Cryptography and Security — Midterm Exam
Solution

Serge Vaudenay

10.11.2022

- duration: 1h45
- no documents allowed, except one 2-sided sheet of handwritten notes
- 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
- answers should not be written with a pencil

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

1 Expected Ciphertext Length for Perfect Secrecy

Let $\mathcal{M}$ be a plaintext domain of size $\#\mathcal{M} \geq 2^n$. We define a random plaintext $X \in \mathcal{M}$ of distribution $\mathcal{D}_X$ and a random key $K \in \mathcal{K}$ of distribution $\mathcal{D}_K$. We assume that the support of $\mathcal{D}_X$ is $\mathcal{M}$. Let $\mathsf{Enc}/\mathsf{Dec}$ be a cipher offering *perfect secrecy* for the distributions $\mathcal{D}_X$ and $\mathcal{D}_K$. We assume that the ciphertext $Y = \mathsf{Enc}_K(X)$ is a bitstring of finite length. That is, $X \in \mathcal{M}$, $K \in \mathcal{K}$, and $Y \in \{0,1\}^*$. We denote by $|Y|$ the length of the bitstring Y. The objective of this exercise is to lower bound the expected length of a ciphertext $E(|\mathsf{Enc}_K(x)|)$ for any fixed $x \in \mathcal{M}$ and a random $K \in \mathcal{K}$.

Q.1 In the following subquestions, we consider X uniformly distributed in $\mathcal{M}$ and $k \in \mathcal{K}$ fixed. We define $Y = \mathsf{Enc}_k(X)$.

Q.1a For any i, prove that $\Pr[|Y| \leq i] \leq 2^{i+1-n}$.
HINT: start by proving $\Pr[|Y| = i] \leq 2^{i-n}$.

To be able to decrypt correctly, Enc_k must be an injective function. Hence, there are no more than 2^i plaintexts which encrypt to a ciphertext of length i. Given that X is uniform, we deduce $\Pr[|Y| = i] \leq 2^i/\#\mathcal{M} \leq 2^{i-n}$. Using the geometric sum, we obtain

$$\Pr[|Y| \leq i] \leq (2^{i+1} - 1)2^{-n} \leq 2^{i+1-n}$$

Q.1b Prove that

$$E(|Y|) = (n-1)\Pr[|Y| \leq n-1] + \sum_{i=n}^{+\infty} i \Pr[|Y| = i] - \sum_{i=0}^{n-2} \Pr[|Y| \leq i]$$

We have

$$\begin{aligned}
E(|Y|) &= \sum_{i=0}^{+\infty} i \Pr[|Y| = i] \\
&= \sum_{i=n}^{+\infty} i \Pr[|Y| = i] + \sum_{i=1}^{n-1} i \left(\Pr[|Y| \leq i] - \Pr[|Y| \leq i-1]\right) \\
&= \sum_{i=n}^{+\infty} i \Pr[|Y| = i] + \sum_{i=1}^{n-1} i \Pr[|Y| \leq i] - \sum_{i=0}^{n-2} (i+1) \Pr[|Y| \leq i] \\
&= \sum_{i=n}^{+\infty} i \Pr[|Y| = i] + (n-1) \Pr[|Y| \leq n-1] - \sum_{i=0}^{n-2} \Pr[|Y| \leq i]
\end{aligned}$$

Q.1c Prove that $E(|Y|) \geq n-2$.

We have

$$\begin{aligned}
&E(|Y|) \\
&= (n-1) \Pr[|Y| \leq n-1] + \sum_{i=n}^{+\infty} i \Pr[|Y| = i] - \sum_{i=0}^{n-2} \Pr[|Y| \leq i] \\
&\geq (n-1) \left(\Pr[|Y| \leq n-1] + \sum_{i=n}^{+\infty} \Pr[|Y| = i]\right) - \sum_{i=0}^{n-2} \Pr[|Y| \leq i] \\
&= n-1 - \sum_{i=0}^{n-2} \Pr[|Y| \leq i] \\
&\geq n-1 - \sum_{i=0}^{n-2} 2^{i+1-n} \\
&= n-1 - (2^{n-1}-1)2^{1-n} \\
&\geq n-2
\end{aligned}$$

Q.2 In the following subquestions, we consider X uniformly distributed in $\mathcal{M}$ and we assume that $K \in \mathcal{K}$ follows the distribution $\mathcal{D}_K$. We define $Y = \mathsf{Enc}_K(X)$.

