Academic Year 2016–2017

Network Security and Cryptography

Exercises

Giacomo Verticale
giacomo.verticale@polimi.it
verticale.faculty.polimi.it

May 22, 2017

Contents

1

Information-Theoretic Security

Vigènere cipher

Given a message m of size l, let $m = m[0] \| m[1] \| \ldots \| m[l]$.

$k = \text{Gen}()$ k is password of length n characters

$c = \text{Enc}(k, m)$ with $c[i] = m[i] + k[i \bmod n] \bmod 26$

$m\,\text{Dec}(k, m)$ $m[i] = c[i] - k[i \bmod n] \bmod 26$

Exercise 1.1 Consider a Vigènere cipher with messages of $l = 2$ characters and a random password. The password is chosen uniformly according to the following rule:

- with probability $1/2$ the password is one-character long ($n = 1$)
- with probability $1/2$ the password is two-character long ($n = 2$)

Show that the above scheme is not secure against the following adversary, $\mathcal{A}$.

Adversary

```
Send $m_0 = \texttt{aa}$ and $m_1 = \texttt{ab}$ to the challenger
Receive the ciphertext $c = c[0] \| c[1]$ from the challenger
if $c[0] = c[1]$ then
    return $b' = 0$
else
    return $b' = 1$
end if
```

Solution We need to calculate the advantage of the adversary.

$$\text{Adv} = |\Pr[\text{EXP}(0) = 1] - \Pr[\text{EXP}(1) = 1]|$$

Consider the case $b = 0$. If the password length is $n = 1$, then $c[0] = c[1]$ independently of the key, so $\Pr[b' = 1] = 0$. Otherwise, if $n = 2$, then $c[0] = c[1]$ only if $k[0] = k[1]$. So, $\Pr[b' = 1] = 25/26$. Thus:

$$\Pr[\text{EXP}(0) = 1] = \frac{1}{2} \cdot 0 + \frac{1}{2} \cdot \frac{25}{26}$$

Consider the case $b = 1$. If $n = 1$, then $c[0] \neq c[1]$. If $n = 2$ then $c[0] = c[1]$ with probability $1/26$. Thus:

$$\Pr[\text{EXP}(1) = 1] = \frac{1}{2} \cdot 1 + \frac{1}{2} \cdot \frac{25}{26}$$

Concluding:

$$\text{Adv} = \left| \frac{1}{2}\frac{25}{26} - \frac{1}{2} - \frac{1}{2}\frac{25}{26} \right| = \frac{1}{2}$$

Since the advantage is not zero, the scheme is not information theoretically-secure against this adversary.

Exercise 1.2 Consider a Vigènere cipher. Messages are 4-character sequences of letters. Passwords are n-character long.

Assume that there are three possible messages, with different probabilities.

m_1	BEBA	0.5
m_2	DEDA	0.3
m_3	CFCB	0.2

a) Calculate the entropy of the source.

b) Calculate the entropy of the key with $n = 1, 2, 4$.

c) Calculate the entropy of the ciphertext with $n = 1, 2, 4$.

d) Calculate the information leak of the ciphertext with $n = 1, 2, 4$

e) Is the cipher of this exercise perfectly secret?

Solution (a)

$$H(\mathcal{M}) = -0.5 \log_2 0.5 - 0.3 \log_2 0.3 - 0.2 \log_2 0.2 = 1.48$$

(b)

n	$H(\mathcal{K})$
1	4.7
2	9.4
4	18.8

(c) Let call C_i the set of possible ciphertexts for message m_i.
With $n = 1$, we have

- $C_1 = \{\text{BEBA}, \text{CFCB}, \text{DGDC}, \dots\}$
- $C_2 = \{\text{DEDA}, \text{EFEB}, \text{FGFB}, \dots\}$
- $C_3 = \{\text{CFCB}, \text{DGDC}, \text{EHED}, \dots\}$

So $C_1 = C_3$, while C_2 is distinct. Thus, we have $2 \times 26 = 52$ ciphertexts. The first set has probability $0.5 + 0.2 = 0.7$. The second set has probability 0.3.

$$H(\mathcal{C}) = -26 \times 0.7/26 \log_2 0.7/26 - 26 \times 0.3/26 \log_2 0.3/26 = 5.58$$

With $n = 2$, we have again $C_1 = C_2 = C_3$. In fact, the key $(2, 0)$ can map BEBA into DEDA, while the key $(1, 1)$ can map BEBA into CFCB. Consequently all the messages generate the same $26^2 = 676$ ciphertexts.

$$H(\mathcal{C}) = -676/676 \log_2(1/676) = 9.4$$

With $n = 4$, we have $C_1 = C_2 = C_3$. Thus, we have $26^4 = 456976$ equally likely ciphertexts.

$$H(\mathcal{C}) = 26^4/26^4 \log_2(1/26^4) = 18.8$$

(d) With $n = 1$

$$H(\mathcal{C}|\mathcal{M}) = H(C_1) \Pr(m_1) + H(C_2) \Pr(m_2) + H(C_3) \Pr(m_3) = 4.7$$

Then

$$I(\mathcal{M}; \mathcal{C}) = H(\mathcal{C}) - H(\mathcal{C}|\mathcal{M}) = 5.58 - 4.7 = 0.88$$

With $n = 2$

$$H(\mathcal{C}|\mathcal{M}) = H(C_1)\Pr(m_1) + H(C_2)\Pr(m_2 + m_3) = 9.4$$

Then

$$I(\mathcal{M};\mathcal{C}) = H(\mathcal{C}) - H(\mathcal{C}|\mathcal{M}) = 9.4 - 9.4 = 0$$

With $n = 4$

$$H(\mathcal{C}|\mathcal{M}) = H(C_1)\Pr(m_1 + m_2 + m_3) = 18.8$$

Then

$$I(\mathcal{M};\mathcal{C}) = H(\mathcal{C}) - H(\mathcal{C}|\mathcal{M}) = 18.8 - 18.8 = 0$$

(e) If these three are the only possible messages, then with $n = 2$ and $n = 4$ we have perfect secrecy.

2

Symmetric Cryptography

2.1 Computational Security

Exercise 2.1 Consider a general encryption scheme with key length n. An adversary running for t computer cycles can break the scheme with probability at most $t/2^n$, corresponding to an exhaustive search.

Calculate the running time required for breaking the scheme for $n = 60$ and $n = 128$ considering a 4-GHz processor.

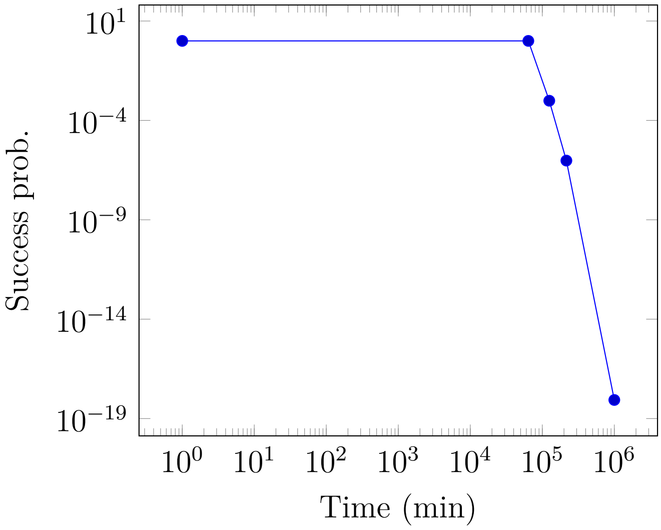
Solution We consider the scheme broken when the success probability is equal to 1, i.e. $t = 2^n$.

For $n = 60$, the running time is $\dfrac{2^{60}}{4 \times 10^9}$ seconds equal to 9 years.

For $n = 128$, the running time is more than 2×10^{21} years.

Exercise 2.2 Consider a general encryption scheme with key length n. An adversary running for n^3 minutes can break the scheme with probability at most $2^{40}/2^n$. Plot the success probability versus the running time.

Solution See figure.



Exercise 2.3 Consider a general encryption scheme with key length n. The encryption algorithm runs for $10^6 \cdot n^2$ cycles. An adversary running for $10^8 \cdot n^4$ cycles breaks the scheme with probability at most $2^{n/2}$. All parties use 2-GHz computers. The key length is $n = 80$. Calculate the running times of the algorithms and the success probability.

Now consider 8-GHz computers. Choose a key length that requires a similar encryption time and calculate the new running time and success probability of the adversary.

Solution With a 2-GHz computer the encryption time is $10^6 \cdot 6400$ cycles, i.e. 3.2 s. The adversary runs for 3 weeks with a success probability of 2^{-40}.

With a 8-GHz computer the new key size is:

$$n = \sqrt{80^2/2 \cdot 8} = 160$$

The adversary runs for 13 weeks with a success probability if 2^{-80}.

Exercise 2.4 Given a seed of size n bits, let $G(s) = s \| (s[0] \oplus \cdots \oplus s[n-1])$, so the output of G is $n+1$ bits. Prove that G is not a secure PRG.

Solution The generator G is predictable, because there exists an algorithm $\mathcal{B} \colon \{0,1\}^n \to \{0,1\}$ that predicts the $(n+1)$th bit with probability 1.

We can write an algorithm $\mathcal{A}$ that can distinguish G from a random generator by using $\mathcal{B}$ as a subroutine.

Adversary $\mathcal{A}$

```
Input: a string $x$ of $n+1$ bits
  Let $x[0..n-1]$ the first $n$ bits of $x$
  $y = \mathcal{B}(x[0..n-1])$
  if $y = x[n]$ then
    return 1
  else
    return 0
  end if
```

We can calculate the advantage:

$$\text{Adv} = |\Pr[\text{EXP}(0) = 1] - \Pr[\text{EXP}(1) = 1]|$$

With $\Pr[\text{EXP}(0) = 1] = 1/2$ and $\Pr[\text{EXP}(1) = 1] = 1$, we get

$$\text{Adv} = 1/2$$

Exercise 2.5 Consider a PRG $G\colon \{0,1\}^n \to \{0,1\}^{2n}$. Consider the following adversary $\mathcal{A}$:

```
Input: a string $x$ of $2n$ bits.
  for $s = 0$ to $2^{n-1}$ do
    if $G(s) = x$ then
      return 1
    end if
  end for
  return 0
```

Calculate the advantage of $\mathcal{A}$ in distinguishing G from a random generator. Is this a proof that G is not secure?

Solution We can calculate the advantage:

$$\text{Adv} = |\Pr[\text{EXP}(0) = 1] - \Pr[\text{EXP}(1) = 1]|$$

In EXP(0), the input is random, so the probability that the random string is actually a possible output of G is the size of the image of G divided by 2^{2n}. Assuming that the image of G is roughly the size of the domain, we get

$$\Pr[\text{EXP}(0) = 1] = \frac{2^n}{2^{2n}} = 2^{-n}$$

If the input is not random, then the algorithm is always right. Therefore:

$$\text{Adv} = |2^{-n} - 1|$$

which is not negligible.

This does not count as a proof, because the algorithm is not efficient, since it has time complexity $O(2^n)$.

Exercise 2.6 Let $F\colon \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ be a length-preserving pseudorandom function. For the following constructions of a keyed function

$$F'\colon \{0,1\}^n \times \{0,1\}^{n-1} \to \{0,1\}^{2n}$$

state whether F' is a pseudorandom function. If yes, prove it; if not, show an attack.

$$F'(k,x) = F(k,0\|x)\|F(k,x\|1)$$

Solution No. Consider the following attacker:

Algorithm $\mathcal{B}$

```
Send the queries $x_1 = 0\dots0$ and $x_2 = 0\dots1$
Receive $y_1$ and $y_2$
Let $y_i[0]$ be the first half of $y_i$ and $y_i[1]$ the second half.
if $y_1[1] = y_2[0]$ then
    return 0
else
    return 1
end if
```

Observe that:

$$\begin{aligned} F'(k,x_1) = F'(k,0\dots0) = F(k,00\dots00)\|F(k,00\dots01) \\ F'(k,x_2) = F'(k,0\dots1) = F(k,00\dots01)\|F(k,00\dots11) \end{aligned}$$

So, if the challenger uses F', the algorithm always gives the right answer. Otherwise the algorithm gives the right answer with probability $1-2^{-n}$.

Thus,

$$|\Pr[\mathrm{EXP}(0)=1] - \Pr[\mathrm{EXP}(1)=1]| = |2^{-n}-1|$$

which is not negligible.

Exercise 2.7 Consider the following keyed function F: For security parameter n, the key is an $n \times n$ boolean matrix A and an n-bit boolean vector b. Define

$$F = Ax + b$$

where all operations are done modulo 2. Show that F is not a pseudorandom function.

Solution Consider the algorithm:

$\mathcal{A}$

```
Choose $x_1 = 0$, and two distinct $x_2$ and $x_3$ randomly nonzero.
Send $x_1, x_2, x_3, x_4 = x_2 \oplus x_3$
Receive $y_1, y_2, y_3, y_4$
if $y_4 = y_1 \oplus y_2 \oplus y_3$ then
  return 1
else
  return 0
end if
```

Note that, if the challenger uses F

$$\begin{aligned} y_4 &= Ax_4 + b = A(x_2 + x_3) + b = Ax_2 + Ax_3 + b \pmod 2 \\ y_1 + y_2 + y_3 &= b + Ax_2 + b + Ax_3 + b = Ax_2 + Ax_3 + b \pmod 2 \end{aligned}$$

Clearly, if the challenger uses F, then $\mathcal{A}$ yields 1 with probability 1. Otherwise $\mathcal{A}$ yields 1 with a small nonzero probabilty.

The advantage is non-negligible.

Exercise 2.8 Prove that if F is a length-preserving pseudorandom function, then

$$G(s) = F(s,1) \| F(s,2) \| \ldots \| F(s,l)$$

is a pseudorandom generator with output size ln bits.

Solution We prove that if $G(s)$ is not secure, then F is not a PRF.

If $G(s)$ is not secure, then there exists an algorithm $\mathcal{B}$ and an index i such that $\mathcal{B}(G(s)[0..i])$ predicts $(G(s)[i+1])$ with non-negligible probability.

Thus, we can write an algorithm $\mathcal{A}$ that can use $\mathcal{B}$ to predict at least one bit of the output of one of the $F(s,j)$.

3

Secret Sharing

Exercise 3.1 A secret is shared among multiple entities using a Shamir's Secret Sharing Scheme with threshold 2 and $p = 17$. The dealer distributes the following shares to different parties: $(1, 8)$, $(2, 11)$, $(4, 0)$.

a) Find the secret number m.

Solution We need to solve the equation:

$$\begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} m \\ s_1 \end{pmatrix} \equiv \begin{pmatrix} 8 \\ 11 \end{pmatrix} \pmod{17}$$

Then

$$\begin{pmatrix} m \\ s_1 \end{pmatrix} \equiv \begin{pmatrix} 2 & -1 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} 8 \\ 11 \end{pmatrix} \pmod{17}$$

So $m = 16 - 11 = 5$ and $s_1 = 11 - 8 = 3$.

b) Verify whether the share $(3, 12)$ is correct.

Solution

$$5 + 3 \times 3 \bmod 17 = 14$$

The share is not correct.

c) An information management system comprises 10 entities of type A and 14 of type B. A dealer wants to distribute a secret so that 3 entities of type A and 2 of type B are necessary to recover the secret. Describe a system that can achieve the desired behavior.

Solution The secret must be split in two parts, then it can be used a (3,10) threshold scheme for the first and a (2,14) scheme for the second.

Exercise 3.2 The key for the European Central Bank (ECB) vault is a number shared among the European countries. To open the vault, at least t out of w Board members are present. The combination of the vault is divided using Shamir's secret sharing scheme, with the public prime p.

a) Let $t = 2$, $w = 10$, $p = 13$, then two of the shares are $(1, 2)$ and $(2, 8)$. Calculate the combination of the vault.

Solution We write the equation as

$$\begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} m \\ s_1 \end{pmatrix} = \begin{pmatrix} 2 \\ 8 \end{pmatrix} \bmod 13$$

from which

$$\begin{pmatrix} m \\ s_1 \end{pmatrix} = \begin{pmatrix} 2 & -1 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} 2 \\ 8 \end{pmatrix} = \begin{pmatrix} 9 \\ 6 \end{pmatrix} \bmod 13$$

$$s(x) = 6x + 9 \bmod 13$$

b) Calculate the secret polynomial $s(x)$ and the share for $x = 3$.

Solution

$$s(3) = 18 + 9 \bmod 13 = 1$$

c) The vault combination is chosen randomly. Calculate the entropy of the vault combination.

Solution There are $p = 13$ possible equally likely combinations, each with probability $q = 1/p$.

$$H = -pq \log_2 q = \log_2 p = 3.7$$

d) Assume that the dealer always chooses p as the smallest prime that satisfies the requirements for Shamir's Secret Sharing. Explain how this algorithm leaks information about the secret. Calculate the entropy of the share.

Solution Let p_i is the smallest prime larger than the secret m. Then, m can be any integer in the prime gap between p_i and p_{i-1}.

Since the density of primes around p is about $1/\log p$, the prime gap is approximately $g = \log p$, so there are g possible secrets, each with probability $q = 1/g$. The entropy is

$$H = -gq \log_2 q = \log_2 \log p$$

For $p = 13$ the gap is 2.56, so the entropy is 1.36.

In fact, if $p = 13$, the secret is either $m = 12$ or $m = 11$, which is one bit of entropy.

4

Hash functions

Exercise 4.1 Consider the block cipher E_k, which uses blocks of 4 bits nd keys of 4 bits implemented as follows:

$$E_k(x) = (10^k \bmod 17) + x \bmod 16$$

Consider a message M made by N blocks of 4 bits each: $m_1, m_2, \ldots, m_N$. Consider also the hash functions $h(M)$ and $g(M)$ obtained as follows:

$$\begin{aligned} f_0 &= 0 & z_0 &= 0 \\ f_i &= E_0(f_{i-1} \oplus m_i) & z_i &= E_{m_i}(z_{i-1}) \\ h(M) &= f_N & g(M) &= z_N \end{aligned}$$

a) Write the equations which define a CBC-MAC obtained from the block cipher E_k and compute the MAC of the binary message `01000100`.

b) Write the equations which define a HMAC obtained from the function g.

c) Considering h and g, find a preimage of the number 3.

d) Considering h and g, find a second preimage for the number 3.

