Advanced Cryptography — Midterm Exam

Solution

Serge Vaudenay

2.5.2017

- duration: 3h
- any document allowed
- a pocket calculator is allowed
- communication devices are not allowed
- the exam invigilators will **not** answer any technical question during the exam
- readability and style of writing will be part of the grade

The exam grade follows a linear scale in which each question has the same weight.

1 DDH Solver in a Group of Order with a Small Factor

We consider a family of cyclic groups G_s generated by some element g_s, where s is the security parameter. The group has order n_s which is divisible by some $m_s > 1$. (In the rest of the exercise, the subscript s is omitted for clarity.) We assume there is a polynomially bounded (in terms of s) algorithm to implement the multiplication in G. We further assume that m is polynomially bounded. The purpose of this exercise is to solve the Decisional Diffie-Hellman (DDH) problem in G.

Q.1 Construct a subgroup H of G with order m.

Let H be the subgroup of all s such that $x^m = 1$. Clearly, H is a subgroup: if $x, y \in H$, then $(xy)^m = x^m y^m = 1$ so $xy \in H$. Furthermore, $1^m = 1$ so $1 \in H$. Finally, $(x^{-1})^m = (x^m)^{-1} = 1$ so $x^{-1} \in H$.
We can show that the order of H is m. Indeed, the cyclic group G is isomorphic to $\mathbf{Z}_n$ so H is isomorphic to the subgroup of $\mathbf{Z}_n$ of all y residues such that $my \bmod n = 0$. This equation is equivalent to $y \bmod \frac{n}{m} = 0$. This is equivalent to y being a multiple of $\frac{n}{m}$. There are exactly m such multiples. So, H has order m.

Q.2 Construct a surjective group homomorphism f from G to H with a polynomially bounded complexity (in terms of s). Describe the algorithm that implements f and prove its complexity.

Using the square-and-multiply algorithm, we can implement $f : x \mapsto x^{\frac{n}{m}}$ with polynomial complexity. Clearly, this is a group homomorphism as

$$f(xy) = (xy)^{\frac{n}{m}} = x^{\frac{n}{m}} y^{\frac{n}{m}} = f(x)f(y)$$

We can see that $f(x) \in H$ as $f(x)^m = x^n = 1$.
We have $f(g) = g^{\frac{n}{m}}$ whose order can only be m as g has order n. So, $f(g)$ generates H. We deduce that f is surjective.

Q.3 Construct a discrete logarithm algorithm in H of polynomial complexity (in terms of s). Describe the algorithm and prove its complexity.

We can compute discrete logarithm by exhaustive search, as the cardinality of H is polynomially bounded.

Q.4 Deduce a DDH distinguisher of polynomial complexity with large advantage. Compute the advantage.

Let $L_{g'}(x')$ be the function computing the discrete logarithm of x' in basis g' in H. We consider the following algorithm.

Input: (g, X, Y, Z)

compute $f(g)$, $L_{f(g)}(f(X))$, $L_{f(g)}(f(Y))$, $L_{f(g)}(f(Z))$
if $L_{f(g)}(f(Z)) \equiv L_{f(g)}(f(X))L_{f(g)}(f(Y)) \pmod{m}$ ***then***
    output 1
else
    output 0
end if

If (g, X, Y, Z) is a DH entry, the output is always 1. If the input (g, X, Y, Z) is random, $f(Z)$ is a uniformly distributed random element of H, independent from the others, so the probability to output 1 is $1/\#H$. So, the advantage is $1 - 1/\#H$. Since H has at least two elements, the advantage is at least $\frac{1}{2}$.

2 MAC vs PRF

In what follows, we consider a function F from $\{0,1\}^{k_s} \times \mathcal{D}_s$ to $\{0,1\}^{\tau_s}$, where s is a security parameter. (For simplicity, s is omitted from notations hereafter.) We can see F either as a Message Authentication Code (MAC) or as a Pseudo Random Function (PRF). By default, we consider chosen message attacks and existential forgeries for the security of MAC functions.

Q.1 Give the following definitions. What does it mean for F to be a secure MAC? What does it mean for F to be a secure PRF?