Q.2a Prove that $E(|Y|) \geq n-2$.

Thanks the to the previous questions, we have $E(|\mathsf{Enc}_k(X)|) \geq n-2$ for any k. Hence, $E(|\mathsf{Enc}_K(X)|) \geq n-2$ for K random as well.

Q.2b Prove that the cipher provides perfect secrecy for X uniform in $\mathcal{M}$.
Hint: invoke a theorem from the course.

We have seen in class that perfect secrecy in some distribution of support $\mathcal{M}$ implies perfect secrecy for any distribution of support included in $\mathcal{M}$. This is the case of the distribution of X (which is uniform).

Q.2c Prove that for any $x \in \mathcal{M}$, $E(|\mathsf{Enc}_K(x)|) \geq n-2$.

Since x is in the support of X, we can consider probabilities conditioned to $X = x$. Due to the independence between X and K, we have $\Pr[\mathsf{Enc}_K(x) = y] = \Pr[\mathsf{Enc}_K(X) = y|X = x] = \Pr[Y = y|X = x]$. Due to perfect secrecy, X and Y are independent, so $\Pr[Y = y|X = x] = \Pr[Y = y]$. We deduce that $\mathsf{Enc}_K(x)$ and Y follow the same distribution. Hence, $E(|\mathsf{Enc}_K(x)|) \geq n - 2$.

2 DDH Modulo pq

We consider a probabilistic polynomial-time algorithm $\mathsf{Setup}(1^\lambda) \to (\mathsf{pp}, n, g)$ which takes a security parameter λ and generates a cyclic group of order n and generator g, together with the public parameters pp which are used to define the group operations. We recall the DDH problem based on Setup:

$\mathsf{DDH}(\lambda, b)$

1: $\mathsf{Setup}(1^\lambda) \to (\mathsf{pp}, n, g)$
2: pick $x, y, z \in \mathbf{Z}_n$ uniformly
3: **if** $b = 1$ **then** $z \leftarrow xy$
4: $X \leftarrow g^x$, $Y \leftarrow g^y$, $Z \leftarrow g^z$
5: $\mathcal{A}(\mathsf{pp}, n, g, X, Y, Z) \to t$
6: **return** t

The advantage of the adversary $\mathcal{A}$ playing this game is

$$\mathsf{Adv}_{\mathcal{A}}(\lambda) = \Pr[\mathsf{DDH}(\lambda, 1) \to 1] - \Pr[\mathsf{DDH}(\lambda, 0) \to 1]$$

We have seen in class that the DDH problem is easy if n has any small factor (larger than 1). In this exercise, we wonder what happens if $n = pq$ with p and q large primes. In a "Diffie-Hellman spirit", the group is public and we assume that p and q are public too (hence, provided in pp).

Q.1 In this question, we assume that n has a small prime factor p (to give an idea: a number of $10 \log_2 \lambda$ bits). In the following subquestions, we construct a probabilistic polynomial-time adversary $\mathcal{A}$ with advantage larger than $\frac{1}{2}$.

Q.1a Given a polynomial-time algorithm which takes n as input and find a prime factor p of $10 \log_2 \lambda$ bits, assuming that n has $c.\lambda^\alpha$ bits, for some constants c and α. Precisely estimate its complexity in terms of λ.

> *With a simple sieving technique, the complexity is $2^{\frac{1}{2}\log_2 p}$ arithmetic operations. Arithmetic operations have quadratic complexity in the length of n. This is $\mathcal{O}(\lambda^5 (\log n)^2)$. We can do better by using the ECM method.*

Q.1b Given $w = \frac{n}{p}$, show that it is easy to check if Z^w is the solution to the computational Diffie-Hellman problem with instance (X^w, Y^w) in the subgroup generated by g^w. Assume that T is the complexity of a group multiplication. Precisely estimate its complexity in terms of λ and T.