Solution a)

$$\begin{aligned} c_0 &= IV \quad \text{usually IV=0} \\ c_1 &= E_k(c_{i-1} \oplus m_1) \\ MAC_k(M) &= c_N \end{aligned}$$

Let $10^k \bmod 17 = \xi$.

$$MAC_k(01000100) = E_k(E_k(4) \oplus 4) == (\xi + (((\xi + 4) \bmod 16) \oplus 4) \bmod 16).$$

b)
$$HMAC_k(M) = g(k \oplus \text{opad} \| g(k \oplus \text{ipad} \| M))$$

c) We are looking for M such that $h(M) = 3$ (or $g(M) = 3$). Let us assume that M is composed by only one block m_1. So, we have:

$$h(m_1) = E_0(m_1) = 1 + x \bmod 16 = 3 \qquad m_1 = 2$$
$$g(m_1) = E_{m_1}(0) = 10^{m_1} \bmod 17 \bmod 16 = 3$$

j, k	10^j	$3 \cdot 10^{-5k}$
0	1	3
1	10	9
2	15	10
3	14	13
4	4	5

Therefore, $10^{1+10} = 3$, and thus $m_1 = 11$.

d) We are looking for M such that $h(M) = H(3)$ (or $g(M) = g(3)$). Let us assume that M is composed by two blocks. For h, we obtain:

$$h(m_1 \| m_2) = E_0((E_0(m_1) \oplus m_2) = E_0(3)$$
$$E_0(m_1) \oplus m_2 = 3$$

Any pair that satisfied the equation above will do, e.g., $m_1 = 15$ and $m_2 = 3$.

For g, we obtain:

$$g(m_1 \| m_2) = E_{m_2}(E_{m_1}(0)) = E_3(0)$$

by choosing $m_2 = 3$, we obtain

$$E_{m_1}(0) = 0$$
$$10^{m_1} \bmod 17 = 16 = -1$$

This is a simple DLP problem: $m_1 = 8$.

Exercise 4.2 Let $h(m) : \{0,1\}^* \to \{0,1\}^{256}$ be a hash function which has the following property:

$$m \equiv m' \pmod{2^{32}} \implies h(m) = h(m')$$

1. Say how many strings of 256 bits have a preimage.
2. Given m, explain how to take advantage of the property to find a second preimage. Find how many evaluations of the function are necessary to ensure a probability of success of at least 50%.
3. Given $h(m)$, explain how to take advantage of the property to find a preimage. Find how many evaluations of the function are necessary to ensure a probability of success of at least 50%.

Solution

1. There are at most 2^{32} distinct images.
2. Given m, any $m' = m + k \cdot 2^{32}$ is a second preimage. You do not need to evaluate the function.
3. We try only with the $m \in \mathbb{Z}_{2^{32}}$. Assuming the images are all distinct, we obtain
$$\frac{q}{2^{32}} = \frac{1}{2}$$
Therefore 2^{31} evaluations are necessary.

Exercise 4.3 Let $h(m) : \{0,1\}^* \to \{0,1\}^{128}$ be a hash function which has the following property:

$$\text{Par}(m) = \text{Par}(h(m)) \tag{4.1}$$

The function Par of a string of n bits is the xor of all the bits of the string:

$$\text{Par}(a_0||a_1|| \cdot ||a_n) = a_0 \oplus a_1 \oplus \cdots \oplus a_n$$

1. Given m, explain how to take advantage of the property to find a second preimage. Approximately compute the probability of finding a second preimage by evaluating the function at most 2^{64} times.
2. Given m, explain how to take advantage of the property to find acollision. Approximately compute the probability of finding a collision by evaluating the function at most 2^{64} times.

Solution

1. If we choose m' such that $\text{Par}(m') = \text{Par}(m)$ the probability of finding $h(m) = h(m')$ is
$$\frac{2^{64}}{2^{127}} \simeq 10^{-19}$$
becomes $1/2^{n-1}$. The complexity is the same of an attack to the second preimage on a hash function of 127 bits, which is $O(2^{127})$.

2. We choose 2^{64} pairs with the same parity, therefore the probability of collision is:
$$1 - \exp\left(-\frac{(2^{64})^2}{2 \cdot 2^{127}}\right) = 1 - \frac{1}{e} = 0,63$$

Exercise 4.4 Consider the following protocol for playing the game ”heads or tails”. The function $h(m)$ is a hash function of128 bits.

1. Alice chooses $x = 0 =$ ”head” or $x = 1 =$ ”tail”. She chooses a key k and computes $y = h(k||x)$. She sends y a Bob.
2. Bob makes his choice and communicates it to Alice
3. Alice reveals the key k and her choice x
4. Bob computes $y = h(k||x)$ and verifies that it is equal to the value obtained at step 1.

Questions:

1. Find how Alice can choose the value of x after knowing Bob’s choice.
2. Say which cryptographic property must $h(m)$ have to make Alice’s attack difficult.
3. Compute the probability that Alice’s attack is successful knowing that Alice can make at most 2^{60} attempts.

Solution

1. Alice must find $k_1 \neq k_2$ such that $h(k_1||0) = h(k_2||1)$. At step 3 she communicates k_1 or k_2 according to Bob’s choice.
2. $h(m)$ must be *strongly collision free*.
3. Using the random oracle model we obtain:
$$\Pr\{\text{Collision}\} = 1 - \exp\left(-\frac{(2^{60})^2}{2 \cdot 2^{128}}\right) = 1 - \exp(-2^{-9}) = 0,2\%$$

Exercise 4.5 Let be $h(m) = m^2 \bmod p$ with $p = 541$.

1. Compute how many different values $h(m)$ can assume.
2. Find a second preimage for $m = 45$.
3. Find a collision.
4. Assuming $p = 547$, find a preimage for $h(m) = 78$.

Exercise 4.6 The Police sequestered to Bob a hard disk. As guarantee for Bob, at the moment of the sequestration a hash of the content of the disc was generated using a cryptographic hash function h, which generates a hash of 160 bits. Let m be the content of the disc and $h(m)$ the hash computed at the moment of the sequestration. During the process, Bob affirms that the content of the disc has been altered after the sequestration, even if the resulting hash is the same. Bob says that the function h is not collision resistant because the hash is too short. Let m' be the content of the disc at the moment of the process.

1. Compute the probability to find a collision for the function h using at most 2^{128} attempts.

2. Assuming you are competent, the judge asks you the following question: «Supposing it is technologically possible to make at most 2^{128} attempts, which is the probability that $m' \neq m$?». Assume the use of the random oracle model.

3. Let be $h(m) = \alpha^m \bmod p$, with p prime of 160 bits and cryptographically secure and α a primitive root of $\mathbb{Z}_p$. Whoa would you answer?

4. Let $h(m)$ be defined as follows:

$$\begin{aligned} m &= m_1 \| m_2 \| \dots \| m_n \qquad \text{where the strings } m_i \text{ have a length of 160 bits} \\ h(m) &= m_1 \oplus m_2 \oplus \cdots \oplus m_n \end{aligned}$$

   What would you answer?

5. Let $h(m)$ be defined as follows:

$$\begin{aligned} m &= m_1 \| m_2 \| \dots \| m_n \qquad \text{where the strings } m_i \text{ have a length of 160 bits} \\ h(m) &= E(m_1) \oplus E(m_2) \oplus \cdots \oplus E(m_n) \end{aligned}$$

   Where the function E is a cryptographically secure hash function. What would you answer?

Solution

1. Let the number of attempts be $r = 2^{128}$ and the number of possible hashes be $N = 2^{160}$. We have:

$$\Pr[\text{collision}] \simeq 1 - \exp\left(-\frac{r^2}{2N}\right) = 1 - \exp\left(-2^{95}\right) \simeq 1$$

2. Finding $m' \neq m$ is an attack to the second preimage. The probability of success is:

$$\Pr[\text{success}] = 1 - \left(1 - \frac{1}{N}\right)^r \simeq \frac{r}{N} = 2^{-32} \simeq 2,3 \cdot 10^{-10}$$

3. A second preimage must satisfy the equation $m' \equiv m \pmod{p-1}$. A second preimage can easily be generated, since it is sufficient that it is congruent to m modulo $p-1$.

4. The function is not resistant to the second preimage, which can be obtained in various ways. For example one can add two identical blocks or permute the original blocks.

5. As the previous answer.

Exercise 4.7 We are looking for the preimage of a hash function $h(\cdot)$ which can generate hashes of arbitrary length. Let us assume that an ASIC can check $2e^7$ messages per second.

1. Assuming to use 64-bit hashes, how much time does it take on average to find a preimage?

2. Which is the probability to find a preimage in 1 hour?

3. According to Moore's law, a processor doubles the number of computations it can perform every 18 months. How long must the hash be in 8 years in order to force an attacker to use the same amount of time computed at question 1?

4. The Groover's quantistic algorithm makes $\sqrt{N}$ operations to search in a list of N elements. Let us assume that in 2020 a quantistic computer performs$2e^7$ operations per second. How long must the hash be in order to force an attacker to use the same amount of time computed at question 1?

Solution 1. $\frac{2^{64}}{2 \cdot 2e^7}$

2. $p \simeq 1 - \exp\left(\frac{-4e^{14}}{2 \cdot 2^{64}}\right)$.

3. In 8 years there are $8 \cdot 12/18$ doubles, which correspond to an increase of complexity of 5.33 bits. Therefore, the length must be 70 bits.

4. We need twice the number of bits, which is $2 \cdot 64 = 128$.

Exercise 4.8 Consider the cryptographic has function $h \colon \{0,1\}^* \to \{0,1\}^l$ and the function

$$g(0) = h(IV)$$
$$g(n) = h(g(n-1))$$

, where IV is known only to Alice. Let us assume that Bob knows that, for a given $0 < m < 128$, $g(m) = k$, and that $g(128) = T$.

1. How can Bob compute $m+1$? How many evaluations of h are necessary?

2. How can Bob compute $m-1$? How many evaluations of h are necessary?

3. Which characteristics must h have in order to make finding $m-1$ computationally hard?

4. Prove that the succession $g(n)$ with $n \geq 0$ is periodic. Compute an upper bound of the period.

Solution

1. Bob computes $g(k)$, $g(g(k)), \ldots$ until he finds T. Let us assume he has evaluated the function ζ times. Then, $m = 128 - \zeta$. Therefore, 128-m evaluations are necessary.

2. See point 1.

3. Finding $m-1$ must require a number of operations higher than a given threshold (e.g. 2^{80}).

4. $g(n)$ can assume at most 2^l distinct values. As soon as, for a given n, it happens that $g(n) = g(m)$ with $m < n$, the functions repeats its outputs with period $m - n$.

Exercise 4.9 Consider the hash functions h_1, h_2 and the has function:

$$g(x) = h_1(x)||h_2(x)$$

Prove that, if $g(x)$ is not collision resistant, then h_1 and h_2 are not collision resistant as well.

Solution If $g(x)$ is not collision resistant, then we know x_1, x_2 such that $g(x_1) = g(x_2)$. But then $h_1(x_1) = h_1(x_2)$, thus h is not collision resistant. The same considerations can be done for h_2.

Exercise 4.10 An attacker wants to find a preimage for a hash function $h(\cdot)$, which outputs 128-bit hashes. Ad ad-hoc ASIC can calculate 2×10^7 hashes per second.

1. What is the probability of finding a preimage in 1 month?

   Solution In one month the ASIC can calculate $q = 5.18 \times 10^{13}$ hashes, which is much less than $2^{128} = 3.4 \times 10^{34}$. So the probability is 1.5×10^{-25}.

2. What is the probability of finding a collision in 1 month?

   Solution

$$\frac{q^2}{2^{129}} = 3.95 \times 10^{-12}$$

3. According to Moore's law, the computational power of processors doubles every 18 months. What will likely be the probability of finding a collision six years from now?

   Solution Computational power will double 4 times, so $q' = 4q = 2 \times 10^{14}$.

   A collision is found with probability

   $$\frac{16q^2}{2^{129}} = 6.3 \times 10^{-11}$$

4. How long must be the output of the hash function to have the same probability as in 2014?

   Solution We need to solve:

   $$\frac{q^2}{2^{129}} = \frac{4^2q^2}{2^{129+x}}$$

   which yields $2^x = 4^2$, thus $x = 4$.

5. Describe how hash functions are used in digital signatures. Describe an attack scenario in which knowledge of a collision can break the security of the digital signature.

   Solution check the theory

6. Suppose that you have a deterministic algorithm that can find a second preimage in 1 hour with probability 2^{-8}. Tell whether that algorithm is useful to to solve the preimage and the collision problems and provide the new probabilities.

   Solution It is not useful for finding a preimage.

   We can use it for finding a collision. Consider the following algorithm.

   ```
   Find collision

   Choose $q$ random inputs $m_1, \dots, m_q$
   for $i = 1 \to q$ do
       Find a second preimage $m'_i$
       if second preimage found then
           Output $(m_i, m'_i)$.
       end if
   end for
   ```

   In 1 month, the algorithm can evaluate $q = 720$ inputs. The probability that the algorithm outputs a collision are:

   $$1 - \left(1 - 2^{-8}\right)^{720} = 0.94$$

which is much better.

Exercise 4.11 Consider the compression function $h_1 \colon \{0,1\}^{2m} \to \{0,1\}^m$. Consider users authenticating to a system using passwords. There are $N = 100{,}000$ users in the system. User passwords are stored in the system's database as $h_1(s_i \| pw_i)$, where s_i is the salt and pw_i is the ith user password truncated or padded as necessary. With $m = 64$, A single evaluation of h_1 requires 0.2 ms.

1. An adversary obtains access to the user database. Describe the attack scenario and attacker capabilities and explain which ones of the following properties h_1 must have to provide password security: preimage resistant, second-preimage resistant, and collision resistant.

   Solution Only preimage resistant.

2. One third of the users choose passwords included in a popular dictionary of 1,000,000 passwords. How many passwords is the attacker able to find in one week? What happens if the number of users in the system increases?

   Solution In a week, the attacker can perform $q = 3.024 \times 10^9$ evaluations of the hash function, which means full evaluation of 3024 users. On average, the attacker finds the password of 1008 users. Nothing happens if there are more users.

3. Explain how the number of found passwords changes if the size of the salt s_i is only 3 bits.

   Solution The attacker can precompute the salted passwords, in fact the resulting dictionary is only 8,000,000 items.

   When the attacker gains access to the dataase, he/she can find the passwords of $N/3 = 33{,}333$ users in a very short time.

4. Design a password stretching system that allows the attacker to find fewer than 1 password in one week. What is the drawback of the new system? Is that acceptable?

   Solution Hashing must be 1008 times slower, thus the access time becomes 0.2 s, which may still be acceptable.

5. Let $h_2 \colon \{0,1\}^{2m} \to \{0,1\}^{2m}$ defined as follows. Let x_1 the first m bits of x, let x_2 the last m bits of x.

   $$h_2(x) = h_1(x_1 \| x_2) \| h_1(x_2 \| x_1)$$

   Prove that, if h_1 is collision-resistant, then h_2 is collision resistant.

Solution Suppose that we have a collision for h_2. Thus, we have $x \neq x'$ s.t. $h_2(x) = h_2(x')$.

But then, $h_1(x) = h_1(x')$ and we have a collision in h_1, which contradicts the hypothesis.

Exercise 4.12 The Police seized Bob's hard disk and gave him a digest of the disk $h(m)$, where h is a 80-bit hash function and m is the disk content. During the trial, the Police shows to the judge the content of the disk m'. Bob says that the content has been altered and that $m \neq m'$ even though $h(m) = h(m')$.

1. What property should h have in order to make it difficult for an attacker to achieve what Bob says?

   Solution Second preimage resistance.

2. Suppose that an attacker can make one evaluation of a hash function in 1 µs. What is the probability that an attacker modified Bob's hard disk after one year?

   Solution In one year the attacker can make $q = 3.15 \times 10^{13}$ attempts. The success probability is $q/2^{80} = 2.6 \times 10^{-11}$.

3. Bob shows a hard disk content m'' such that $h(m'') = h(m)$. How could have Bob generated m''? How long did it take to Bob to find m''?

   Solution

   Bob could have found a collision. In addition to the found disk, Bob had additional disk with the same digest. Finding a collision with a reasonably high probability (e.g. 1/2 requires q attempts with:

   $$\exp(q^2 2^{-81}) = 1/2$$

   Hence $q = 1.3 \times 10^{12}$, which means 15 years.

4. Suppose that $h(m) = \alpha^m \bmod p$, with p an 80-bit prime and α a primitive root of $\mathbb{Z}_p$. How can an attacker modify Bob's hard disk after it has been seized?

   Solution Una seconda preimmagine deve rispettare l'equazione $m' \equiv m \pmod{p-1}$. Si può facilmente generare una seconda preimmagine, è sufficiente che sia congruente a m modulo $p - 1$.

5. Suppose that $h(m)$ is defined as:

   $$\begin{aligned} m &= m_1 \| m_2 \| \dots \| m_n \qquad \text{where } m_i \text{ are 80-bit blocks} \\ h(m) &= m_1 \oplus m_2 \oplus \cdots \oplus m_n \end{aligned}$$

How can an attacker can easily modify Bob's hard disk after it has been seized?

Solution Le funzioni non sono resistenti alla seconda preimmagine, che si può ottenere in vari modi. Per esempio aggiungendo due blocchi identici oppure permutando i blocchi originali.

6. Suppose that $h(m)$ is defined as:

$$\begin{aligned} m &= m_1 \| m_2 \| \dots \| m_n \qquad \text{where } m_i \text{ are 80-bit blocks} \\ h(m) &= E(m_1) \oplus E(m_2) \oplus \dots \oplus E(m_n) \end{aligned}$$

where $E(m)$ is a secure block cipher. How can an attacker easily modify Bob's hard disk after it has been seized?

Exercise 4.13 Consider the cryptographically secure hash function $h\colon \{0,1\}^* \to \{0,1\}^l$ and the function

$$\begin{aligned} g(0) &= h(IV) \\ g(n) &= h(g(n-1)) \end{aligned}$$

with IV known only to Alice.