F is a secure MAC if for any PPT algorithm $\mathcal{A}$,

$$\Pr[\mathcal{A}^{F(K,.)} \text{ forges}] = \mathsf{negl}(s)$$

where $K \in \{0,1\}^k$ is random, (X,t) a pair of random variables defined as the output of $\mathcal{A}^{F(K,.)}$, and "$\mathcal{A}^{F(K,.)}$ forges" is the event that $F(K,X) = t$ and that $\mathcal{A}$ did not query X to the $F(K,.)$ oracle.
F is a secure PRF if for any PPT algorithm $\mathcal{A}$,

$$\Pr[\mathcal{A}^{F(K,.)} \to 1] - \Pr[\mathcal{A}^{F^*(\cdot)} \to 1] = \mathsf{negl}(s)$$

where $K \in \{0,1\}^k$ is random and $F^*(\cdot)$ is a random function from $\mathcal{D}$ to $\{0,1\}^\tau$.

Q.2 If F is a secure PRF and $2^{-\tau}$ is negligible (in terms of s), prove that it is a secure MAC.

We assume that F is a secure PRF. Let $\mathcal{A}$ be a PPT chosen message attack with access to an oracle $\mathcal{O}$ mapping $\mathcal{D}$ elements to $\{0,1\}^\tau$. Let $p = \Pr[\mathcal{A}^{F(K,.)}\ \mathsf{forges}]$. We want to show that $p = \mathsf{negl}(s)$. We define $\mathcal{B}$ as follows:

```
simulate A and forward oracle queries x_i and answers t_i between A and O
eventually, A outputs some (X, t) pair
query O(X) = t'
if X different from all x_i and t = t' then
   output 1
else
   output 0
end if
```

When $\mathcal{O}$ is the oracle $F(K,.)$, $\mathcal{B}$ outputs 1 with probability equal to p. When $\mathcal{O}$ is the oracle F^*, $\mathcal{B}$ outputs 1 with probability $q2^{-\tau}$, where q is the probability that X is different from all x_i. (Indeed, if X differs from all x_i, the value $F^*(X)$ is undetermined so independent from t; the distribution of t' is uniform an independent from t, hence the output is 1 with probability $2^{-\tau}$.) The advantage of $\mathcal{B}$ as a PRF distinguisher is thus $p - q2^{-\tau}$. Since F is a secure PRF, we have $p = q2^{-\tau} + \mathsf{negl}(s)$. Clearly, $q2^{-\tau} \leq 2^{-\tau}$. Assuming that $2^{-\tau}$ is negligible, we deduce that p is negligible.

Q.3 If $2^{-\tau}$ is not negligible (in terms of s), prove that F is not a secure MAC. Describe an attack and analyze its complexity.

We define $\mathcal{A}$ as follows:

```
set X ∈ D arbitrarily
pick t ∈ {0,1}^τ at random with uniform distribution
output (X, t)
```

Clearly, X is not queried to the oracle (there is no query at all). $\mathcal{A}$ forges with probability $2^{-\tau}$. As $2^{-\tau}$ is not negligible, the above attack shows that F is not a secure MAC.

Q.4 Let $0^\tau = (0,\ldots,0) \in \{0,1\}^\tau$. We assume that $2^{-\tau}$ is negligible. Given F (which is from $\{0,1\}^k \times \mathcal{D}$ to $\{0,1\}^\tau$), we consider $G(K,x) = (F(K,x), 0^\tau)$ from $\{0,1\}^k \times \mathcal{D}$ to $\{0,1\}^{2\tau}$.

Q.4a If F is a secure MAC, prove that G is a secure MAC.

We consider a chosen message attack $\mathcal{A}$ against G. Let $p = \Pr[\mathcal{A}^{G(K,.)} \text{ forges}]$. We want to show that $p = \mathsf{negl}(s)$. We define an attack $\mathcal{B}$ against F as follows:

1: *simulate $\mathcal{A}$ but when it makes a query x_i, forward the query x_i, get the answer t_i, and answer $(t_i, 0^\tau)$ to the simulation of $\mathcal{A}$*
2: *eventually, $\mathcal{A}$ outputs some (X, t) pair*
3: ***if** $t = (t', 0^\tau)$ for some t' **then***
4:     *answer (X, t')*
5: ***else***
6:     *abort*
7: ***end if***

Clearly, $\mathcal{B}$ forges with probability p. Since F is a secure MAC, we deduce $p = \mathsf{negl}(s)$.

Q.4b Prove that G is not a secure PRF, even is F is a secure PRF. Describe an attack and analyze its complexity.