> *First of all, g^w has order p, which is small. Hence, the baby-step giant-step algorithm computes discrete logarithms in $\mathcal{O}(\sqrt{p})$ group operations (of complexity T). This is $\mathcal{O}(\lambda^5 T)$.*
> *Finally, $\mathcal{A}$ returns 1 if and only if $\log Z^w = (\log X^w) \times (\log Y^w) \bmod p$.*

Q.1c By using the previous questions, construct a polynomial-time adversary $\mathcal{A}$, give its complexity in terms of λ and T and show that it has an advantage in the DDH game close to 1.

Overall, the complexity is dominated by $\mathcal{O}(\lambda^5(T+(\log n)^2))$. The complexity is polynomial.
We have $\Pr[\mathsf{DDH}(\lambda,1)\to 1]=1$ because we always have $\log Z=(\log X)\times(\log Y) \bmod n$ in this case.
We have $\Pr[\mathsf{DDH}(\lambda,0)\to 1]=\frac{1}{p}$ because Z^w is uniform in the subgroup generated by g^w and independent from X and Y.
Hence, the advantage is $1-\frac{1}{p}$ which is close to 1.

Q.2 Let m, p, and q be primes such that $p\neq q$ and pq divides $m-1$. Let $h\in\mathbf{Z}_m^*$ be random and uniformly distributed. Prove that $h^{\frac{m-1}{p}} \bmod m=1$ and $h^{\frac{m-1}{q}} \bmod m=1$ are two independent events of probability $\frac{1}{p}$ and $\frac{1}{q}$ respectively.

Let p^α and q^β the largest powers dividing $m-1$. We write $m-1=p^\alpha q^\beta k$. We know that $\mathbf{Z}_m^*$ is cyclic of order $m-1$, hence isomorphic to $\mathbf{Z}_{m-1}$. Thanks to the Chinese Remainder Theorem, this is isomorphic to $\mathbf{Z}_{p^\alpha}\times\mathbf{Z}_{q^\beta}\times\mathbf{Z}_k$. Let $\Psi:\mathbf{Z}_m^*\to\mathbf{Z}_{p^\alpha}\times\mathbf{Z}_{q^\beta}\times\mathbf{Z}_k$ be a group isomorphism. If h is uniform in $\mathbf{Z}_m^*$, then $\Psi(h)=(h_p,h_q,h_k)$ is uniform in $\mathbf{Z}_{p^\alpha}\times\mathbf{Z}_{q^\beta}\times\mathbf{Z}_k$. The event $h^{\frac{m-1}{p}} \bmod m=1$ is equivalent to $\frac{m-1}{p}(h_p,h_q,h_k)=(0,0,0)$. Since $\frac{m-1}{p}=p^{\alpha-1}q^\beta k$, $\frac{m-1}{p}h_q=0$ is always the case, as well as $\frac{m-1}{p}h_k=0$. Hence, $h^{\frac{m-1}{p}} \bmod m=1$ is equivalent to $\frac{m-1}{q}h_p=0$. Since $q^\beta k$ is invertible modulo p^α, this is equivalent to $p^{\alpha-1}h_p=0$, which is equivalent to $h_p \bmod p=0$, which occurs with probability $\frac{1}{p}$. Similarly, the event $h^{\frac{m-1}{q}} \bmod m=1$ is equivalent to $h_q \bmod q=0$, which occurs with probability $\frac{1}{q}$. As h_p and h_q are independent, the events are independent as well.

Q.3 Given a constant c, we let $f(\lambda)=c.\lambda^3$ be the required bitlength of a modulus m. Construct $\mathsf{Setup}^*(1^\lambda)\to((m,p,q),n,g)$ with $\mathsf{pp}=(m,p,q)$: a probabilitstic polynomial-time algorithm which generates three prime numbers m, p, q such that m is of $f(\lambda)$ bits, p and q are different and of 2λ bits, a number n such that $n=pq$ and n divides $m-1$, and also $g\in\mathbf{Z}_m^*$ which is of order n. Analyze its complexity heuristically.