Suppose that Bob knows that, for a given $0 < m < 128$, then $g(m) = k$ and $g(128) = T$ for some known values of k and T.

1. Tell how Bob can calculate $m+1$ and how many evaluations of h are necessary.

   Solution Bob calcola $g(k)$, $g(g(k))$, ecc. finché non trova T. Supponiamo che abbia valutato la funzione ζ volte. A questo punto $m = 128 - \zeta$. Occorrono 128-m valutazioni.

2. Tell how Bob can calculate $m-1$ and how many evaluations of h are necessary.

   Solution come sopra

3. What characteristics must h have so that the calculation of $m-1$ is computationally hard?

   Solution Deve richiedere un numero di operazioni maggiore di una qualche soglia (ad es. 2^{80}).

4. Prove that the sequence $g(n)$ with $n \geq 0$ is periodic. Give an upper bound to the period.

   Solution $g(n)$ può assumere al più 2^l distinti valori. Appena per un qualche n accade che $g(n) = g(m)$ con $m < n$, la funzione comincia a ripetersi con periodo $m - n$.

4.1 Finite fields

Exercise 4.14 Consider the polynomial function $f(x)$ with real coefficients and with coefficient of maximum degree equal to 1. Two polynomials $f(x)$ and $g(x)$ are coprime if $\gcd(f(x), g(x)) = 1$.

Find $d(x) = \gcd(x^4 + x^2 + 1, x^2 + 1)$ and find the polynomials $s(x)$ and $t(x)$ such that $d(x) = s(x)f(x) + t(x)g(x)$.

Solution .

	$s_j(x)$	$t_j(x)$
$-$	1	0
$-$	0	1
$x^4 + x^2 + 1 = x^2 \cdot (x^2 + 1) + 1$	1	$-x^2$
$x^2 + 1 = (x^2 + 1) \cdot 1 + 0$	$-(x^2 + 1)$	$x^4 + x^2 + 1$

$$d(x) = 1$$
$$s(x) = 1$$
$$t(x) = -x^2$$

Indeed, $1 = 1 \cdot (x^4 + x^2 + 1) - x^2 \cdot (x^2 + 1)$.

Exercise 4.15 Compute $d(x) = \gcd(x^4 - 4x^3 + 6x^2 - 4x + 1, x^3 - x^2 + x - 1)$.

Solution .

	$s_j(x)$	$t_j(x)$
$--$	1	0
$--$	0	1
$x^4 - 4x^3 + 6x^2 - 4x + 1 =$ $= (x - 3)(x^3 - x^2 + x - 1) + (2x^2 - 2)$	1	$-(x - 3)$
$x^3 - x^2 + x - 1 = (\frac{1}{2}x - \frac{1}{2})(2x^2 - 2) + (2x - 2)$	$-\frac{1}{2}(x - 1)$	$\frac{1}{2}(x^2 - 4x + 5)$
$2x^2 - 2 = (x + 1)(2x - 2) + 0$	$\frac{1}{2}(x^2 + 1)$	$-\frac{1}{2}(x^3 - 3x^2 + 3x + 1)$

The last non null remainder is $2x - 2 = 2(x - 1)$. We consider as gcd the monic polynomial $x - 1$.

$$d(x) = x - 1$$
$$u(x) = -\frac{1}{4}x + \frac{1}{4}$$
$$v(x) = \frac{1}{4}x^2 - x + \frac{5}{4}$$

Exercise 4.16 If a polynomial $f(x)$ has multiple roots, these are also roots of $\gcd(f(x), f'(x))$.

Find the multiple root of the polynomial $x^4 - 2x^3 - x^2 + 2x + 1$.

Solution

$$f'(x) = 4x^3 - 6x^2 - 2x + 2$$

$$\begin{aligned} x^4 - 2x^3 - x^2 + 2x + 1 &= \left(\frac{1}{4}x - \frac{1}{8}\right)\left(4x^3 - 6x^2 - 2x + 2\right) + \left(-\frac{5}{4}x^2 + \frac{5}{4}x + \frac{5}{4}\right) \\ 4x^3 - 6x^2 - 2x + 2 &= \left(-\frac{16}{5}x + \frac{8}{5}\right)\left(-\frac{5}{4}x^2 + \frac{5}{4}x + \frac{5}{4}\right) + 0 \end{aligned}$$

Therefore, the multiple root is $x^2 - x - 1$. Indeed,

$$x^4 - 2x^3 - x^2 + 2x + 1 = (x^2 - x - 1)^2$$

Exercise 4.17 Let $f(x) = x^4 + x^3 + x^2 + 1$ and $g(x) = x^3 + 1$ be polynomials with coefficients in $\mathbb{Z}_2$. Find $d(x) = \gcd(f(x), g(x))$ and express it as linear combination of f and g.

Solution

$r(x)$		$s(x)$	$t(x)$
$x^4 + x^3 + x^2 + 1$		1	0
$x^3 + 1$		0	1
$x^2 + x$	$x^4 + x^3 + x^2 + 1 = (x + 1)(x^3 + 1) + x^2 + x$	1	$-(x + 1)$
$x + 1$	$x^3 + 1 = (x + 1)(x^2 + x) + (x + 1)$	$-(x + 1)$	x^2
0	$x^2 + x = x(x + 1) + 0$	$\cdots$	$\cdots$

Remember the correct method to compute the division of polynomials in a modulus 2 arithmetic ($1 + 1 = 0$):

$$\begin{array}{ccccc|l} x^4 & +x^3 & +x^2 & & +1 & x^3+1 \\ \cline{6-6} x^4 & & & x & & x+1 \\ \cline{1-5} & x^3 & +x^2 & +x & +1 & \\ & x^3 & & & 1 & \\ \cline{1-5} & & x^2 & +x & & \end{array}$$

$$\begin{array}{cccc|l} x^3 & & & +1 & x^2+x \\ \cline{5-5} x^3 & +x^2 & & & x+1 \\ \cline{1-4} & x^2 & & 1 & \\ & x^2 & +x & & \\ \cline{1-4} & & x & +1 & \end{array}$$

The gcd is $d(x) = x + 1$ and can be written as:

$$x + 1 = (x + 1)(x^4 + x^3 + x^2 + 1) + x^2(x^3 + 1)$$

Exercise 4.18 Find the multiple factor in $f(x) = x^4 - x^2 + 1 \in \mathbb{Z}_3[x]$.

Solution If a factor is multiple, it is contained in both $f(x)$ and $f'(x)$, therefore it is contained also in $\gcd(f, f')$.

Here we have $f'(x) = 4x^3 - 2x = x^3 + x$.

$r(x)$	$q(x)$	
$x^4 - x^2 + 1$	$--$	
$x^3 + x$	x	
$x^2 + 1$	x	
0	$--$	
$\dots$		

Therefore $\gcd(f, f') = x^2 + 1$.

Exercise 4.19 Prove that the polynomial $r(x) = x^2 + 1 \in \mathbb{Z}_3[x]$ is irreducible. find the multiplicative inverse of $(2x + 1) \bmod r(x)$.

Solution If $r(x)$ were irreducible, it would be divisible by a polynomial $d(x)$ of lower degree (degree one). The polynomials of degree 1 are:

$$\begin{aligned} &x \\ &x+1 \\ &x+2 \\ 2x &= 2 \cdot x \\ 2x+1 &= 2(x+2) \\ 2x+2 &= 2(x+1) \end{aligned}$$

Excluding the polynomials that can be obtained through multiplication by a constant, the remaining are: $x, x+1, x+2$:

$$x^2 + 1 = x \cdot x + 1 \tag{4.2}$$
$$x^2 + 1 = (x+2)(x+1) + 2 \tag{4.3}$$
$$\tag{4.4}$$

Therefore $(x^2 + 1)$ is not reducible.

In order to compute $(2x+1)^{-1}$ we use the Euclidean algorithm:

$r(x)$	$q(x)$		$s(x)$	$t(x)$
$x^2 + 1$	$--$		1	0
$2x + 1$	$2x + 2$		0	1
2	$\cdots$	$x^2 + 1 = (2x+2)(2x+1) + 2$	1	$x + 1$
$\dots$				

We report the division $\frac{x^2+1}{2x+1}$. Remember that $x^2 = -1 = 2$.

$$\begin{array}{ccc|c} x^2 & & +1 & -x+1 \\ \cline{4-4} x^2 & -x & & -x-1 \\ \cline{1-3} & x & +1 & \\ & x & -1 & \\ \cline{1-3} & & 2 & \end{array}$$

Therefore:

$$(2x+1)(x+1) \bmod (x^2+1) = 2$$

and the inverse of $2x+1$ is $(-x-1) \equiv (2x+2)$.

Alternatively, we can compute the inverse as $(2x+1)^{9-2} \bmod (x^2+1)$ using Square-and-Multiply.

1	$1^2 \times (2x+1) = (2x+1)$
1	$(2x+1)^2 \times (2x+1) = (-x+1)^3 = -x^3+1 = -x(-1)+1 = x+1$
1	$(x+1)^2 \times (2x+1) = (x^2+2x+1)(2x+1) = 2x(2x+1) = 4x^2+x = 2x+2$

Exercise 4.20 (Hill Cypher on $GF(4)$) The binary plaintext $P =$ `011000110100` is ciphered using the Hill cipher on $GF(4)$. As usual, the elements of the field are the remainders of the residue class modulo $r(x) = x^2+x+1$. The Hill cipher uses as key a 2×2 matrix K.

1. Enumerate the requisites that K must satisfy.

2. Cipher the message P assuming

$$K = \begin{pmatrix} 0 & 1 \\ x+1 & x+1 \end{pmatrix}$$

3. Decipher the ciphertext obtained above.

4. Using the plaintext/ciphertext pair obtained above, apply a *known plaintext* attack and compute the key K.

Solution 1. The key K must satisfy the following requirements:

$$\begin{cases} \det(K) \neq 0 \\ \gcd(\det(K), x^2+x+1) = 1 \end{cases}$$

Note that, since we are considering a finite field, condition 2 is implied by 1.

2. We first convert the binary plaintext in elements of $GF(4)$. Since the elements are $\{0, 1, x, x+1\}$ the easiest way it so convert every bit pair in a polynomial. Therefore, the message becomes:

$$P = \begin{pmatrix} 1 & x & 0 & x+1 & 1 & 0 \end{pmatrix}$$

The key is a 2×2 matrix, therefore the plaintext must be subdivided in vectors P_i of two elements each:

$$\begin{aligned} P_1 &= \begin{pmatrix} 1 & x \end{pmatrix} \\ P_2 &= \begin{pmatrix} 0 & x+1 \end{pmatrix} \\ P_3 &= \begin{pmatrix} 1 & 0 \end{pmatrix} \end{aligned}$$

We then apply the formula:

$$C_i = P_i \cdot M \pmod 4 \tag{4.5}$$

$$\begin{aligned} C_1 &= \begin{pmatrix} 1 & x \end{pmatrix} \begin{pmatrix} 0 & 1 \\ x+1 & x+1 \end{pmatrix} = \begin{pmatrix} x^2+x & x^2+x+1 \end{pmatrix} \\ &= \begin{pmatrix} 1 & 0 \end{pmatrix} \pmod{x^2+x+1} \\ C_2 &= \begin{pmatrix} 0 & x+1 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ x+1 & x+1 \end{pmatrix} = \begin{pmatrix} x^2+2x+1 & x^2+2x+1 \end{pmatrix} \\ &= \begin{pmatrix} x & x \end{pmatrix} \pmod{x^2+x+1} \\ C_3 &= \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ x+1 & x+1 \end{pmatrix} = \begin{pmatrix} 0 & 1 \end{pmatrix} \pmod{x^2+x+1} \end{aligned}$$

The ciphertext C turns out to be:

$$C = \begin{pmatrix} 1 & 0 & x & x & 0 & 1 \end{pmatrix}$$

which, in binary form, becomes $C =$ `010010100001`.

3. We first must compute the inverse matrix of the key:

$$K^{-1} = \frac{1}{-(x+1)} \begin{pmatrix} x+1 & -1 \\ -(x+1) & 0 \end{pmatrix} = x \begin{pmatrix} x+1 & 1 \\ x+1 & 0 \end{pmatrix} = \begin{pmatrix} 1 & x \\ 1 & 0 \end{pmatrix} \pmod{x^2+x+1}$$

To do the calculation, consider that $\frac{1}{x+1} \equiv x$, therefore $x(x+1) = x^2 + x = x + 1 + 1 = x$.

We compute the plaintext:

$$\begin{aligned}
P_1 &= \begin{pmatrix}1 & 0\end{pmatrix}\begin{pmatrix}1 & x\\ 1 & 0\end{pmatrix} = \begin{pmatrix}1 & x\end{pmatrix} \pmod{x^2+x+1}\\
P_2 &= \begin{pmatrix}x & x\end{pmatrix}\begin{pmatrix}1 & x\\ 1 & 0\end{pmatrix} = \begin{pmatrix}2x & x^2\end{pmatrix} = \begin{pmatrix}0 & x+1\end{pmatrix} \pmod{x^2+x+1}\\
P_3 &= \begin{pmatrix}0 & 1\end{pmatrix}\begin{pmatrix}1 & x\\ 1 & 0\end{pmatrix} = \begin{pmatrix}1 & 0\end{pmatrix} \pmod{x^2+x+1}
\end{aligned}$$

Therefore, the plaintext is:

$$P = \begin{pmatrix}1 & x & 0 & x+1 & 1 & 0\end{pmatrix}$$

4. We must impose:

$$\begin{pmatrix}P_i\\ P_j\end{pmatrix} K = \begin{pmatrix}C_i\\ C_j\end{pmatrix}$$

where the rows P_i, C_i e P_j, C_j are two plaintext/ciphertext pairs of length two.

We can obtain the key by solving the equation with respect to the unknown matrix. To solve the equation, the plaintext matrix must be invertible, so we must choose the rows of the matrix appropriately.

Fortunately, the matrix built using P_1 e P_2 is invertible, since:

$$\det\begin{pmatrix}1 & x\\ 0 & x+1\end{pmatrix} = x+1$$

which is non null.

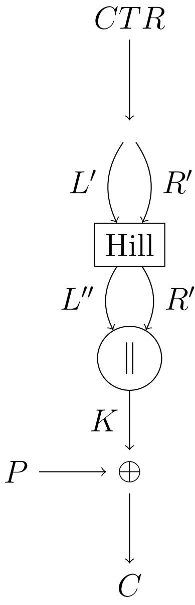
Therefore, we can write:

$$\begin{aligned}
K &= \begin{pmatrix}1 & x\\ 0 & x+1\end{pmatrix}^{-1} \cdot \begin{pmatrix}1 & 0\\ x & x\end{pmatrix} = \frac{1}{x+1}\begin{pmatrix}x+1 & -x\\ 0 & 1\end{pmatrix} \cdot \begin{pmatrix}1 & 0\\ x & x\end{pmatrix}\\
&= x\begin{pmatrix}x+1 & x\\ 0 & 1\end{pmatrix} \cdot \begin{pmatrix}1 & 0\\ x & x\end{pmatrix}\\
&= \begin{pmatrix}1 & x+1\\ 0 & x\end{pmatrix} \cdot \begin{pmatrix}1 & 0\\ x & x\end{pmatrix} = \begin{pmatrix}x^2+x+1 & x^2+x\\ x^2 & x^2\end{pmatrix}\\
&= \begin{pmatrix}0 & 1\\ x+1 & x+1\end{pmatrix} \pmod{x^2+x+1}
\end{aligned}$$

5

Block ciphers and cryptoanalysis

Exercise 5.1 Consider the following cryptosystem operating in *(counter mode)*.



Hill cipher is defined over $\mathbb{Z}_2[x]/g(x)$ with

$$g(x) = x^2 + x + 1$$

and is defined by equation:

$$\begin{pmatrix} L'' & R'' \end{pmatrix} = \begin{pmatrix} L' & R' \end{pmatrix} M \mod g(x)$$

The key is:

$$M = \begin{pmatrix} x & x+1 \\ 0 & 1 \end{pmatrix}$$

The initial value of the counter is $IV = 9$.

1. Verify that the key of the Hill cipher is a valid key.
2. Find the decryption key and verify it.
3. Encrypt the message `1100 0101`.
4. Knowing that for $IV = 3$ the plaintext $P =$ `0100 0010` is encrypted as $C =$ `0101 1011`, find the key.

Solution Siccome $\det(M) = x$ è non nullo e primo con $g(x)$, la chiave è valida.

$$M^{-1} = x^{-1}\begin{pmatrix} 1 & x+1 \\ 0 & x \end{pmatrix} = \begin{pmatrix} x+1 & x \\ 0 & 1 \end{pmatrix}$$

Come verifica calcoliamo $MM^{-1} = I$:

$$\begin{pmatrix} x & x+1 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} x+1 & x \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} x^2+x & x^2+x+1 \\ 0 & 1 \end{pmatrix} \bmod g(x) = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = I$$

Al passo 1 il contatore vale

$$CTR_1 = 9 = \texttt{1001} = \begin{pmatrix} x & 1 \end{pmatrix}$$

In questo calcolo abbiamo espresso il valore del contatore in forma binaria e poi come vettore di polinomi di grado massimo 1.

