodes $a_7a_6a_5a_4$, and v encodes $a_3a_2a_1a_0$ by the following encoding rule:

0000	→0	0100	→4	1000	→8	1100	→c
0001	→1	0101	→5	1001	→9	1101	→d
0010	→2	0110	→6	1010	→a	1110	→e
0011	→3	0111	→7	1011	→b	1111	→f

Q.1 Provide a pseudocode for **AESdecryption**(s, W), for AES decryption.

We remark that **AddRoundKey** is self-inverse. We further remark that **SubBytes** and **ShiftRows** commute.

AESdecryption(s, W)

- 1: **AddRoundKey**(s, W_n)
- 2: **for** $r = n - 1$ **down to** 1 **do**
- 3: **InvSubBytes**(s)
- 4: **InvShiftRows**(s)
- 5: **AddRoundKey**(s, W_r)
- 6: **InvMixColumns**(s)
- 7: **end for**
- 8: **InvSubBytes**(s)
- 9: **InvShiftRows**(s)
- 10: **AddRoundKey**(s, W_0)

InvSubBytes(s) is replacing s by a new matrix in which the term at position i, j is $S^{-1}(s_{i,j})$. **InvShiftRows**(s) is replacing s by a new matrix in which the term at position i, j is $s_{i, -i+j \bmod 4}$. **InvMixColumns**(s) is replacing s by a new matrix in which the column at position j is $M^{-1} \times s_{.,j}$.

Q.2 Which polynomial does 0x2b represent?

2 encodes 0010 and b encodes 1011, so 0x2b encodes the bitstring 0010 1011 which represents $x^5 + x^3 + x + 1$.

Q.3 Compute 0x53 + 0xb8.

Addition is a simple XOR. 0x53 encodes 0101 0011 and 0xb8 encodes 1011 1000. The XOR is 1110 1011 which is encoded by 0xeb. So, $0x53 + 0xb8 = 0xeb$.

Q.4 Compute $0x21 \times 0x25$.

0x21 represents the polynomial $x^5 + 1$. 0x25 represents the polynomial $x^5 + x^2 + 1$. We have

$$(x^5 + 1) \times (x^5 + x^2 + 1) = x^{10} + x^7 + 2x^5 + x^2 + 1 \equiv x^{10} + x^7 + x^2 + 1$$

Since $x^8 \equiv x^4 + x^3 + x + 1$ we have $x^9 \equiv x^5 + x^4 + x^2 + x$ and $x^{10} \equiv x^6 + x^5 + x^3 + x^2$. So,

$$(x^5 + 1) \times (x^5 + x^2 + 1) \equiv x^{10} + x^7 + x^2 + 1 \equiv x^7 + x^6 + x^5 + x^3 + 2x^2 + 1 \equiv x^7 + x^6 + x^5 + x^3 + 1$$

Now, $x^7 + x^6 + x^5 + x^3 + 1$ is represented by 0xe9. So, $0x21 \times 0x25 = 0xe9$.

Q.5 Compute the inverse of 0x02.

Hint: look at $P(x)$.

Since $x^8 + x^4 + x^3 + x + 1 \equiv 0$, by multiplying by x^{-1} we obtain $x^7 + x^3 + x^2 + 1 + x^{-1} \equiv 0$, so $x^{-1} = x^7 + x^3 + x^2 + 1$. Changing this into hexadecimal bytes, this gives

$$0x02^{-1} = 0x8d$$

Q.6 Show that M^{-1} is of form

$$M^{-1} = \begin{pmatrix} 0x0e & 0x0b & 0x0d & 0x09 \\ 0x09 & \cdot & \cdot & \cdot \\ 0x0d & \cdot & \cdot & \cdot \\ 0x0b & \cdot & \cdot & \cdot \end{pmatrix}.$$

where all missing terms are in the set $\{0x09, 0x0b, 0x0d, 0x0e\}$.

We first compute

$$\begin{pmatrix} 0x02 & 0x03 & 0x01 & 0x01 \\ 0x01 & 0x02 & 0x03 & 0x01 \\ 0x01 & 0x01 & 0x02 & 0x03 \\ 0x03 & 0x01 & 0x01 & 0x02 \end{pmatrix} \times \begin{pmatrix} 0x0e \\ 0x09 \\ 0x0d \\ 0x0b \end{pmatrix} = M \times \begin{pmatrix} 0x0e \\ 0x09 \\ 0x0d \\ 0x0b \end{pmatrix}$$

By writing this with polynomials, this gives

$$M \times \begin{pmatrix} 0x0e \\ 0x09 \\ 0x0d \\ 0x0b \end{pmatrix} = \begin{pmatrix} x & x+1 & 1 & 1 \\ 1 & x & x+1 & 1 \\ 1 & 1 & x & x+1 \\ x+1 & 1 & 1 & x \end{pmatrix} \times \begin{pmatrix} x^3 + x^2 + x \\ x^3 + 1 \\ x^3 + x^2 + 1 \\ x^3 + x + 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

By rotating the columns of M and the rows of the vector in the product we obtain

$$\begin{pmatrix} 0x01 & 0x02 & 0x03 & 0x01 \\ 0x01 & 0x01 & 0x02 & 0x03 \\ 0x03 & 0x01 & 0x01 & 0x02 \\ 0x02 & 0x03 & 0x01 & 0x01 \end{pmatrix} \times \begin{pmatrix} 0x0b \\ 0x0e \\ 0x09 \\ 0x0d \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Now, by rotating the rows of the matrix and of the result, we obtain

$$\begin{pmatrix} 0x02 & 0x03 & 0x01 & 0x01 \\ 0x01 & 0x02 & 0x03 & 0x01 \\ 0x01 & 0x01 & 0x02 & 0x03 \\ 0x03 & 0x01 & 0x01 & 0x02 \end{pmatrix} \times \begin{pmatrix} 0x0b \\ 0x0e \\ 0x09 \\ 0x0d \end{pmatrix} = M \times \begin{pmatrix} 0x0b \\ 0x0e \\ 0x09 \\ 0x0d \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}$$

By redoing the same, we obtain

$$M \times \begin{pmatrix} 0x0d \\ 0x0b \\ 0x0e \\ 0x09 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}$$

and

$$M \times \begin{pmatrix} 0x09 \\ 0x0d \\ 0x0b \\ 0x0e \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

So,

$$M \times \begin{pmatrix} 0x0e & 0x0b & 0x0d & 0x09 \\ 0x09 & 0x0e & 0x0b & 0x0d \\ 0x0d & 0x09 & 0x0e & 0x0b \\ 0x0b & 0x0d & 0x09 & 0x0e \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

which gives the inverse of M and proves the required properties.