We consider the following distinguisher:

1: *set $X \in \mathcal{D}$ arbitrarily*
2: *query X to the oracle and get t*
3: ***if** t ends with τ zeros **then***
4:     *return 1*
5: ***else***
6:     *return 0*
7: ***end if***

When the oracle is $G(K, .)$, the distinguisher always outputs 1. When the oracle is a random function G^*, the distinguisher outputs 1 with probability $2^{-\tau}$. So, the advantage is $1 - 2^{-\tau}$. Since $\tau \geq 1$, the advantage is greater than $\frac{1}{2}$ which is not negligible. (Actually, $2^{-\tau}$ is negligible so the advantage is close to 1.) So, G is not a secure PRF.

3 Distribution in a Subgroup

We consider two odd prime numbers p and q and $g \in \mathbf{Z}_p^*$ an element of order q. Let D_1 be the uniform distribution in $\langle g \rangle$. Let D_2 be the uniform distribution in $\mathbf{Z}_p^*$.

Q.1 Compute d, the statistical distance between D_1 and D_2.

All elements of $\langle g \rangle$ occur with probability $\frac{1}{q}$ resp. $\frac{1}{p-1}$ with D_1 resp. D_2. Others occur with probability 0 resp. $\frac{1}{p-1}$. So,

$$d = \frac{1}{2} q \left| \frac{1}{q} - \frac{1}{p-1} \right| + \frac{1}{2}(p-1-q)\frac{1}{p-1} = 1 - \frac{q}{p-1}$$

Q.2 Construct a distinguisher between D_1 and D_2 with advantage d.

We know from the theory that a best distinguisher would be

```
input: X
if X ∈ ⟨g⟩ then
    return 1
else
    return 0
end if
```

and that its advantage would be d. We can easily show again that the advantage is $1 - \frac{q}{p-1}$: with distribution D_1, the output is always 1. with distribution D_2, the output is 1 with probability $\frac{q}{p-1}$.

Q.3 We assume that 2 has an order bigger than q in $\mathbf{Z}_p^*$. We assume that $p > 2^n$ has n bits and we consider a binary encoding $\mathsf{bin} : \{0,1\}^n \to \mathbf{Z}_p^*$ such that

$$\mathsf{bin}(b_1, \ldots, b_n) = 1 + \sum_{i=1}^{n} b_i 2^{i-1}$$

We use the textbook Diffie-Hellman key exchange to produce a random key K with distribution D_1 between Alice and Bob, following which Alice encrypts a message $x \in \{0,1\}^n$ by sending $y = \mathsf{bin}(x) \times K \bmod p$. Prove that if $x = (b, 0, \ldots, 0)$ where b is uniformly distributed in $\{0,1\}$, we can make a decryption attack in ciphertext-only mode. Propose a countermeasure.

If $b = 0$, then $y = K \in \langle g \rangle$. If $b = 1$, then $y = 2K$. If we had $2K \in \langle g \rangle$, this would imply that $2 \in \langle g \rangle$ but this is not the case as the order of 2 is bigger than q. So, $2K \notin \langle g \rangle$. So, we can deduce b by checking if y belongs to the subgroup. We can do so by checking $y^q = 1$.

We could fix it by using a key derivation function (KDF) and having $y = \mathsf{bin}(x) \times \mathsf{KDF}(K)$.

4 Distinguishers for 3-Round Feistel Schemes

In this exercise, we consider a 3-round Feistel scheme with round functions F_1, F_2, F_3. The input is a pair $x = (x_l, x_r)$ and the output is a pair $y = (y_l, y_r)$. We call x_l and x_r the left input and the right input, respectively. We call y_l and y_r the left output and the right output, respectively. We define

$$z = x_l \oplus F_1(x_r) \quad , \quad y_r = x_r \oplus F_2(z) \quad , \quad y_l = z \oplus F_3(y_r)$$

where $\oplus$ denotes the bitwise exclusive OR. All values are n-bit strings. We assume that F_1, F_2, F_3 are independent uniformly distributed random functions.

Q.1 In the following subquestions, we consider distinguishers between the Feistel scheme and a uniformly distributed random function over $2n$-bit strings which are limited to q chosen input queries.

Q.1a Construct a distinguisher with advantage roughly $\frac{q^2}{2}2^{-n}$.
HINT: Consider a distinguisher making q chosen inputs $x = (x_l, a)$ for a fixed value a and q different values x_l, getting $y = (y_l, y_r)$ and expecting to find two outputs sharing the same y_r. Make a decision based on the obtained input-output pairs.