L'uscita del cifrario di Hill sarà:

$$\begin{pmatrix} x & 1 \end{pmatrix}\begin{pmatrix} x & x+1 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} x+1 & 0 \end{pmatrix}$$

I primi 4 bit del keystream, K_1, saranno:

$$K_1 = \begin{pmatrix} x+1 & 0 \end{pmatrix} = \texttt{1100}$$

Quindi il testo cifrato sarà:

$$C_1 = P_1 \oplus K_1 = 1100 \oplus 1100 = 0000$$

Il calcolo per il secondo blocco è simile ma con $CTR_2 = IV + 1 = 10$.

$$CTR_2 = 10 = \texttt{1010} = \begin{pmatrix} x & x \end{pmatrix}$$

L'uscita del cifrario di Hill sarà:

$$\begin{pmatrix} x & x \end{pmatrix}\begin{pmatrix} x & x+1 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} x+1 & x+1 \end{pmatrix}$$

Quindi

$$C_2 = P_2 \oplus K_2 = 1010$$

Per trovare la chiave, cioè la matrice M, occorre ricordare che il cifrario di Hill non agisce sul testo in chiaro, ma sul contatore e dà in uscita il keystream. Nel nostro caso il keystream è $K = C \oplus P = \texttt{0001 1001}$, mentre i corrispondenti bit del contatore sono $CTR = 3||4 = \texttt{0011 0100}$. I bit del keystream e del contatore devono essere espressi come elementi in $GF(2^2)$ e posti in matrice per ottenere la equazione:

$$
\begin{aligned}
M = CTR^{-1}K &= \begin{pmatrix} 0 & x+1 \\ 1 & 0 \end{pmatrix}^{-1} \begin{pmatrix} 0 & 1 \\ x & 1 \end{pmatrix} = (x+1)^{-1} \begin{pmatrix} 0 & x+1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ x & 1 \end{pmatrix} \\
&= x \begin{pmatrix} 1 & x+1 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & x \end{pmatrix}
\end{aligned}
$$

Exercise 5.2 Consider the block cipher E_k, which uses blocks of 4 bits nd keys of 4 bits implemented as follows:

$$E_k(x) = (10^k \bmod 17) + x \bmod 16$$

Consider a message M made by N blocks of 4 bits each: $m_1, m_2, \dots, m_N$. Consider also the hash functions $h(M)$ and $g(M)$ obtained as follows:

$$
\begin{aligned}
f_0 &= 0 & z_0 &= 0 \\
f_i &= E_0(f_{i-1} \oplus m_i) & z_i &= E_{m_i}(z_{i-1}) \\
h(M) &= f_N & g(M) &= z_N
\end{aligned}
$$

a) Write the equations which define a CBC-MAC obtained from the block cipher E_k and compute the MAC of the binary message `01000100`.

b) Write the equations which define a HMAC obtained from the function g.

c) Considering h and g, find a preimage of the number 3.

d) Considering h and g, find a second preimage for the number 3.

Solution a)

$$
\begin{aligned}
c_0 &= IV \quad \text{usually IV=0} \\
c_1 &= E_k(c_{i-1} \oplus m_1) \\
MAC_k(M) &= c_N
\end{aligned}
$$

Let $10^k \bmod 17 = \xi$.

$$MAC_k(01000100) = E_k(E_k(4) \oplus 4) == (\xi + (((\xi + 4) \bmod 16) \oplus 4) \bmod 16).$$

b)

$$HMAC_k(M) = g(k \oplus \text{opad} \| g(k \oplus \text{ipad} \| M))$$

c) We are looking for M such that $h(M) = 3$ (or $g(M) = 3$). Let us assume that M is composed by only one block m_1. So, we have:

$$h(m_1) = E_0(m_1) = 1 + x \bmod 16 = 3 \qquad\qquad m_1 = 2$$
$$g(m_1) = E_{m_1}(0) = 10^{m_1} \bmod 17 \bmod 16 = 3$$

j, k	10^j	$3 \cdot 10^{-5k}$
0	1	3
1	10	9
2	15	10
3	14	13
4	4	5

Therefore, $10^{1+10} = 3$, and thus $m_1 = 11$.

d) We are looking for M such that $h(M) = H(3)$ (or $g(M) = g(3)$). Let us assume that M is composed by two blocks. For h, we obtain:

$$h(m_1 \| m_2) = E_0((E_0(m_1) \oplus m_2) = E_0(3)$$
$$E_0(m_1) \oplus m_2 = 3$$

Any pair that satisfied the equation above will do, e.g., $m_1 = 15$ and $m_2 = 3$.

For g, we obtain:

$$g(m_1 \| m_2) = E_{m_2}(E_{m_1}(0)) = E_3(0)$$

by choosing $m_2 = 3$, we obtain

$$E_{m_1}(0) = 0$$
$$10^{m_1} \bmod 17 = 16 = -1$$

This is a simple DLP problem: $m_1 = 8$.

Exercise 5.3 Consider the following cipher, where P e C are a plaintext block and a ciphertext block respectively. Both blocks are 3-bits-long. The key K is 3-bits-long

$$C = S(P \oplus K) \oplus K$$

The substitution function $B = S(A)$ is defined as the following expression, where A e B are interpreted as polynomials in $\mathbb{Z}_2[x]/g(x)$.

$$B = S(A) = A^{-1} \bmod g(x)$$
$$g(x) = x^3 + x + 1$$

If $A = 0$, then we assume $B = 0$. Note that $g(x)$ is irreducible.

1. Encrypt the message $P = 7$ using the $K = 5$.

2. Tabulate the function S. Knowing that $A = a_0|a_1|a_2$ e $B = b_0|b_1|b_2$, compute the probability to satisfy the following linear equation:

$$a_0 \oplus a_1 \oplus b_1 = 0 \tag{5.1}$$

Solution

$$\begin{aligned} B &= S(2) = S(x) = x^2 + 1 = 5 \\ C &= 5 \oplus 5 = 0 \end{aligned}$$

A	$B = S(A)$	$a_0 \oplus a_1 \oplus b_1$
000	000	0
001	001	0
010	101	1
011	110	0
100	111	0
101	010	0
110	011	1
111	100	0

Equation (5.1) is satisfied with probability 3/4.

Exercise 5.4 A message m is is divided in blocks of $l = 4$ bit, named m_1, m_2, ..., m_N and encrypted with a block cipher $E(k, x)$ (k is the encryption key (128 bit) and x is the plaintext (128 bit). Assume passive attacks only and that the key k is used for multiple messages.

1. The message is then encrypted as follows:

$$\begin{aligned} b_0 &= 0 \\ b_i &= E(k_1, b_{i-1} \oplus m_i) \quad 1 \leq i \leq N \\ c &= b_1 \| \dots \| b_N \end{aligned}$$

   where c is the cyphertext. Show an attack that breaks semantic security of this cipher and propose a modification to the encryption scheme.

   Solution

2. The message is then encrypted as follows:

$$
\begin{aligned}
b_i &= E(k_1, i) \oplus m_i \quad 1 \leq i \leq N \\
c &= b_1 \| \ldots \| b_N
\end{aligned}
$$

   where c is the cyphertext. Show an attack that breaks semantic security of this cipher and propose a modification to the encryption scheme.

3. Consider now an active attacker. Is the cipher secure against active attackers? What is needed?

6

Public Key Cryptography

6.1 RSA

Exercise 6.1 (Ciphering) Alice uses the RSA algorithm to receive messages from Bob. Alice chooses two prime numbers $p = 13$ e $q = 23$. She also chooses a ciphering exponent $e = 35$.

Alice publishes the product $N = pq = 299$ and the ciphering exponent, but she keeps secret the two numbers p e q.

Questions:

1. verify that the exponent e satisfies the requirements of the RSA algorithm;
2. compute the deciphering exponent d;
3. cipher the number $P = 15$ and verify that it can be correctly deciphered using the computed exponent d.

Solution We first compute $\phi(N) = (p - 1)(q - 1) = 264$. The exponent e must be coprime with $\phi(N)$.

r	q	s	t
264	–	1	0
35	7	0	1
19	1	1	-7
16	1	-1	8
3	5	2	-15
1	3	-11	83

Hence:

$$\gcd(35, 264) = 1 = -11 \times 264 + 83 \times 35$$

The exponent e is valid, the deciphering exponent turns out to be:

$$\begin{aligned} d &= e^{-1} \bmod \phi(N) \\ d &= 35^{-1} \bmod 264 \\ d &= 83 \end{aligned}$$

The exponentiation can be done with the Square-&-Multiply technique. Given $P = 15$, we can compute:

$$C = P^e \bmod N = 15^{35} \bmod 299 = 189$$

$$\begin{array}{c|r} 1 & 15 \\ 0 & 15^2 = 225 \pmod{299} \\ 0 & 225^2 = 94 \pmod{299} \\ 0 & 94^2 = 165 \pmod{299} \\ 1 & 165^2 \times 15 = 240 \pmod{299} \\ 1 & 240^2 \times 15 = 189 \pmod{299} \end{array}$$

Deciphering, we obtain:

$$P = C^d \bmod N = 189^{83} \bmod 299 = 15$$

Exercise 6.2 Alice uses texbook RSA algorithm to receive messages from Bob. She publishes $N = 385$ and $e = 7$.

Bob needs to send a one-digit id; computes $c = 262$ and sends it to Alice. The message must be protected from an eavesdropper, Eve, who can perform at most 20 exponentiations.

Ron suggests that textbook RSA is not secure and suggests to prepend a random number to the message. (This is similar to the PKCS#1 v. 1.5 RSA modification).

Questions:

1. Show that textbook RSA is not semantic secure.

2. Show how Eve can decrypt the message c without knowing the private key d. Decrypt c.

3. Write the modified RSA algorithm as suggested by Ron.

4. Calculate the probability that Eve can decrypt the message after the modifications by Ron. Assume that RSA encryption is all-or-nothing.

5. Assuming that Alice uses a 1024-bit modulus n and that Eve can perform up to 2^{80} exponentiations, calculate the probability that Eve can decrypt the message, whether Ron's modification is necessary and how long must be the random prefix in the following cases:

   (a) Bob sends random 500-bit messages

   (b) Bob sends messages out of an alphabet of two characters

Solution Adversary sends m_0, m_1. If it receives $m_0^e \bmod N$, then $b' = 0$. Otherwise it is 1.

Eve tries all messages starting from 0.

m	c
0	0
1	1
2	128
3	262

So $m = 3$.

Bob:

- Choose r randomly in $\mathbb{Z}_{38}$.
- Calculate $x = r \| m$.
- If $x \geq 385$ choose a new r.
- Calculate $c = x \bmod N$.

Alice:

- Calculate $x = c \bmod N$.
- Calculate $m = x \bmod 10$.

Eve finds the correct decryption if she finds both the correct m and the correct r. This happens with probability $1/N$.

In case of random messages, Eve can guess the correct message with probability $2^{-500+80}$. Ron's modifications are not necessary.

In case of two messages, Eve can guess the correct message with probability 1. Ron's modifications are necessary. With an 80-bit prefix, Eve can guess with probability $2^{80}/2^{81}$. With a 160-bit prefix, Eve can guess with probability $2^{80}/2^{161} = 2^{-81}$.

Exercise 6.3 (Digital signature with RSA) Alice publishes the following data: $N = pq = 221$ and $e = 13$. Bob receives the message $P = 65$ and the corresponding digital signature $F = 182$.

Ron suggest using RSA-FDH using the function $h\colon \{0,1\}^* \to \{0,1\}^7$.

1. Verify the signature.
2. Suppose that Oscar can perform 10 hashes or exponentiations. Calculate the probability that Oscar can make a selective forgery of a signature with a pk-only attack.
3. Calculate the probability that Oscar can make an existential forgery of a signature with a pk-only attack.

Solution The signature is valid if $P = F^e \bmod N$. Here we have:

$$F^e \bmod N = 182^{13} \bmod 221 = 65$$

For a selective forgery of message m, Oscar calculates the hash of m. Then he needs to find $F = h(m)^d \bmod N$. Trying randomly, Oscar finds the correct signature with probability $10/N$.

For an existential forgery, Oscar chooses a random F, then $F^e \bmod N$. Note that with probability $1/2$ $F^e \geq 128$ and a new F must be chosen. Then, Oscar needs to find a preimage for F. Trying randomly, Oscar finds it with probability $10/2^7$.

Exercise 6.4 Alice publishes the following RSA public key: $K_{pub} = (e, n) = (93, 17947)$.

1. Cipher the message $m = 7$ using the S & M algorithm.
2. Using Fermat's factorization method, find p, q.
3. Using the $p - 1$ factorization algorithm, find p, q. (Use $a = 100$ and $B = 3$, $B = 4$).
4. Verify that n is not prime using Miller-Rabin's algorithm.
5. Compute the private key (d, n).

Solution The ciphertext is $c = m^e \bmod n = 7^{93} \bmod 17947$. To use S & M we must start from the binary representation of $93 = 1011101$.

1	7
0	$7^2 = 49$
1	$49^2 \times 7 = 16807 = -1140$
1	$1140^2 \times 7 = 16018 = -1929$
1	$1929^2 \times 7 = 6190$
0	$6190^2 = 17202 = -745$
1	$745^2 = 8623$

Therefore $c = 8623$.

Fermat's factorization For Fermat's factorization we start considering that:

$$t = \frac{p+q}{2} \simeq \sqrt{n} = 133,96$$

Supposing $\hat{t} = 134$, we have

$$\hat{s}^2 = \hat{t}^2 - n = 134^2 - 17947 = 9 = 3^2$$

We write the equations:

$$\begin{aligned} t &= \frac{p+q}{2} = 134 \\ s &= \frac{p-q}{2} = 3 \end{aligned}$$

from which we obtain $p = 137$, $q = 131$.

$p - 1$ **factorization** We start from $a = 100$ and compute the succession $b_1, b_2, \ldots$:

$$\begin{aligned} b_1 &= a = 100 \\ b_2 &= b_1^2 \mod n = 10000 \\ b_3 &= b_2^3 \mod n = 15754 \\ b_4 &= b_3^4 \mod n = 12057 \end{aligned}$$

For b_3 we obtain that $\gcd(b_3 - 1, n) = \gcd(15753, 17947) = 1$, since:

$$\gcd(15753, 17947) = \gcd(17947 \mod 15753, 17947) = \gcd(2194, 17947) = \cdots = 1$$

On the other hand, for b_4 we obtain $\gcd(b_4 - 1, n) = 137$.

Miller-Rabin We chose $a = 2$.

$$n - 1 = 17946 = 2 \times 8973$$

From which it derives that $k = 1$, $m = 8973$. We compute

$$b_0 = a^m \mod n = 2^{8973} \mod n = 3545$$

so the number is composite.

To find the private key it is necessary to calculate

$$d = e^{-1} \mod \phi(n) = e^{-1} \mod (p-1)(q-1) = 93^{-1} \mod 17680 = 12357$$

6.2 Factorization

6.2.1 Fermat's factorization

Exercise 6.5 Alice publishes the following data: $N = 6557$, $e = 131$. Find the deciphering exponent d.

Solution Assuming $p > q$ the following equalities are always true:

$$N = t^2 - s^2 = \left(\frac{p+q}{2}\right)^2 - \left(\frac{p-q}{2}\right)^2$$

Fermat's factorization is efficient if $p \simeq q$. In this case we have $t \simeq \sqrt{n}$ and $s \simeq 0$.

We try with all the integer numbers $t > \sqrt{N}$ and we compute:

$$\widehat{s}^2 = t^2 - N$$

We go on until $\widehat{s}^2$ is not a perfect square.
In this case it must be $t > 80,9$.
We try $t = 81$. For $t = 81$ we have $\widehat{s}^2 = 6561 - 6557 = 4$. Therefore:

$$p + q = 162$$
$$p - q = 4$$

We obtain $p = 83$ e $q = 79$.
The deciphering exponent is:

$$d = e^{-1} \bmod \phi(N) = 131^{-1} \bmod (p-1)(q-1) = 131^{-1} \bmod 6396$$

The inverse of 131 modulo 6396 can be found using the extended Euclidean algorithm:

r	q	s	t
6396	–	1	0
131	48	0	1
108	1	1	-48
23	4	-1	49
16	1	5	-244
7	2	-6	293
2	3	17	-830
1	2	-57	2783
0	–	131	-6396

The inverse of 131 is $d = 2783 \pmod{6396}$.

6.2.2 p − 1 algorithm

Let n be the number to be factorized. We chose an integer number $a > 1$, e.g. $a = 2$ and a limit B.

We compute $b = a^{B!} \bmod n$ and $d = \gcd(b-1, n)$. The result d is a factor of n; the algorithm succeeds and in case $d \neq 1$ or $d \neq n$ we have obtained a non banal factor.

Exercise 6.6 Factorize the number 1001, using the basis $a = 2$.

Solution We chose $B = 3$. Then:

$$\begin{aligned} b &= a^{B!} \bmod n = 2^{3!} \bmod 1001 = 2^6 \bmod 1001 = 64 \\ d &= \gcd(b-1, n) = \gcd(63, 1001) = 7 \end{aligned}$$

Therefore 7 is a factor. Indeed, $1001/7 = 143$. As 143 is not prime, we must go on with the factorization.

We try to increment B.

$$\begin{aligned} b &= 2^{4!} \bmod 143 = 64^4 \bmod 143 = 27 \\ d &= \gcd(26, 143) = 13 \end{aligned}$$

Therefore, 13 is another factor. We conclude that:

$$1001 = 7 \times 11 \times 13$$

6.3 Discrete Logarithm Problem

Exercise 6.7 Consider the following commitment scheme (Pedersen Commitment).

Let G a group with prime order q where DLP is hard. Let g, h random elements in E.

1. Commit. Alice commits to x by choosing a random r with $x, r \in \mathbb{Z}_q$. Alice sends Bob $c = g^x h^r$.
2. Bob stores c and does his computations. Later, Alice reveals x, r.
3. Verification. Bob verifies that $c = g^x h^r$.

Questions

1. Show that Bob learns nothing about x from c.

2. Show that if Alice changes x after the commitment, Alice knows how to solve a DLP.

Solution Let $h = g^a$ for some a. Then $c = g^{x+ar}$. Given x, r, x', there always exists an r' s.t. $g^{x+ar} = g^{x'+ar'}$.

If Alice finds x, r, x', r' s.t. $g^{x+ar} = g^{x'+ar'}$, then

$$x + ar = x' + ar' \bmod q$$

So, $a = (x - x')(r' - r)^{-1} \bmod q$.

Exercise 6.8 Let Alice and Bob use Pedersen Commitments.

A trusted generator chooses the curve

$$E\colon y^2 = x^3 + x + 9 \bmod 13$$

with 14 points, and the base point $P = (8, 10)$. From P, the generator publishes $G = 2 * P$ and $H = 3G$.

Alice publishes (6,7). She later reveals $x = 2, r = 3$.