We use the method seen in class to generate the prime numbers (i.e. keep picking random numbers of appropriate length until one is prime, following a primality test). Then, we take $m = kpq + 1$ by keeping picking a random k until m is prime. Finally, we pick $g = h^k \bmod m$ with h random until neither $g^p \bmod m$ nor $g^q \bmod m$ is equal to 1. We have $g^n \bmod m = 1$ so the order divides n but divides neither p nor q. The order can only be n. The pseudocode is as follows:

$\mathsf{Setup}^*(1^\lambda)$

```
generate a random prime number p of 2λ bits§
generate a random prime number q of 2λ bits§
if p = q then start again
repeat
    pick k of f(λ) − 4λ bits
    m ← kpq + 1
until m is prime
repeat
    pick h ∈ Z*_m at random
    g ← h^k mod m
until g^p mod m > 1 and g^q mod p > 1
return ((m, p, q), n, g)
```

The prime number generation has complexity $\mathcal{O}(\lambda^4)$. The event $p = q$ occurs with negligible probability. The first loop has the same complexity of the prime number generation, i.e. $\mathcal{O}(f(\lambda)^4)$. We have seen in that the events $h^{\frac{m-1}{p}} \bmod m = 1$ and $h^{\frac{m-1}{q}} \bmod m = 1$ are independent and of probability $\frac{1}{p}$ and $\frac{1}{q}$ respectively. Hence, the condition to iterate the second loop occurs with probability $1 - (1 - \frac{1}{p})(1 - \frac{1}{q}) = \frac{1}{p} + \frac{1}{q} - \frac{1}{pq}$ which is negligible. Hence, the second loop is unlikely to iterate. Its complexity is $\mathcal{O}(f(\lambda)^3)$. Overall, the complexity of Setup^* is $\mathcal{O}(f(\lambda)^4)$.

Q.4 Let Setup_1^* be defined by

$\mathsf{Setup}_1^*(1^\lambda)$

$\mathsf{Setup}^*(1^\lambda) \to ((m, p, q), n, g)$

$g_1 \leftarrow g^q \bmod m$

return (m, p, g_1)

We define Setup_2^* similarly. Prove that if DDH is hard for Setup^*, then DDH is hard for Setup_1^* and for Setup_2^*.

We assume that DDH is hard for Setup^* *and we consider an adversary* $\mathcal{A}$ *playing the DDH game with* Setup_1^*. *We construct an adversary* $\mathcal{B}(m, p, q, n, g, X, Y, Z)$ *playing the DDH game with* Setup^* *as follows:*

$\mathcal{B}(m, p, q, n, g, X, Y, Z)$

1: $(g', X', Y', Z') \leftarrow (g^q, X^q, Y^q, Z^q) \bmod m$

2: $\mathcal{A}(m, p, g', X', Y', Z') \to t$

3: ***return*** t

Picking $x, y, z \in \mathbf{Z}_p$ *then* $(X', Y', Z') = (g_1^x, g_1^y, g_1^z)$ *gives the same distribution as picking* $x, y, z \in \mathbf{Z}_n$ *then* $(X', Y', Z') = (g^{qx}, g^{qy}, g^{qz})$. *Similarly, picking* $x, y \in \mathbf{Z}_p$ *then* $(X', Y', Z') = (g_1^x, g_1^y, g_1^{xy})$ *gives the same distribution as picking* $x, y \in \mathbf{Z}_n$ *then* $(X', Y', Z') = (g^{qx}, g^{qy}, g^{qxy})$. *Therefore,* $\mathsf{Adv}_{\mathcal{A}}(\lambda) = \mathsf{Adv}_{\mathcal{B}}(\lambda)$. *By the DDH assumption, this is negligible. Hence, for every* $\mathcal{A}$, $\mathsf{Adv}_{\mathcal{A}}(\lambda)$ *is negligible. The result for* Setup_2^* *follows by a change of notation* $p \leftrightarrow q$.