We consider the following distinguisher:

```
pick a ∈ {0,1}^n arbitrarily
for q pairwise different x_l, query x = (x_l, a) and collect y = (y_l, y_r)
for each pair (x, x') such that x ≠ x' and y_r = y'_r do
    if x_l ⊕ y_l ≠ x'_l ⊕ y'_l then
        return 0
    end if
end for
return 1
```

When querying a Feistel scheme, if $x_r = x'_r$ and $y_r = y'_r$, we notice that

$$x_l \oplus y_l = F_1(x_r) \oplus F_3(y_r) = F_1(x'_r) \oplus F_3(y'_r) = x'_l \oplus y'_l$$

So, the output is never 0 for the Feistel scheme. The probability that the output is 0 for a random function is the probability p_1 that we find a pair (x, x') with $x \neq x'$ and $y_r = y'_r$, multiplied by the probability p_2 that at least one of these pairs satisfies $x_l \oplus y_l \neq x'_l \oplus y'_l$. The advantage is thus $p_1 p_2$.
We have

$$1 - p_1 = (1 - 2^{-n})(1 - 2 \cdot 2^{-n}) \cdots (1 - (q-1) \cdot 2^{-n}) \geq 1 - \frac{q(q-1)}{2} 2^{-n}$$

so $p_1 \approx \frac{q^2}{2}2^{-n}$.
Given a pair, the probability that $x_l \oplus y_l = x'_l \oplus y'_l$ is 2^{-n}. So, $p_2 \geq 1 - 2^{-n}$.
Hence, the advantage is roughly $\frac{q^2}{2}2^{-n}$.

Q.1b Give an upper bound for the advantage of any distinguisher limited to q queries.

> *The Luby-Rackoff Theorem says that the advantage is bounded by $q^2.2^{-n}$. So, the distinguisher from the previous question is close to optimal, if not optimal already.*

Q.2 In this question, we consider a stronger security notion. The adversary has access to the encryption oracle (chosen plaintext) and to the decryption oracle (chosen ciphertext). We consider distinguishers between the Feistel scheme and a uniformly distributed random permutation over $2n$-bit strings which are limited to q chosen plaintext or ciphertext queries.

We consider the following distinguisher:

1: select a nonzero $\delta \in \{0,1\}^n$ arbitrarily
2: pick $x = (x_l, x_r) \in \{0,1\}^{2n}$ at random
3: set $x' = (x_l \oplus \delta, x_r)$
4: query with input x and x' and get $y = (y_l, y_r)$ and $y' = (y'_l, y'_r)$
5: set $y'' = (y_l \oplus \delta, y_r)$
6: query with output y'' and get $x'' = (x''_l, x''_r)$
7: take a decision based on x, y, x', y', x'', y''

Complete the last step to get a very good advantage and estimate it.

> *The distinguisher outputs 1 if and only if*
>
> $$y_r \oplus x''_r = y'_r \oplus x_r$$
>
> *Indeed, for the Feistel scheme, we have*
>
> $$y_r \oplus x''_r = y_r \oplus y''_r \oplus F_2(y''_l \oplus F_3(y''_r)) = F_2(y_l \oplus \delta \oplus F_3(y_r)) = F_2(x_l \oplus \delta \oplus F_1(x_r))$$
>
> *and*
>
> $$y'_r \oplus x_r = x'_r \oplus F_2(x'_l \oplus F_1(x'_r)) \oplus x_r = F_2(x_l \oplus \delta \oplus F_1(x_r))$$
>
> *so the distinguisher always outputs 1.*
> *For the random permutation, x and x' are two different random inputs, so y and y' are two different random outputs. There is a small probability $\frac{1}{2^{2n}-1}$ that the event E that $y'' = y'$ occurs. If not the case, then x'' is a random input different from x and x'. So, x''_r is equal to x_r with probability $\frac{2^n-2}{2^{2n}-2}$ and equal to any other value t with probability $\frac{2^n}{2^{2n}-2}$. Hence,*
>
> $$\Pr[y_r \oplus x''_r = y'_r \oplus x_r] \leq \Pr[E] + \frac{2^n}{2^{2n}-2} \leq \frac{1}{2^{2n}-1} + \frac{2^n}{2^{2n}-2}$$
>
> *So, the advantage is close to 1.*