1. Tell why G and H are good choices for the public parameters and tell the value of q.

2. Verify the commitment.

Solution If $\text{ord}(E) = 14$, then $7G = 0$, and $\text{ord}(G) = 7$. Since $H \in< G >$ and $q = 7$ it also has order 7.

Note that $G = (0, 10)$ and $H = (6, 6)$.

Bob must verify:

$$(6, 7) = 2G + 3H = 2(0, 10) + 3(6, 6) = (4, 8) + (4, 8) = (6, 7)$$

Exercise 6.9 Consider the following authentication scheme (Shnorr). It allows the same client to reuse the same private key on multiple servers.

Let G a group with prime order q. A client (C) holds a private key. $x \in \mathbb{Z}_q$ and publishes $y = g^x$.

Then it executes the following authentication protocol with server S:

1. $C \to S$. Choose random $k \in \mathbb{Z}_q$. Send $r = g^k$.

2. $S \to C$. Send random challenge $e \in \mathbb{Z}_q$.

3. $C \to S$. Calculate $s = k + xe \bmod q$. Send s.

4. The client is authenticated if $g^s = ry^e$.

Questions:

1. Show that, if k is reused, an eavesdropper can calculate x.
2. Suppose that Eve observed a full protocol exchange. Eve tries to authenticate by repeating message 1 and message 3. What is the probability of success?

Solution If k is reused, then Eve knows k, e, s, k, e', s' s.t.

$$s - xe = s' - xe' \bmod q$$

thus $x = (s - s')(e - e')^{-1} \bmod q$.

Eve succeeds only if $s = k + xe = k + xe'$, but this possible only if $e = e'$. This happens with probability $1/q$.

7

Quadratic residues

7.1 Chinese remainder theorem

Exercise 7.1 Find the solutions of the following congruence:

$$3x \equiv 4 \pmod 7$$

Solution Consider that

$$3^{-1} \cdot 3x \equiv 3^{-1} \cdot 4 \pmod 7$$

Using the extended Euclidean theorem we obtain that

$$3 \cdot 5 = 15 = 7 \cdot 2 + 1 \equiv 1 \pmod 7$$

Therefore, the inverse of 3 is 5. We can write that:

$$\begin{aligned} 5 \cdot 3x &\equiv 5 \cdot 4 \pmod 7 \\ x &\equiv 20 \bmod 7 \equiv 6 \pmod 7 \end{aligned}$$

Therefore, the solution is:

$$x = 6 + 7k, \quad \forall k \in \mathbb{Z}$$

Exercise 7.2 Find the solutions of the following system of equations:

$$\begin{cases} x & \equiv 2 \pmod 3 \\ x & \equiv 3 \pmod 5 \\ x & \equiv 4 \pmod{11} \\ x & \equiv 5 \pmod{16} \end{cases}$$

Solution The Chinese remainder theorem guarantees that a solution exists in case $m_i = \{3, 5, 11, 16\}$ are coprime pairs. It is easy to verify that 3, 5 and 11 are prime and that $16 = 2^4$, therefore there are no common factors.

We then define:

$$\begin{aligned}
a_i &= \{2, 3, 4, 5\} \\
m_i &= \{3, 5, 11, 16\} \\
M &= \prod_{i=1}^{4} m_i = 3 \cdot 5 \cdot 11 \cdot 16 = 2640 \\
z_i &= \frac{M}{m_i} \\
y_i &= z_i^{-1} \pmod{m_i} \\
x &= \sum_{i=1}^{4} a_i y_i z_i \bmod M
\end{aligned}$$

In this case:

i	a_i	m_i	z_i	y_i
1	2	3	880	1
2	3	5	528	2
3	4	11	240	5
4	5	16	165	13

Therefore:

$$x = 2 \cdot 1 \cdot 880 + 3 \cdot 2 \cdot 528 + 4 \cdot 5 \cdot 240 + 5 \cdot 13 \cdot 165 \equiv 1973 \pmod{2640}$$

7.1.1 Fast exponentiation with the Chinese Remainder Theorem

Let $n = pq$, where p and q are primes. One way to perform the exponentiation x^d mod n efficiently is to exploit CRT in order to compute two individual exponentiations modulo the two “short” primes p and q rather than the “long” modulus n. This can be done in three steps:

1. we reduce the base element x modulo the two factors p and q of the modulus n:

$$x_p \equiv x \mod p$$

$$x_q \equiv x \mod q$$

2. we perform the following two exponentiations:

$$y_p \equiv x_p^{d_p} \mod p$$

$$y_q \equiv x_q^{d_q} \mod q$$

   where the two new exponents are given by:

$$d_p \equiv d \mod (p-1)$$

$$d_q \equiv d \mod (q-1)$$

   Note that both exponents d_p and d_q are bounded by p and q respectively. The same holds for y_p and y_q.

3. compute the final result as:

$$y \equiv [qc_p]y_p + [pc_q]y_q \mod n$$

   where the coefficients c_p and c_q are computed as:

$$c_p \equiv q^{-1} \mod p$$

$$c_q \equiv p^{-1} \mod q$$

Exercise 7.3 Let the RSA parameters be given by $p = 11$, $q = 13$, $n = pq = 143$, $e = 7$, $d = e^{-1} = 103 \mod 120$. Compute an RSA decryption for the ciphertext $y = 15$ using the CRT.

Solution We have to compute $y^d = 15 \mod 143$. As first step, we compute:

$$y_p \equiv 15 \equiv 4 \mod 11$$

$$y_q \equiv 15 \equiv 2 \mod 13$$

Then we compute the exponents:

$$d_p \equiv 103 \equiv 3 \mod 10$$

$$d_q \equiv 103 \equiv 7 \mod 12$$

and the following exponentiations:

$$x_p \equiv y_p^{d_p} = 4^3 = 64 \equiv 9 \mod 11$$

$$x_q \equiv y_q^{d_q} = 2^7 = 128 \equiv 11 \mod 13$$

Finally, we obtain the coefficients:

$$c_p \equiv 13^{-1} \equiv 2^{-1} \equiv 6 \mod p$$

$$c_q \equiv 11^{-1} \equiv 6 \mod 13$$

The plaintext follows as:

$$x \equiv [qc_p]x_p + [pc_q]x_q \mod n$$

$$x \equiv [13 \cdot 6]9 + [11 \cdot 6]11 \mod 143$$

$$x \equiv 702 + 276 \equiv 1428 \equiv 141 \mod 143$$

7.2 Quadratic residues

Exercise 7.4 Compute $\left(\frac{91}{167}\right)$.

Solution Note that 167 is prime, so we can use the definition of the Legendre symbol:

$$\left(\frac{a}{p}\right) = \begin{cases} +1 & \text{if } a\text{is a square} \quad \pmod p \\ -1 & \text{otherwise} \end{cases}$$

Moreover, a is a square mod$p$ if and only if:

$$a^{\frac{p-1}{2}} \equiv 1 \pmod p$$

Here we must compute (91^{83} mod 167).To perform the calculation we use the Square-and-Multiply algorithm.

i	c_i	$z \pmod{167}$
6	1	$1^2 \times 91 = 91$
5	0	$91^2 = 98$
4	1	$98^2 \times 91 = 53$
3	0	$53^2 = 137$
2	0	$137^2 = 65$
1	1	$65^2 \times 91 = 41$
0	1	$41^2 \times 91 = 166$

Therefore

$$91^{83} \equiv 166 \equiv -1 \pmod{167}$$

So, 91 is not a square and the Legendre symbol has value -1:

$$\left(\frac{91}{167}\right) = -1$$

8

Discrete logarithms

8.1 Discrete logarithms

8.1.1 Pohlig-Hellman

Exercise 8.1 Knowing that $4 \equiv 3^x \pmod 7$, calculate x.

Solution Pohlig-Hellman algorithm is useful if $p-1$ is a product of small prime numbers. Here we have $p-1 = 6 = 3 \times 2$.

Our goal is writing a system of equation of this kind:

$$\begin{aligned} x \bmod 2 &= \dots \\ x \bmod 3 &= \dots \end{aligned}$$

We first compute $x \bmod 2$. We know that $x_0 = x \bmod 2$ is a solution of the equation:

$$\beta^{(p-1)/q} \equiv \alpha^{x_0(p-1)/q} \pmod p$$

with $\beta = 4$, $\alpha = 3$, $p = 7$, $q = 2$. We also know that $0 \leq x_0 \leq q-1$.

So, we must find the solution of the equation:

$$\begin{aligned} 4^3 &\equiv 3^{3x_0} \pmod 7 \\ 1 &\equiv 3^{3x_0} \pmod 7 \\ x_0 &= 0 \end{aligned}$$

We find $x_0 = x \bmod 3$, which is solution of:

$$\begin{aligned} 4^2 &\equiv 3^{2x_0} \pmod 7 \\ 2 &\equiv 3^{2x_0} \pmod 7 \end{aligned}$$

Trying with $x_0 = 0, 1, 2$ (in this order,) we find that $x_0 = 1$.

The system of equation to be solved with the Chinese remainder theorem is the following:

$$\begin{aligned} x \bmod 2 &\equiv 0 \\ x \bmod 3 &\equiv 1 \end{aligned}$$

which gives $x = 4$.

Exercise 8.2 Knowing that $5 \equiv 7^x \pmod{13}$, calculate x.

Solution Our goal is a system of equation of this kind:

$$\begin{aligned} x \bmod 4 &= \dots \\ x \bmod 3 &= \dots \end{aligned}$$

We first compute the value of $x \bmod 4 = x_0 + 2x_1$.

The term x_0 is solution of the equation:

$$\begin{aligned} \beta^{(p-1)/q} &\equiv \alpha^{x_0(p-1)/q} \pmod{13} \\ 5^6 &\equiv 7^{6x_0} \pmod{13} \\ 12 &\equiv 7^{6x_0} \pmod{13} \end{aligned}$$

which has $x_0 = 1$ as solution.

Once x_0 is known, we can compute:

$$\beta_1 \equiv \beta\alpha^{-x_0} \equiv 5 \times 7^{-1} \equiv 5 \times 2 \equiv 10 \bmod 13 \tag{8.1}$$

The term x_1 is solution of the equation:

$$\begin{aligned} \beta_1^{(p-1)/q^2} &\equiv \alpha^{x_1(p-1)/q} \pmod{13} \\ 10^3 &\equiv 7^{6x_1} \pmod{13} \\ 12 &\equiv 7^{6x_1} \pmod{13} \end{aligned}$$

which has $x_1 = 1$ as solution.

Therefore:

$$x \bmod 4 = 1 + 2 = 3$$

In order to compute $x \bmod 3 = x_0$, we solve the equation:

$$\begin{aligned} 5^4 &\equiv 7^{4x_0} \pmod{13} \\ 1 &\equiv 7^{4x_0} \end{aligned}$$

which gives $x_0 = 0$.

The system to be solved becomes:

$$\begin{aligned} x \bmod 4 &= 3 \\ x \bmod 3 &= 0 \end{aligned}$$

which gives $x = 3$.

8.1.2 Baby step, giant step

Exercise 8.3 Knowing that $11^x \equiv 5 \pmod{31}$, find x.

Solution We choose $N = \lceil \sqrt{31} \rceil = 6$. We compute

$$\alpha^{-N} \equiv 11^{-6} \equiv (11^{-1})^6 \equiv 17^6 \equiv 8 \pmod{31}$$

and build the table:

j, k	$\alpha^j \equiv 11^j$	$\beta\alpha^{-Nk} \equiv 5 \times 12^k$
0	1	$5 \times 1 \equiv 5$
1	11	$5 \times 8 \equiv 9$
2	$11 \times 11 \equiv 28$	...
3	$28 \times 11 \equiv 29$	...
4	$29 \times 11 \equiv 9$	...

We find out that

$$11^4 \equiv 1 \times 11^{-6}$$

Therefore $x = 10$.

Exercise 8.4 Knowing that $5^x \equiv 27 \pmod{103}$, find x.

Solution We choose $N = \lceil \sqrt{103} \rceil + 1 = 12$ and compute:

$$\alpha^{-N} \equiv 5^{-12} \equiv (5^{-1})^{12} \equiv 62^{12} \equiv 100 \pmod{103}$$

We build the table:

j, k	$\alpha^j \equiv 5^j$	$\beta\alpha^{-Nk} \equiv 27 \times 100^k$
0	1	27
1	5	$27 \times 100 \equiv 22$
2	$5 \times 5 \equiv 25$	$22 \times 100 \equiv 37$
3	$25 \times 5 \equiv 22$	...

We find out that:

$$\begin{aligned} 5^3 &\equiv 27 \times 5^{-12\times 1} \\ 5^{3+12} &\equiv 27 \end{aligned}$$

Therefore $x = 15$.

8.2 Applications of discrete algorithms

8.2.1 Diffie-Hellman key exchange

Exercise 8.5 Alice and Bob use the Diffie-Hellman key exchange and publish the numbers $\alpha = 45$ and $p = 113$. Eve intercepts the message from Alice to Bob:

$$A \to B. \quad \alpha^x \equiv 29 \pmod p$$

and the message from Bob to Alice:

$$B \to A. \quad \alpha^y \equiv 10 \pmod p$$

Questions:

1. using an algorithm for the computation of the discrete algorithm, compute x;
2. compute the key K.

Solution We must solve the equation:

$$45^x \equiv 29 \pmod{113}$$

Since $p - 1 = 112 = 2^4 \times 7$ we can use the Pohlig-Hellman algorithm.
We find out that $x = 5$.
We compute now:

$$K = 10^5 \bmod 113 = 108$$

8.2.2 ElGamal cryptosystem

Exercise 8.6 Alice uses the ElGamal cryptosystem $e_K(x, k) = (r, t)$ with:

$$r = \alpha^k \bmod p \tag{8.2}$$

$$t = x\beta^k \bmod p \tag{8.3}$$

Alice choses a secret value a and publishes the following data: $p = 107, \alpha = 8, \beta = \alpha^a \bmod p = 30$.

Bob sends to Alice the following ciphered text in ECB mode in which she erroneously used the same *nonce* k for all the ciphering operations.

$$(84, 15)(84, 68)$$

Eve knows that the plaintext corresponding to the first ciphered block is the number $x_1 = 45$.

Calcolare:

1. the plaintext corresponding to the second block;
2. the ciphertext corresponding to the plaintext $x_3 = 88$;
3. knowing that $a = 4$, write the operations that Alice performs to decipher the last encrypted block.

Solution Eve knows that $x\beta^k \equiv t$, therefore:

$$\begin{aligned} 45\beta^k &\equiv 15 \pmod{107} \\ x_2\beta^k &\equiv 68 \pmod{107} \end{aligned}$$

From which we easily obtain that:

$$\begin{aligned} \beta^k &\equiv 15 \times 45^{-1} \equiv 15 \times 88 \pmod{107} = 36 \pmod{107} \\ x_2 &\equiv 45 \times 68 \times 15^{-1} \equiv 45 \times 68 \times 50 \equiv 97 \pmod{107} \end{aligned}$$

To cipher $x_3 = 88$ using the same k we compute:

$$\begin{aligned} r &= \alpha^k \bmod p = 84 \\ t &= x_3\beta^k \bmod p = 88 \times 36 \bmod 107 = 65 \end{aligned}$$

The deciphering procedure is:

$$x = tr^{-a} \bmod p = 65 \times 84^{-4} \bmod 107 = 65 \times 93^4 \bmod 107 = 88$$

8.2.3 ElGamal digital signature

Exercise 8.7 Alice uses the ElGamal digital signature. Alice chooses a secret number a and publishes the numbers $p = 313, \alpha = 55, \beta = \alpha^a \bmod p = 28$. Alice signs two documents $m_1 = 45, m_2 = 255$. Alice erroneously the same *nonce* for both messages and computes:

$$\begin{aligned} r_1 = r_2 = \alpha^k \bmod p = 146 \\ s_1 = k^{-1}(m_1 - ar_1) \bmod (p-1) = 5 \\ s_2 = k^{-1}(m_2 - ar_2) \bmod (p-1) = 35 \end{aligned}$$

Then she publishes the signatures:

$$\begin{aligned} (m_1, r_1, s_1) = (45, 146, 5) \\ (m_2, r_2, s_2) = (255, 146, 35) \end{aligned}$$

1. Verify the signature of message m_1.
2. Taking advantage of the reutilization of k, compute the private key a.

Solution The verification of the signature of message m_1 is obtained by computing:

$$\begin{aligned} v = \beta^r r_1^s \bmod p = 28^{146} \times 146^5 \bmod 313 = 72 \times 203 \bmod 313 = 218 \\ w = \alpha^m \bmod p = 55^{45} \bmod 313 = 218 \end{aligned}$$

Since $v = w$ the signature is valid.
Eve knows that:

$$\begin{aligned} s_1 k - m_1 \equiv -ar \equiv s_2 k - m_2 \pmod{p-1} \\ (s_1 - s_2)k \equiv m_1 - m_1 \pmod{p-1} \end{aligned}$$

Here we have:

$$282k \equiv 102 \pmod{312}$$

Since $\gcd(312, 282) = 6$ we can write:

$$\begin{aligned} 47k \equiv 17 \pmod{52} \\ k \equiv 31 \times 17 \equiv 7 \pmod{52} \end{aligned}$$

The possible values of k are:

$$k = 7, 59, 111, 163, 215, 267$$

Computing r_1 for all the possible values of k we obtain the correct solution $k = 7$.

Once k is known, Eve solves the equation:

$$\begin{aligned} ar &\equiv m_1 - ks_1 \pmod{p-1} \\ 146a &\equiv 45 - 7 \times 5 && \equiv 10 \pmod{312} \\ 73a &\equiv 5 \pmod{156} \end{aligned}$$

from which two possible values $a = 77$, $a = 233$ can be obtained. Computing β we verify that the correct solution is $a = 77$.

9

Elliptic curves

9.1 Plaintext representation

Exercise 9.1 We want to represent the numbers from 0 to 9 as points of the elliptic curve $E : y^2 \equiv x^3 + 2x + 1 \pmod{31}$ using the Koblitz method with $K = 3$. Choose a point to represent the message $m = 2$.

Solution The messages m are represented as points with abscissa $x = mK + j$ where $0 \leq j < K$ is chosen in such a way that the curve has a corresponding point to x.

In case $m = 2$ we must try with $x = 6$, $x = 7$, $x = 8$.

If $x = 6$, we obtain $y^2 \equiv 12 \pmod{31}$. We must verify whether 12 is a quadratic residue mod31:

$$\left(\frac{12}{31}\right) = \left(\frac{2}{31}\right)^2 \left(\frac{3}{31}\right) = -\left(\frac{1}{3}\right) = -1$$

We try with $x = 7$ and obtain $y^2 \equiv 17$. We verify whether it is a residue:

$$\left(\frac{17}{31}\right) = \left(\frac{14}{17}\right) = \left(\frac{2}{17}\right)\left(\frac{7}{17}\right) = \left(\frac{3}{7}\right) = -\left(\frac{1}{3}\right) = -1$$

We try then with $x = 8$ and obtain $y^2 \equiv 2$, which is a residue. Moreover:

$$y = 2^{32/4} \bmod 31 = 8$$

The point $P = (8, 8)$ represents the message $m = 2$.

9.2 Discrete logarithm

Exercise 9.2 Consider the elliptic curve E with equation: $y^2 \equiv x^3 + 3 \pmod{31}$. The curve E has 43 points.

Let be $A = (1, 2)$ e $B = (17, 24)$. Calculate the number a such that $B = aA$.

Solution We use the baby step, giant step algorithm. We choose $N = \lceil\sqrt{43}\rceil = 7$. We compute

$$-NA = 7(-A) = 7(1, -2) = (25, 2)$$

and build the table:

j, k	jA	$B - NkA = B + k(-NA)$
0	$\mathcal{O}$	$(17, 24)$
1	$(1, 2)$	$(17, 24) + (25, 2) = (14, 22)$
2	$(1, 2) + (1, 2) = (16, 10)$	$(14, 22) + (25, 2) = (24, 30)$
3	$(16, 10) + (1, 2) = (22, 24)$	$(24, 30) + (25, 2) = (22, 7)$
4	$(22, 24) + (1, 2) = (24, 30)$	$\dots$
5	$\dots$	

We find out that $4A = B - 7 \cdot 2A$, therefore $B = (14 + 4)A = 18A$ and $a = 18$.

Note that in order to compute $7(1, -2)$ we can use the Double-&-Add technique:

1	$2\mathcal{O} + (1, -2) = (1, -2)$
1	$2(1, -2) + (1, -2) = (16, 21) + (1, -2) = (22, 7)$
1	$2(22, 7) + (1, -2) = (5, 29) + (1, -2) = (25, 2)$

9.3 Diffie-Hellman

Alice e Bob exchange a session key using the Diffie-Hellman protocol. They publish an elliptic curve $E : y^2 \equiv x^3 + 18x + 2 \pmod{29}$. This curve has $n = 27$ points. They also publish $P = (4, 14)$.

Alice sends the message $A = aP = (7, 6)$ and receives the message $B = bP = (9, 9)$.

1. Verify that P is a primitive generator.
2. Compute b using the Pohlig-Hellman algorithm.
3. Compute the session key.

Solution The number of points $n = 27 = 3^3$ is not prime, some elements have order 3 or 9. If P is primitive, it has order 27. Therefore $27/3P \not\equiv \mathcal{O}$. Here we have:

$$9P = 9(4,14) = 2^3P + P = (10,14)$$

P is primitive.

Let be $x = b$. We must solve the equation $\beta = x\alpha$:

$$(9,9) = x(4,14)$$

The Pohlig-Hellman algorithm allows us to easily compute:

$$x \pmod{27} = x_0 + 3x_1 + 9x_2$$

The term x_0 is solution of the equation:

$$\begin{aligned} 9(9,9) &= 9x_0(4,14) \\ (10,14) &= x_0(10,14) \\ x_0 &= 1 \end{aligned}$$

To calculate the following term, we must first compute $\beta_1 = \beta - x_0\alpha = (7,6)$. The term x_1 is solution of the equation:

$$\begin{aligned} 3(7,6) &= 9x_1(4,14) \\ (10,14) &= x_1(10,14) \\ x_1 &= 1 \end{aligned}$$

For the last term we have $\beta_2 = \beta_1 - 3x_1\alpha = \mathcal{O}$.

$$\begin{aligned} \mathcal{O} &= 9x_2(4,14) \\ x_2 &= 0 \end{aligned}$$

Therefore $b = x = 1 + 3 = 4$.

The session key is the point $K = bA = 4(7,6) = (3,24)$.

9.4 DSA signature

Exercise 9.3 Alice signs using the DSA technique. She publishes the curve $E : y^2 \equiv x^3 + 2x + 2 \pmod{13}$ and a basepoint $A = (2,1)$. Alice verifies the order of A and obtains $q = 5$, she chooses a secret a. She computes $B = aA = (2,-1)$ and publishes it.

Alice signs the message $m = 4$, chooses a random and secret k and computes:

$$\begin{aligned} R &= kA = (6, -3) = (x_R, y_R) \\ s &\equiv k^{-1}(m + ax_R) \equiv 4 \pmod 5 \end{aligned}$$

The signature: $(m, R, s) = (4, (6, -3), 4)$.
Successively Alice signs the message $m = 2$ and obtains: $(m, R, s) = (2, (6, -3), 3)$.
Questions:

1. Verify that the order of A is q.

2. Verify Alice's signature.

3. Taking advantage of Alice's mistake, compute the secret key a.

Solution We compute:

$$5A = 2^2(2, 1) + (2, 1) = 2(6, -3) + (2, 1) = (2, 12) + (2, 1) = \mathcal{O}$$

The order of A is 5.

We verify the signature:

$$\begin{aligned} u_1 &\equiv s^{-1}m \equiv 4 \cdot 4 \equiv 1 \pmod 5 \\ u_2 &\equiv s^{-1}x_R \equiv 4 \cdot 6 \equiv 4 \pmod 5 \\ V &= u_1A + u_2B = A + 4B = (2, 1) + 4(2, -1) = \\ &= (2, 1) + (2, 1) = (6, -3) = R \end{aligned}$$

The signature is valid.

Alice used the same k twice,so we can write the following equation:

$$\begin{aligned} s_1k - m_1 \equiv ax_R &\equiv s_2k - m2 \pmod q \\ (s_1 - s_2)k &\equiv m_1 - m_2 \pmod q \\ k &= 2 \end{aligned}$$

Now we substitute the value of k in the equation $sk - m \equiv ax_R$ and obtain:

$$\begin{aligned} a &\equiv x_R^{-1}(sk - m) \pmod q \\ a &\equiv 6^{-1}(4 \cdot 2 - 4) \equiv 4 \pmod 5 \end{aligned}$$

Exercise 9.4 Alice uses a DSA signature.She publishes the curve with equation:

$$E : y^2 \equiv x^3 + 3 \bmod 31;$$

The curve has $q = 43$ points. Alice chooses a point $A = (11, 1)$ and a secret $a = 7$, then she publishes $B = aA$. She signs the message $m = 10$ using the *nonce* $k = 17$.

1. Verify whether P satisfies the requirements f the DSA signature.

2. Compute B.

3. Sign the message m.

4. Verify the signature.

Solution The order of P must be prime. In this case the number of points is prime, therefore all the points have order q.

$$B = aA = 7(11,1) = 8(11,1) + (11,-1) = (9,22) + (11,-1) = (27,-1)$$

Signature:

$$\begin{aligned} R &= kA = 17A = 2(9,22) + (11,1) = (22,24) \\ s &\equiv k^{-1}(m + ax_R) = 38(10 + 7 \cdot 22) = 40 \pmod{43} \end{aligned}$$

Verification:

$$\begin{aligned} u_1 &\equiv s^{-1}m = 14 \cdot 10 = 11 \pmod{43} \\ u_2 &\equiv s^{-1}x_R = 14 \cdot 22 = 7 \pmod{43} \\ V &= u_1A + u_2B = 11A + 7B = 8A + 2A + A + 8B - B = \\ &= (9,22) + (6,23) + (11,1) + (8,9) + (27,1) = \\ &= (23,24) + (26,8) + (27,1) = (22,24) \end{aligned}$$

Since $V = R$, the signature is verified.

9.5 ElGamal signature

Exercise 9.5 Alice uses the following ElGamal signature with elliptic curves. Alice chooses the curve:

$$E : y^2 \equiv x^3 + 3 \pmod{31}$$

The number $p = 31$ is prime. Alice computes the number of points n which belong to the curve and obtain $n = 43$. On the curve E she chooses the point $A = (1,2)$ and the secret number $a = 18$. She then computes the position of the point $B = aA$ and obtains:

$$B = aA = (17,24)$$

Alice publishes the curve E, the number p and the position of the points A e B. The number a is kept secret.

To sign a message m, Alice chooses a random number $k \in \mathbb{Z}_n^*$ con $\gcd(k,n)=1$ and computes:

$$\begin{aligned} R &= kA = (x,y) \\ s &\equiv k^{-1}(m-ax) \pmod{n} \end{aligned}$$

Bob's verification of the signature is done by computing:

$$\begin{aligned} V_1 &= xB + sR \\ V_2 &= mA \end{aligned}$$

If $V_1 = V_2$, the message is genuine.

1. Alice wants to send the message $m=7$ and chooses the random number $k=3$. Compute Alice's signature.

2. Verify the signature.

Solution To calculate sums on the curve E we remember that:

$$\begin{aligned} x_1 + x_2 + x_3 &= m^2 \\ y_1 + y_3 &= m(x_1 - x_3) \end{aligned}$$

where m is the slope of the line that connect the two known points. If the two points are distinct, then:

$$m = (y_2 - y_1)(x_2 - x_1)^{-1} \mod 31$$

otherwise

$$m = (3x_1^2)(2y_1)^{-1} \mod 31$$

Alice must compute:

$$R = kA = 3 \cdot (1,2) = (1,2) + (1,2) + (1,2)$$

We first compute $(1,2)+(1,2)$, obtaining:

$$m = 3 \cdot 4^{-1} \mod 31 = 3 \cdot 8 = 24$$

To compute the inverse of 25 modulo 31 we can use the extended Euclidean algorithm. By solving the equations for x_3 e y_3 we obtain:

$$\begin{aligned} x_3 &= m^2 - x_1 - x_2 = 24^2 - 2 \cdot 1 \mod 31 = 16 \\ y_3 &= m(x_1 - x_3) - y_1 = 24(1-16) - 2 = 10 \end{aligned}$$

We then sum $(1,2)+(16,10)$ and obtain:

$$\begin{aligned}
m &= 8\cdot 15^{-1} \mod 31 = 8\cdot(-2) = 15\\
x_3 &= 15^2 - 1 - 16 \mod 31 = 22\\
y_3 &= 15\cdot(1-22) - 2 \mod 31 = 24
\end{aligned}$$

Therefore $R=(22,24)$.
We compute s:

$$\begin{aligned}
s = k^{-1}(m - ax_R) = 3^{-1}(7 - 18\cdot 22) \mod 43 =\\
= 29\cdot(-389) \mod 43 = 28
\end{aligned}$$

Alice publishes the message m and the signature (m,R,s).

$$\begin{aligned}
V_1 &= x_R B + sR = 22\cdot(17,24) + 28\cdot(22,24) = (9,22)+(16,21) = (25,29)\\
V_2 &= mA = 7\cdot(1,2) = (25,29)
\end{aligned}$$

The signature is verified.

9.6 ElGamal cryptosystem

Exercise 9.6 Alice uses the public key ElGamal cryptosystem. She publishes the curve $E: y^2 \equiv x^3 + 2x + 2 \pmod{13}$ and the point $A=(3,3)$ of order 15. She also chooses a secret number $a=7$ and publishes the point $B=aA$. Bob wants to send to Alice a message corresponding to the point $P_m=(8,6)$.

Questions:

1. Calculate B.
2. Cipher the message using $k=3$.
3. Decipher the message.

Solution

$$\begin{aligned}
B = 7A = 2^3 A - A = 2^2(4,3) - (3,3) =\\
= 2(8,7) + (3,-3) = (11,9) + (3,-3) = (11,4)
\end{aligned}$$

Bob computes:

$$\begin{aligned}
Y_1 &= kA = 3(3,3) = 2(3,3) + (3,3) = (4,3) + (3,3) = (6,-3)\\
Y_2 &= P_m + kB = (8,6) + 3(11,4) = (8,6) + (3,-3) + (11,4) = (4,3)
\end{aligned}$$

Alice deciphers:

$$\begin{aligned} P_m = Y_2 - aY_1 = (4,3) - 7(6,-3) = (4,3) + 8(6,3) + (6,-3) = \\ = 4(2,1) + (12,8) = (2,12) + (12,8) = (8,6) \end{aligned}$$

10

Public Key Cryptography

10.1 RSA

Exercise 10.1 (Ciphering) Alice uses the RSA algorithm to receive messages from Bob. Alice chooses two prime numbers $p = 13$ e $q = 23$. She also chooses a ciphering exponent $e = 35$.

Alice publishes the product $N = pq = 299$ and the ciphering exponent, but she keeps secret the two numbers p e q.

Questions:

1. verify that the exponent e satisfies the requirements of the RSA algorithm;
2. compute the deciphering exponent d;
3. cipher the number $P = 15$ and verify that it can be correctly deciphered using the computed exponent d.

Solution We first compute $\phi(N) = (p-1)(q-1) = 264$. The exponent e must be coprime with $\phi(N)$.

r	q	s	t
264	–	1	0
35	7	0	1
19	1	1	-7
16	1	-1	8
3	5	2	-15
1	3	-11	83

Hence:

$$\gcd(35, 264) = 1 = -11 \times 264 + 83 \times 35$$

The exponent e is valid, the deciphering exponent turns out to be:

$$\begin{aligned} d &= e^{-1} \bmod \phi(N) \\ d &= 35^{-1} \bmod 264 \\ d &= 83 \end{aligned}$$

The exponentiation can be done with the SQUARE-&-MULTIPLY technique. Given $P = 15$, we can compute:

$$C = P^e \bmod N = 15^{35} \bmod 299 = 189$$

$$\begin{array}{c|r} 1 & 15 \\ 0 & 15^2 = 225 \pmod{299} \\ 0 & 225^2 = 94 \pmod{299} \\ 0 & 94^2 = 165 \pmod{299} \\ 1 & 165^2 \times 15 = 240 \pmod{299} \\ 1 & 240^2 \times 15 = 189 \pmod{299} \end{array}$$

Deciphering, we obtain:

$$P = C^d \bmod N = 189^{83} \bmod 299 = 15$$

Exercise 10.2 Alice uses texbook RSA algorithm to receive messages from Bob. She publishes $N = 385$ and $e = 7$.

Bob needs to send a one-digit id; computes $c = 262$ and sends it to Alice. The message must be protected from an eavesdropper, Eve, who can perform at most 20 exponentiations.

Ron suggests that textbook RSA is not secure and suggests to prepend a random number to the message. (This is similar to the PKCS#1 v. 1.5 RSA modification).

Questions:

1. Show that textbook RSA is not semantic secure.
2. Show how Eve can decrypt the message c without knowing the private key d. Decrypt c.
3. Write the modified RSA algorithm as suggested by Ron.
4. Calculate the probability that Eve can decrypt the message after the modifications by Ron. Assume that RSA encryption is all-or-nothing.

5. Assuming that Alice uses a 1024-bit modulus n and that Eve can perform up to 2^{80} exponentiations, calculate the probability that Eve can decrypt the message, whether Ron's modification is necessary and how long must be the random prefix in the following cases:

   (a) Bob sends random 500-bit messages

   (b) Bob sends messages out of an alphabet of two characters

Solution Adversary sends m_0, m_1. If it receives $m_0^e \bmod N$, then $b' = 0$. Otherwise it is 1.

Eve tries all messages starting from 0.

m	c
0	0
1	1
2	128
3	262

So $m = 3$.

Bob:

- Choose r randomly in $\mathbb{Z}_{38}$.
- Calculate $x = r \| m$.
- If $x \geq 385$ choose a new r.
- Calculate $c = x \bmod N$.

Alice:

- Calculate $x = c \bmod N$.
- Calculate $m = x \bmod 10$.

Eve finds the correct decryption if she finds both the correct m and the correct r. This happens with probability $1/N$.

In case of random messages, Eve can guess the correct message with probability $2^{-500+80}$. Ron's modifications are not necessary.

In case of two messages, Eve can guess the correct message with probability 1. Ron's modifications are necessary. With an 80-bit prefix, Eve can guess with probability $2^{80}/2^{81}$. With a 160-bit prefix, Eve can guess with probability $2^{80}/2^{161} = 2^{-81}$.

Exercise 10.3 (Digital signature with RSA) Alice publishes the following data: $N = pq = 221$ and $e = 13$. Bob receives the message $P = 65$ and the corresponding digital signature $F = 182$.

Ron suggest using RSA-FDH using the function $h\colon \{0,1\}^* \to \{0,1\}^7$.

1. Verify the signature.
2. Suppose that Oscar can perform 10 hashes or exponentiations. Calculate the probability that Oscar can make a selective forgery of a signature with a pk-only attack.
3. Calculate the probability that Oscar can make an existential forgery of a signature with a pk-only attack.

Solution The signature is valid if $P = F^e \bmod N$. Here we have:

$$F^e \bmod N = 182^{13} \bmod 221 = 65$$

For a selective forgery of message m, Oscar calculates the hash of m. Then he needs to find $F = h(m)^d \bmod N$. Trying randomly, Oscar finds the correct signature with probability $10/N$.

For an existential forgery, Oscar chooses a random F, then $F^e \bmod N$. Note that with probability $1/2$ $F^e \geq 128$ and a new F must be chosen. Then, Oscar needs to find a preimage for F. Trying randomly, Oscar finds it with probability $10/2^7$.

Exercise 10.4 Alice publishes the following RSA public key: $K_{pub} = (e, n) = (93, 17947)$.

1. Cipher the message $m = 7$ using the S & M algorithm.
2. Using Fermat's factorization method, find p, q.
3. Using the $p - 1$ factorization algorithm, find p, q. (Use $a = 100$ and $B = 3$, $B = 4$).
4. Verify that n is not prime using Miller-Rabin's algorithm.
5. Compute the private key (d, n).

Solution The ciphertext is $c = m^e \mod n = 7^{93} \mod 17947$. To use S & M we must start from the binary representation of $93 = 1011101$.

1	7
0	$7^2 = 49$
1	$49^2 \times 7 = 16807 = -1140$
1	$1140^2 \times 7 = 16018 = -1929$
1	$1929^2 \times 7 = 6190$
0	$6190^2 = 17202 = -745$
1	$745^2 = 8623$

Therefore $c = 8623$.

Fermat's factorization For Fermat's factorization we start considering that:

$$t = \frac{p+q}{2} \simeq \sqrt{n} = 133,96$$

Supposing $\hat{t} = 134$, we have

$$\hat{s}^2 = \hat{t}^2 - n = 134^2 - 17947 = 9 = 3^2$$

We write the equations:

$$\begin{aligned} t &= \frac{p+q}{2} = 134 \\ s &= \frac{p-q}{2} = 3 \end{aligned}$$

from which we obtain $p = 137$, $q = 131$.

$p - 1$ **factorization** We start from $a = 100$ and compute the succession $b_1, b_2, \ldots$:

$$\begin{aligned} b_1 &= a = 100 \\ b_2 &= b_1^2 \mod n = 10000 \\ b_3 &= b_2^3 \mod n = 15754 \\ b_4 &= b_3^4 \mod n = 12057 \end{aligned}$$

For b_3 we obtain that $\gcd(b_3 - 1, n) = \gcd(15753, 17947) = 1$, since:

$$\gcd(15753, 17947) = \gcd(17947 \mod 15753, 17947) = \gcd(2194, 17947) = \cdots = 1$$

On the other hand, for b_4 we obtain $\gcd(b_4 - 1, n) = 137$.

Miller-Rabin We chose $a = 2$.

$$n - 1 = 17946 = 2 \times 8973$$

From which it derives that $k = 1$, $m = 8973$. We compute

$$b_0 = a^m \mod n = 2^{8973} \mod n = 3545$$

so the number is composite.

To find the private key it is necessary to calculate

$$d = e^{-1} \mod \phi(n) = e^{-1} \mod (p-1)(q-1) = 93^{-1} \mod 17680 = 12357$$

10.2 Factorization

10.2.1 Fermat's factorization

Exercise 10.5 Alice publishes the following data: $N = 6557$, $e = 131$. Find the deciphering exponent d.

Solution Assuming $p > q$ the following equalities are always true:

$$N = t^2 - s^2 = \left(\frac{p+q}{2}\right)^2 - \left(\frac{p-q}{2}\right)^2$$

Fermat's factorization is efficient if $p \simeq q$. In this case we have $t \simeq \sqrt{n}$ and $s \simeq 0$.

We try with all the integer numbers $t > \sqrt{N}$ and we compute:

$$\widehat{s}^2 = t^2 - N$$

We go on until $\widehat{s}^2$ is not a perfect square.
In this case it must be $t > 80, 9$.
We try $t = 81$. For $t = 81$ we have $\widehat{s}^2 = 6561 - 6557 = 4$. Therefore:

$$p + q = 162$$
$$p - q = 4$$

We obtain $p = 83$ e $q = 79$.
The deciphering exponent is:

$$d = e^{-1} \bmod \phi(N) = 131^{-1} \bmod (p-1)(q-1) = 131^{-1} \bmod 6396$$

The inverse of 131 modulo 6396 can be found using the extended Euclidean algorithm:

r	q	s	t
6396	–	1	0
131	48	0	1
108	1	1	−48
23	4	−1	49
16	1	5	−244
7	2	−6	293
2	3	17	−830
1	2	−57	2783
0	–	131	−6396

The inverse of 131 is $d = 2783 \pmod{6396}$.

10.2.2 p − 1 algorithm

Let n be the number to be factorized. We chose an integer number $a > 1$, e.g. $a = 2$ and a limit B.

We compute $b = a^{B!} \bmod n$ and $d = \gcd(b-1, n)$. The result d is a factor of n; the algorithm succeeds and in case $d \neq 1$ or $d \neq n$ we have obtained a non banal factor.

Exercise 10.6 Factorize the number 1001, using the basis $a = 2$.

Solution We chose $B = 3$. Then:

$$\begin{aligned} b &= a^{B!} \bmod n = 2^{3!} \bmod 1001 = 2^6 \bmod 1001 = 64 \\ d &= \gcd(b-1, n) = \gcd(63, 1001) = 7 \end{aligned}$$

Therefore 7 is a factor. Indeed, $1001/7 = 143$. As 143 is not prime, we must go on with the factorization.

We try to increment B.

$$\begin{aligned} b &= 2^{4!} \bmod 143 = 64^4 \bmod 143 = 27 \\ d &= \gcd(26, 143) = 13 \end{aligned}$$

Therefore, 13 is another factor. We conclude that:

$$1001 = 7 \times 11 \times 13$$

10.3 Discrete Logarithm Problem

Exercise 10.7 Consider the following commitment scheme (Pedersen Commitment).

Let G a group with prime order q where DLP is hard. Let g, h random elements in E.

1. Commit. Alice commits to x by choosing a random r with $x, r \in \mathbb{Z}_q$. Alice sends Bob $c = g^x h^r$.
2. Bob stores c and does his computations. Later, Alice reveals x, r.
3. Verification. Bob verifies that $c = g^x h^r$.

Questions

1. Show that Bob learns nothing about x from c.

2. Show that if Alice changes x after the commitment, Alice knows how to solve a DLP.

Solution Let $h = g^a$ for some a. Then $c = g^{x+ar}$. Given x, r, x', there always exists an r' s.t. $g^{x+ar} = g^{x'+ar'}$.

If Alice finds x, r, x', r' s.t. $g^{x+ar} = g^{x'+ar'}$, then

$$x + ar = x' + ar' \bmod q$$

So, $a = (x - x')(r' - r)^{-1} \bmod q$.

Exercise 10.8 Let Alice and Bob use Pedersen Commitments.

A trusted generator chooses the curve

$$E\colon y^2 = x^3 + x + 9 \bmod 13$$

with 14 points, and the base point $P = (8, 10)$. From P, the generator publishes $G = 2 * P$ and $H = 3G$.

Alice publishes (6,7). She later reveals $x = 2, r = 3$.

1. Tell why G and H are good choices for the public parameters and tell the value of q.

2. Verify the commitment.

Solution If $\text{ord}(E) = 14$, then $7G = 0$, and $\text{ord}(G) = 7$. Since $H \in< G >$ and $q = 7$ it also has order 7.

Note that $G = (0, 10)$ and $H = (6, 6)$.

Bob must verify:

$$(6, 7) = 2G + 3H = 2(0, 10) + 3(6, 6) = (4, 8) + (4, 8) = (6, 7)$$

Exercise 10.9 Consider the following authentication scheme (Shnorr). It allows the same client to reuse the same private key on multiple servers.

Let G a group with prime order q. A client (C) holds a private key. $x \in \mathbb{Z}_q$ and publishes $y = g^x$.

Then it executes the following authentication protocol with server S:

1. $C \to S$. Choose random $k \in \mathbb{Z}_q$. Send $r = g^k$.

2. $S \to C$. Send random challenge $e \in \mathbb{Z}_q$.

3. $C \to S$. Calculate $s = k + xe \bmod q$. Send s.

4. The client is authenticated if $g^s = ry^e$.

Questions:

1. Show that, if k is reused, an eavesdropper can calculate x.
2. Suppose that Eve observed a full protocol exchange. Eve tries to authenticate by repeating message 1 and message 3. What is the probability of success?

Solution If k is reused, then Eve knows k, e, s, k, e', s' s.t.

$$s - xe = s' - xe' \bmod q$$

thus $x = (s - s')(e - e')^{-1} \bmod q$.

Eve succeeds only if $s = k + xe = k + xe'$, but this possible only if $e = e'$. This happens with probability $1/q$.

$\mathcal{A}$

Modular Arithmetic

A.1 GCD and Euclidean Algorithm

Exercise A.1 Compute $d = \gcd(360, 294)$ in two ways:

1. by factorizing each of the two numbers and then factorizing d;
2. using the Euclidean algorithm.

Solution Notice that

$$\begin{aligned} 360 &= 6^2 \times 10 = 5 \times 3^2 \times 2^3 \\ 294 &= 2 \times 3 \times 7^2 \end{aligned}$$

The gcd is the product of the common factors, each one elevated to the lowest exponent. Therefore

$$d = 3 \times 2 = 6$$

We define:

$$\begin{aligned} r_0 &= \max(a, b) \\ r_1 &= \min(a, b) \end{aligned}$$

Our aim is obtaining a sequence of expressions of this kind:

$$\begin{aligned} r_0 &= q_1 r_1 + r_2 \\ &\dots \\ r_{j-2} &= q_{j-1} r_{j-1} + r_j \\ &\dots \end{aligned}$$

Where j is the index of the expression, r_j e q_j are integer numbers. The algorithm is initialized with:

$$
\begin{aligned}
r_0 &\leftarrow a \\
r_1 &\leftarrow b \\
j &\leftarrow 0
\end{aligned}
$$

The algorithm stops when $r_j = 0$ and we obtain $d = \gcd(a, b) = r_{j-1}$.

j	r_j	
0	360	$--$
1	294	$--$
2	66	$360 = 1 \cdot 294 + 66$
3	30	$294 = 4 \cdot 66 + 30$
4	6	$66 = 2 \cdot 30 + 6$
5	0	$30 = 5 \cdot 6 + 0$

Therefore, $d = \gcd(360, 294) = 6$.

Exercise A.2 Find $d = \gcd(841, 294)$ and express d as linear combination of the two numbers.

Solution The extended Euclidean algorithm allows us to find two numbers s and t such that:

$$
\begin{aligned}
\gcd(a, b) &= r_0 s + r_1 t \\
&\text{with} \\
r_0 &= \max(a, b) \\
r_1 &= \min(a, b)
\end{aligned}
$$

We define the following recursions:

$$
s_j = \begin{cases} 1 & \text{if } j = 0 \\ 0 & \text{if } j = 1 \\ s_{j-2} - q_{j-1} s_{j-1} & \text{if } j \geq 2 \end{cases}
$$

e

$$
t_j = \begin{cases} 0 & \text{if } j = 0 \\ 1 & \text{if } j = 1 \\ t_{j-2} - q_{j-1} t_{j-1} & \text{if } j \geq 2 \end{cases}
$$

From this definition, it follows that s_j e t_j always satisfy the following equality:

$$
r_j = r_0 s_j + r_1 t_j
$$

Therefore, the values of s, t which allow us to express the gcd as linear combination of two numbers are the coefficients s_j, t_j obtained for $r_j = \gcd(a, b)$, that is, for the last non null remainder.

The table reports all the passages of the algorithm, showing r_j, q_j and the computation of the expression $r_j = r_0 s_j + r_1 t_j$.

j	r_j	q_j		s_j	t_j	$r_0 s_j + r_1 t_j$
0	841	–	–	1	0	841
1	294	2	–	0	1	294
2	253	1	$841 = 2 \cdot 294 + 253$	1	-2	253
3	41	6	$294 = 1 \cdot 253 + 41$	-1	3	41
4	7	5	$253 = 6 \cdot 41 + 7$	7	-20	7
5	6	1	$41 = 5 \cdot 7 + 6$	-36	103	6
6	1	6	$7 = 1 \cdot 6 + 1$	43	-123	1
7	0	–	$6 = 6 \cdot 1 + 0$	-294	841	0

As the last non null remainder is obtained for $j = 6$, the two numbers are coprime and the two required coefficients are $s = 43$ e $t = -123$.

In the remainder of the workbook, we will fill the table only with the expressions $r_{j-2} = q_{j-1} r_{j-1} + r_j$ and the columns s_j and t_j.

	s_j	t_j
–	1	0
–	0	1
$841 = 2 \cdot 294 + 253$	1	-2
$294 = 1 \cdot 253 + 41$	-1	3
$253 = 6 \cdot 41 + 7$	7	-20
$41 = 5 \cdot 7 + 6$	-36	103
$7 = 1 \cdot 6 + 1$	43	-123
$6 = 6 \cdot 1 + 0$	-294	841

A.2 Coding with matrices

Exercise A.3 (Hill code book on $\mathbb{Z}_4$) The binary plaintext $P =$ 011000110100is ciphered using a Hill code book on $\mathbb{Z}_4$. The key K is a 2×2 matrix.

1. Which requirements must K satisfy?
2. Cipher the P considering

$$K = \begin{pmatrix} 0 & 1 \\ 3 & 3 \end{pmatrix}$$

3. Decipher the ciphertext obtained above.

4. Compute the key K using a *known plaintext* attack on the plaintext-ciphertext pair obtained above.

Solution 1. The key K must satisfy the two following requirements:

$$\begin{cases} \det(K) \neq 0 \\ \gcd(\det(K), 4) = 1 \end{cases}$$

Note that requirement 2 is implied by 1.

2. We first convert the plaintext in elements of $\mathbb{Z}_4$. As the possible elements are $\{0, 1, 2, 3\}$ each couple of bits is converted in a number. Therefore, the plaintext becomes:

$$P = \begin{pmatrix} 1 & 2 & 0 & 3 & 1 & 0 \end{pmatrix}$$

The key is a 2×2 matrix, so the plaintext must be divided in vectors P_i of two elements each:

$$\begin{aligned} P_1 &= \begin{pmatrix} 1 & 2 \end{pmatrix} \\ P_2 &= \begin{pmatrix} 0 & 3 \end{pmatrix} \\ P_3 &= \begin{pmatrix} 1 & 0 \end{pmatrix} \end{aligned}$$

We now apply the formula:

$$C_i = P_i \cdot K \pmod 4$$

$$\begin{aligned} C_1 &= \begin{pmatrix} 1 & 2 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 3 & 3 \end{pmatrix} = \begin{pmatrix} 6 & 7 \end{pmatrix} = \begin{pmatrix} 2 & 3 \end{pmatrix} \pmod 4 \\ C_2 &= \begin{pmatrix} 0 & 3 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 3 & 3 \end{pmatrix} = \begin{pmatrix} 9 & 9 \end{pmatrix} = \begin{pmatrix} 1 & 1 \end{pmatrix} \pmod 4 \\ C_3 &= \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 3 & 3 \end{pmatrix} = \begin{pmatrix} 0 & 1 \end{pmatrix} \pmod 4 \end{aligned}$$

The ciphertext C is:

$$C = \begin{pmatrix} 2 & 3 & 1 & 1 & 0 & 1 \end{pmatrix}$$

which becomes $C = \texttt{101101010001}$ if expressed in binary form.

3. We must first compute the inverse matrix of the key:

$$K^{-1} = \frac{1}{-3}\begin{pmatrix} 3 & -1 \\ -3 & 0 \end{pmatrix} \pmod 4$$

We must compute the inverse element of -3 in $\mathbb{Z}_4$. But $-3 \equiv 1 \pmod 4$, and the inverse of 1 is still 1, therefore:

$$K^{-1} = \begin{pmatrix} 3 & 3 \\ 1 & 0 \end{pmatrix} \pmod 4$$

We compute the plaintext:

$$\begin{aligned} P_1 &= \begin{pmatrix} 2 & 3 \end{pmatrix}\begin{pmatrix} 3 & 3 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} 9 & 6 \end{pmatrix} = \begin{pmatrix} 1 & 2 \end{pmatrix} \pmod 4 \\ P_2 &= \begin{pmatrix} 1 & 1 \end{pmatrix}\begin{pmatrix} 3 & 3 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} 4 & 3 \end{pmatrix} = \begin{pmatrix} 0 & 3 \end{pmatrix} \pmod 4 \\ P_3 &= \begin{pmatrix} 0 & 1 \end{pmatrix}\begin{pmatrix} 3 & 3 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 \end{pmatrix} \pmod 4 \end{aligned}$$

Therefore, the plaintext is:

$$P = \begin{pmatrix} 1 & 2 & 0 & 3 & 1 & 0 \end{pmatrix}$$

4. We must build an equation of this kind:

$$\begin{pmatrix} P_i \\ P_j \end{pmatrix} K = \begin{pmatrix} C_i \\ C_j \end{pmatrix}$$

where the rows P_i, C_i e P_j, C_j are two plaintext/ciphertext pairs of length two.

By solving the equation for the matrix of variables, we obtain the key K.To do so, the plaintext matrix must be invertible;we have to consider this requirement when we chose the rows of the matrix.

Fortunately, the matrix built with P_1 e P_2 satisfies the requirement:

$$\det\begin{pmatrix} 1 & 2 \\ 0 & 3 \end{pmatrix} = 3$$

which is neither null nor a multiple of 2.

therefore, we can write:

$$K = \begin{pmatrix} 1 & 2 \\ 0 & 3 \end{pmatrix}^{-1} \cdot \begin{pmatrix} 2 & 3 \\ 1 & 1 \end{pmatrix} = \frac{1}{3} \begin{pmatrix} 3 & -2 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 2 & 3 \\ 1 & 1 \end{pmatrix}$$
$$= 3 \begin{pmatrix} 4 & 7 \\ 1 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 3 & 3 \end{pmatrix} \pmod 4 \quad \text{(A.1)}$$

In the previous equation we considered $1/3 \equiv 3 \pmod 4$: the extended Euclidean algorithm can be used to compute the inverse of 3, otherwise it is worth noting that $3 \cdot 3 = 9 \equiv 1 \pmod 4$.

Exercise A.4 Alice and Bob use an affine ciphering technique based on arithmetics in $\mathbb{Z}_{10}$. The ciphering algorithm has the following expression:

$$C_i = P_i \cdot K + B \pmod{10}$$

The row vectors 1×2 C_i e P_I contain the i-th pair of plain and ciphered numbers, the key is composed by the matrix K and the vector B.

The symmetric key is:

$$K = \begin{pmatrix} 3 & 7 \\ 2 & 7 \end{pmatrix}$$
$$B = \begin{pmatrix} 4 & 9 \end{pmatrix}$$

1. Verify that K is a valid key.
2. Cipher the message:
$$P = \begin{pmatrix} 6 & 7 & 3 & 9 & 3 & 6 \end{pmatrix}$$
3. Decipher the ciphertext.
4. Find the key (K e B) through a *known-plaintext* attack.

Solution To be valid, K must satisfy the following requirements:

$$\begin{cases} \det(K) \neq 0 \\ \gcd(\det(K), 10) = 1 \end{cases}$$

In this case we have $\det(K) = 7$, so the two conditions are satisfied.

We decompose the message P in 3 row vectors:

$$P_1 = \begin{pmatrix} 6 & 7 \end{pmatrix}$$
$$P_2 = \begin{pmatrix} 3 & 9 \end{pmatrix}$$
$$P_3 = \begin{pmatrix} 3 & 6 \end{pmatrix}$$

We compute the ciphertexts:

$$C_1 = \begin{pmatrix} 6 & 7 \end{pmatrix} \begin{pmatrix} 3 & 7 \\ 2 & 7 \end{pmatrix} + \begin{pmatrix} 4 & 9 \end{pmatrix} = \begin{pmatrix} 6 & 0 \end{pmatrix}$$
$$C_2 = \begin{pmatrix} 3 & 9 \end{pmatrix} \begin{pmatrix} 3 & 7 \\ 2 & 7 \end{pmatrix} + \begin{pmatrix} 4 & 9 \end{pmatrix} = \begin{pmatrix} 1 & 3 \end{pmatrix}$$
$$C_3 = \begin{pmatrix} 3 & 6 \end{pmatrix} \begin{pmatrix} 3 & 7 \\ 2 & 7 \end{pmatrix} + \begin{pmatrix} 4 & 9 \end{pmatrix} = \begin{pmatrix} 5 & 2 \end{pmatrix}$$

Therefore, the ciphertext is:

$$C = \begin{pmatrix} 6 & 0 & 1 & 3 & 5 & 2 \end{pmatrix}$$

To decipher the ciphertext we must invert the matrix K. the deciphering algorithm has the following expression:

$$P = CK^{-1} - BK^{-1}$$

$$K^{-1} = \frac{1}{7} \begin{pmatrix} 7 & -7 \\ -2 & 3 \end{pmatrix} \bmod 10$$

The inverse of 7 (mod 10) can be computed with the extended Euclidean algorithm:

r_j	q_j		s_j	t_j
10	–	–	1	0
7	1	–	0	1
3	2	$10 = 1 \cdot 7 + 3$	1	-1
1	3	$7 = 2 \cdot 3 + 1$	-2	3
0	–	$3 = 3 \cdot 1 + 0$	7	-10

The inverse of 7 is 3, indeed:

$$7 \cdot 3 \bmod 10 = 21 \quad \bmod 10 = 1$$

Therefore:

$$K^{-1} = 3 \begin{pmatrix} 7 & 3 \\ 8 & 3 \end{pmatrix} \bmod 10 = \begin{pmatrix} 1 & 9 \\ 4 & 9 \end{pmatrix}$$
$$-BK^{-1} = -\begin{pmatrix} 4 & 9 \end{pmatrix} \begin{pmatrix} 1 & 9 \\ 4 & 9 \end{pmatrix} = \begin{pmatrix} 0 & 3 \end{pmatrix}$$

We compute the plaintexts:

$$P_1 = \begin{pmatrix} 6 & 0 \end{pmatrix} \begin{pmatrix} 1 & 9 \\ 4 & 9 \end{pmatrix} + \begin{pmatrix} 0 & 3 \end{pmatrix} = \begin{pmatrix} 6 & 7 \end{pmatrix}$$

$$P_2 = \begin{pmatrix} 1 & 3 \end{pmatrix} \begin{pmatrix} 1 & 9 \\ 4 & 9 \end{pmatrix} + \begin{pmatrix} 0 & 3 \end{pmatrix} = \begin{pmatrix} 3 & 9 \end{pmatrix}$$

$$P_3 = \begin{pmatrix} 5 & 2 \end{pmatrix} \begin{pmatrix} 1 & 9 \\ 4 & 9 \end{pmatrix} + \begin{pmatrix} 0 & 3 \end{pmatrix} = \begin{pmatrix} 3 & 6 \end{pmatrix}$$

Given a set of (P_i, C_i) pairs, in order to compute the key we must solve the following equation system:

$$C_1 = P_1 K + B \tag{A.2}$$

$$C_2 = P_2 K + B \tag{A.3}$$

$$C_3 = P_3 K + B \tag{A.4}$$

By subtracting (A.4) to (A.3) and (A.2), we obtain a new equation without the constant B, from which the key K can be computed:

$$\begin{pmatrix} C_1 - C_3 \\ C_2 - C_3 \end{pmatrix} = \begin{pmatrix} P_1 - P_3 \\ P_2 - P_3 \end{pmatrix} K \tag{A.5}$$

$$K = \begin{pmatrix} P_1 - P_3 \\ P_2 - P_3 \end{pmatrix}^{-1} \begin{pmatrix} C_1 - C_3 \\ C_2 - C_3 \end{pmatrix} \tag{A.6}$$

$$K = \begin{pmatrix} 3 & 1 \\ 0 & 3 \end{pmatrix}^{-1} \begin{pmatrix} 1 & -2 \\ -4 & 1 \end{pmatrix} \tag{A.7}$$

To make the attack effective, the plaintext matrix must be invertible. As:

$$\det \begin{pmatrix} 3 & 1 \\ 0 & 3 \end{pmatrix} = 9$$

the matrix is invertible and there are no common factors with 10, therefore there is a unique inverse matrix:

$$K = \frac{1}{9} \begin{pmatrix} 3 & -1 \\ 0 & 3 \end{pmatrix} \begin{pmatrix} 1 & -2 \\ -4 & 1 \end{pmatrix} = 9 \begin{pmatrix} 7 & -7 \\ -12 & 3 \end{pmatrix} = \begin{pmatrix} 63 & -63 \\ -108 & 27 \end{pmatrix}$$

$$K \equiv \begin{pmatrix} 3 & 7 \\ 2 & 7 \end{pmatrix} \pmod{10}$$

Note that $9^{-1} \equiv 9 \pmod{10} = 9$, since $9 \cdot 9 \bmod 10 = 1$.

Once K is found, we can use (A.2) to find B:

$$B = C_1 - P_1 K = \begin{pmatrix} 6 & 0 \end{pmatrix} - \begin{pmatrix} 2 & 1 \end{pmatrix} = \begin{pmatrix} 4 & -1 \end{pmatrix}$$

$$B \equiv \begin{pmatrix} 4 & 9 \end{pmatrix} \pmod{10}$$

A.3 Primality Test

A.3.1 Fermat Test

Fermat theorem says that if $n > 1$ is prime, then $a^{n-1} \equiv 1 \pmod{n}$ per $\forall a < n-1$.

Therefore we can define a *Fermat test* to verify whether n is prime. We choose a set of basis $1 < a_i < n-1$. If a number a_i exists, such that:

$$a_i^{n-1} \not\equiv 1 \pmod{n}$$

we can certainly affirm that n is not prime. If we do not find a_i we can conclude that n is *probably* prime. Obviously, the higher number of basis we consider, the higher the probability that a_i is prime.

Exercise A.5 Using Fermat test, find a prime number greater or equal to 27.

Solution We start trying with basis 2:

$$\begin{aligned} 2^{26} \bmod 27 &= 13 \text{ is not prime} \\ 2^{28} \bmod 29 &= 1 \end{aligned}$$

The number 29 might be prime. We try then other basis:

$$\begin{aligned} 3^{28} \bmod 29 &= 1 \\ 5^{28} \bmod 29 &= 1 \\ 7^{28} \bmod 29 &= 1 \end{aligned}$$

We conclude that 29 is probably prime.

A.3.2 Miller-Rabin test

Given the candidate prime number n, we define two numbers k and m such that:

$$n - 1 = 2^k m$$

with m odd. We then choose a basis a, $1 \leq a \leq n-1$.

We compute $b_0 = a^m \bmod n$.

If $b_0 = \pm 1$, then n is probably prime.

Otherwise, we compute $b_i = b_{i-1}^2 \bmod n$ with $1 \leq i \leq k-1$.

If $b_i = -1$, n is probably prime.

If $b_i = 1$, n is composite and $\gcd(b_{i-1} - 1, n)$ is a factor.

If $b_i \neq \pm 1$, $i \leftarrow i + 1$ and the test is repeated.

If we reach $i = k - 1$ and $b_i \neq -1$, then n is composite.

Exercise A.6 Verify whether 341 is prime using the Fermat test and the Miller-Rabin test.

Solution Fermat test:

$$\begin{aligned} 2^{340} \bmod 341 &= 1 \\ 3^{340} \bmod 341 &= 56 \end{aligned}$$

Therefore 341 is not prime, but it is pseudoprime with respect to the basis 2.

Miller-Rabin test: We compute k and m:

$$340 = 2^2 \cdot 85$$

from which we obtain $k = 2$ and $m = 85$.

We use the basis $a = 2$

$$\begin{aligned} b_0 &= a^m \bmod n = 2^{85} \bmod 341 = 32 \neq \pm 1 \\ b_1 &= b_0^2 \bmod n = 32^2 \bmod 341 = 1 \end{aligned}$$

Therefore the number 341 is composite. Moreover, we can compute a factor p_1:

$$p_1 = \gcd(b_0 - 1, n) = \gcd(31, 341) = 31$$

Indeed, $341 = 11 \times 31$.

Exercise A.7 Verify whether the number 313 is prime using Miller-Rabin test with the basis 2 and 3.

Solution We compute k and m:

$$312 = 2^3 \cdot 39$$

from which we obtain $k = 3$ and $m = 39$.

We use the basis $a = 2$

$$\begin{aligned} b_0 &= a^m \bmod n = 2^{39} \bmod 313 = 25 \neq \pm 1 \\ b_1 &= b_0^2 \bmod n = 25^2 \bmod 313 = 312 = -1 \end{aligned}$$

We try with the basis $a = 3$

$$b_0 = a^m \bmod n = 3^{39} \bmod 313 = 1$$

We conclude that the number is probably prime.

Exercise A.8 Verify whether the number 17 is prime using Miller-Rabin test with the basis 2 and 3.

Solution We compute k e m:

$$16 = 2^4 \cdot 1$$

from which we obtain $k = 4$ and $m = 1$. We use the basis $a = 2$

$$\begin{aligned} b_0 &= a^m \bmod n = 2^1 \bmod 17 = 2 \neq \pm 1 \\ b_1 &= b_0^2 \bmod n = 2^2 \bmod 17 = 4 \\ b_2 &= b_0^2 \bmod n = 4^2 \bmod 17 = 16 = -1 \end{aligned}$$

We use now the basis $a = 3$

$$\begin{aligned} b_0 &= a^m \bmod n = 3^1 \bmod 17 = 3 \neq \pm 1 \\ b_1 &= b_0^2 \bmod n = 3^2 \bmod 17 = 9 \\ b_2 &= b_0^2 \bmod n = 9^2 \bmod 17 = 13 \\ b_3 &= b_0^2 \bmod n = 13^2 \bmod 17 = 16 = -1 \end{aligned}$$

The number is probably prime.

A.4 Density of Primes

Now we will answer the question whether the likelihood that a randomly picked integer p is a prime is sufficiently high. From looking at the first few positive integers, we know that primes become less dense as the value increases: $2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, \ldots$. The chance that a randomly chosen integer $\overline{p}$ is prime follows from the famous prime number theorem and is approximately $\frac{1}{\ln \overline{p}}$. In practice, we consider only odd numbers, so that the probability is doubled and becomes:

$$P(\overline{p} \text{ is prime}) = \frac{2}{\ln \overline{p}}$$

Exercise A.9 Compute the average number of primality test needed to find a prime number p of 512 bits by random selection among all the 512-bit long odd numbers.

Solution

$$P(\overline{p} \text{ is prime}) = \frac{2}{\ln 2^{512}} = \frac{2}{512 \ln 2} \approx \frac{1}{177}$$

This means that we expect to test 177 random numbers before we find one that is a prime.

Exercise A.10 Assuming that a primality test over a 2048-bit long number takes on average 1 second, compute how many 2048-bit long primes can be found in 1 hour.

Solution

$$P(\overline{p} \text{ is prime}) = \frac{2}{\ln 2^{2048}} = \frac{2}{2048 \ln 2} \approx 0.0014$$

Since 1 hours contains 3600 seconds and each test takes 1 second, we have 3600 trials, thus the average number of primes we can obtain in 1 hour is $3600 \cdot 0.0014 \approx 5$.

$\mathcal{B}$

Quadratic residues

B.1 Chinese remainder theorem

Exercise B.1 Find the solutions of the following congruence:

$$3x \equiv 4 \pmod 7$$

Solution Consider that

$$3^{-1} \cdot 3x \equiv 3^{-1} \cdot 4 \pmod 7$$

Using the extended Euclidean theorem we obtain that

$$3 \cdot 5 = 15 = 7 \cdot 2 + 1 \equiv 1 \pmod 7$$

Therefore, the inverse of 3 is 5. We can write that:

$$\begin{aligned} 5 \cdot 3x &\equiv 5 \cdot 4 \pmod 7 \\ x &\equiv 20 \bmod 7 \equiv 6 \pmod 7 \end{aligned}$$

Therefore, the solution is:

$$x = 6 + 7k, \quad \forall k \in \mathbb{Z}$$

Exercise B.2 Find the solutions of the following system of equations:

$$\begin{cases} x & \equiv 2 \pmod 3 \\ x & \equiv 3 \pmod 5 \\ x & \equiv 4 \pmod{11} \\ x & \equiv 5 \pmod{16} \end{cases}$$

Solution The Chinese remainder theorem guarantees that a solution exists in case $m_i = \{3, 5, 11, 16\}$ are coprime pairs. It is easy to verify that 3, 5 and 11 are prime and that $16 = 2^4$, therefore there are no common factors.

We then define:

$$\begin{aligned}
a_i &= \{2, 3, 4, 5\} \\
m_i &= \{3, 5, 11, 16\} \\
M &= \prod_{i=1}^{4} m_i = 3 \cdot 5 \cdot 11 \cdot 16 = 2640 \\
z_i &= \frac{M}{m_i} \\
y_i &= z_i^{-1} \pmod{m_i} \\
x &= \sum_{i=1}^{4} a_i y_i z_i \bmod M
\end{aligned}$$

In this case:

i	a_i	m_i	z_i	y_i
1	2	3	880	1
2	3	5	528	2
3	4	11	240	5
4	5	16	165	13

Therefore:

$$x = 2 \cdot 1 \cdot 880 + 3 \cdot 2 \cdot 528 + 4 \cdot 5 \cdot 240 + 5 \cdot 13 \cdot 165 \equiv 1973 \pmod{2640}$$

B.1.1 Fast exponentiation with the Chinese Remainder Theorem

Let $n = pq$, where p and q are primes. One way to perform the exponentiation $x^d \bmod n$ efficiently is to exploit CRT in order to compute two individual exponentiations modulo the two “short” primes p and q rather than the “long” modulus n. This can be done in three steps:

1. we reduce the base element x modulo the two factors p and q of the modulus n:

$$x_p \equiv x \mod p$$

$$x_q \equiv x \mod q$$

2. we perform the following two exponentiations:

$$y_p \equiv x_p^{d_p} \mod p$$

$$y_q \equiv x_q^{d_q} \mod q$$

   where the two new exponents are given by:

$$d_p \equiv d \mod (p-1)$$

$$d_q \equiv d \mod (q-1)$$

   Note that both exponents d_p and d_q are bounded by p and q respectively. The same holds for y_p and y_q.

3. compute the final result as:

$$y \equiv [qc_p]y_p + [pc_q]y_q \mod n$$

   where the coefficients c_p and c_q are computed as:

$$c_p \equiv q^{-1} \mod p$$

$$c_q \equiv p^{-1} \mod q$$

Exercise B.3 Let the RSA parameters be given by $p = 11$, $q = 13$, $n = pq = 143$, $e = 7$, $d = e^{-1} = 103 \mod 120$. Compute an RSA decryption for the ciphertext $y = 15$ using the CRT.

Solution We have to compute $y^d = 15 \mod 143$. As first step, we compute:

$$y_p \equiv 15 \equiv 4 \mod 11$$

$$y_q \equiv 15 \equiv 2 \mod 13$$

Then we compute the exponents:

$$d_p \equiv 103 \equiv 3 \mod 10$$

$$d_q \equiv 103 \equiv 7 \mod 12$$

and the following exponentiations:

$$x_p \equiv y_p^{d_p} = 4^3 = 64 \equiv 9 \mod 11$$

$$x_q \equiv y_q^{d_q} = 2^7 = 128 \equiv 11 \mod 13$$

Finally, we obtain the coefficients:

$$c_p \equiv 13^{-1} \equiv 2^{-1} \equiv 6 \mod p$$

$$c_q \equiv 11^{-1} \equiv 6 \mod 13$$

The plaintext follows as:

$$x \equiv [qc_p]x_p + [pc_q]x_q \mod n$$

$$x \equiv [13 \cdot 6]9 + [11 \cdot 6]11 \mod 143$$
$$x \equiv 702 + 276 \equiv 1428 \equiv 141 \mod 143$$

B.2 Quadratic residues

Exercise B.4 Compute $\left(\frac{91}{167}\right)$.

Solution Note that 167 is prime, so we can use the definition of the Legendre symbol:

$$\left(\frac{a}{p}\right) = \begin{cases} +1 & \text{if } a \text{is a square} \quad \pmod p \\ -1 & \text{otherwise} \end{cases}$$

Moreover, a is a square mod$p$ if and only if:

$$a^{\frac{p-1}{2}} \equiv 1 \pmod p$$

Here we must compute (91^{83} mod 167).To perform the calculation we use the Square-and-Multiply algorithm.

i	c_i	$z \pmod{167}$
6	1	$1^2 \times 91 = 91$
5	0	$91^2 = 98$
4	1	$98^2 \times 91 = 53$
3	0	$53^2 = 137$
2	0	$137^2 = 65$
1	1	$65^2 \times 91 = 41$
0	1	$41^2 \times 91 = 166$

Therefore

$$91^{83} \equiv 166 \equiv -1 \pmod{167}$$

So, 91 is not a square and the Legendre symbol has value -1:

$$\left(\frac{91}{167}\right) = -1$$