Advanced Cryptography — Midterm Exam

Serge Vaudenay

10.5.2015

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

1 Recovering a Secret RSA Modulus

Some people use RSA signature with exponent $e = 2^{16} + 1$ but they use too small prime numbers p and q to be secure. So, to prevent n from being factored, they decide to keep $n = pq$ secret. Only legitimate verifiers will receive n.

Q.1 Given a message $m \in \mathbf{Z}_n$ and a valid signature s, show that we can easily recover a multiple of n.

Q.2 What is the complexity?

Q.3 Given a prime number r, what is roughly the probability that r divides the multiple of n recovered in Q.1? (Assume that m is random.)

Q.4 With two message/signature (m_i, s_i) pairs, show that we can recover n with high probability.

2 Finding Four-Term Zero Sums

Looking for collisions is frequent in cryptography. A collision of bitstrings is nothing but a two-term zero sum, using the XOR (denoted by $\oplus$) to define addition. A variant of this problem is to find four-term zero sums. For instance, if we define the signature of a pair of strings (x_1, x_2) of specific format to be the signature of $x_1 \oplus x_2$, we have a forgery attack by looking for a four-term zero sum $x_1 \oplus x_2 \oplus x_3 \oplus x_4 = 0$ with strings x_1, x_2, x_3, x_4 taken from lists of strings of a specific format.

In what follows, we call *a random list of ℓ-bit strings* the sequence $L = (x_1, \dots, x_n)$ obtained by picking all x_i independently uniformly at random in $\{0,1\}^\ell$. We call n the *length* of the list. We denote by $\oplus$ the bitwise XOR operation between bitstrings.

Q.1 Given two lists L_1 and L_2 of length n_1 and n_2, respectively, in the following subquestions, we consider algorithms to find all (i,j) pairs such that the ith element of L_1 and the jth element of L_2 give a XOR of zero.

Q.1a Compute n_3, the expected number of such pairs (i,j).

Q.1b Give an algorithm with complexity $\mathcal{O}(n_1\ell + n_2\ell + n_3 \log \max(n_1, n_2))$ to find these n_3 pairs.

In the following questions, we discard the ℓ factors from the complexities for simplicity. I.e., the cost of copying or comparing ℓ-bit strings is $\mathcal{O}(1)$. Similarly, copying an index i or j is assumed to take $\mathcal{O}(1)$.

Q.1c What is the optimal value for n_1 and n_2 to make $n_3 = 1$ and minimize the complexity at the same time? What is the complexity with these parameters?

Q.2 We denote $L_j = (x_{j,1}, \dots, x_{j,n})$. Given four lists L_1, L_2, L_3, L_4 of same length n, we want to find tuples (i_1, i_2, i_3, i_4) such that $x_{1,i_1} \oplus x_{2,i_2} \oplus x_{3,i_3} \oplus x_{4,i_4} = 0$.

Q.2a What is the expected number of solutions?
Give an efficient algorithm to find them all and its complexity.

Q.2b We now want to find all tuples (i_1, i_2, i_3, i_4) such that $x_{1,i_1} \oplus x_{2,i_2}$ and $x_{3,i_3} \oplus x_{4,i_4}$ have both their b most significant bits equal to zero and $x_{1,i_1} \oplus x_{2,i_2} = x_{3,i_3} \oplus x_{4,i_4}$.
What is the expected number of solutions?
Give an algorithm to find them all with complexity $\mathcal{O}(n + n^2 2^{-b} + n^4 2^{-\ell-b})$.

Q.2c Give an optimal b and n such that we can find one expected tuple with zero XOR.
Give the corresponding complexity.
NOTE: to simplify the computations, allow b to take any real value.

Q.2d What is the complexity to obtain $\alpha \leq n$ solutions instead of just one?
As an application, give n, b, and the complexity for $\alpha = n$.

3 Number of Samples to Distinguish Two Distributions

Given two distributions P_0 and P_1, we recall that the statistical distance $d(P_0, P_1)$ is defined by

$$d(P_0, P_1) = \frac{1}{2} \sum_z |P_0(z) - P_1(z)|$$

We define the Hellinger distance $H(P_0, P_1)$ by

$$H(P_0, P_1) = \sqrt{\frac{1}{2} \sum_z \left(\sqrt{P_0(z)} - \sqrt{P_1(z)} \right)^2}$$

If P is a distribution, we denote by $P^{\otimes n}$ the distribution of the tuple $(X_1, \ldots, X_n)$ where all X_i are independent random variables following the distribution P.

Q.1 Show that

$$H(P_0, P_1) = \sqrt{1 - \sum_z \sqrt{P_0(z)P_1(z)}}$$

Q.2 We have a biased dice with faces numbered from 1 to 6. We consider the distribution P_0 such that $P_0(1) = \frac{1}{6} - \varepsilon$ and $P_0(x) = \frac{1}{6} + \frac{\varepsilon}{5}$ for $x = 2, \ldots, 6$. We consider the uniform distribution P_1.
Compute an asymptotic equivalent of $d(P_0, P_1)$ and $H(P_0, P_1)$ for $\varepsilon \to 0$.
HINT: $\sqrt{1+t} = 1 + \frac{1}{2}t - \frac{1}{8}t^2 + o(t^2)$ when $t \to 0$.

Q.3 Using an upper bound for $d(P_0^{\otimes n}, P_1^{\otimes n})$ in terms of $d(P_0, P_1)$, show that for $n \leq n_{0.5}$, the advantage of any distinguisher between P_0 and P_1 using n samples has an advantage lower than 0.5, where

$$n_{0.5} = \frac{0.5}{d(P_0, P_1)}$$

Q.4 One problem with the previous approach is that we do not know what to say when $n \geq n_{0.5}$. Actually, the bound we obtain is very loose, as we will see.
In the following questions, we estimate $d(P_0^{\otimes n}, P_1^{\otimes n})$ in terms of $H(P_0, P_1)$.

Q.4a Show that $1 - H(P_0^{\otimes n}, P_1^{\otimes n})^2 = (1 - H(P_0, P_1)^2)^n$.
So, as n grows, we can estimate $H(P_0^{\otimes n}, P_1^{\otimes n})$ using $H(P_0, P_1)$ with no loss at all.

Q.4b Show that

$$H(P_0, P_1)^2 \leq d(P_0, P_1) \leq \sqrt{1 - (1 - H(P_0, P_1)^2)^2}$$

HINT: $\left(\sqrt{a} - \sqrt{b}\right)^2 \leq |a - b| = |\sqrt{a} - \sqrt{b}| \times (\sqrt{a} + \sqrt{b})$

Q.4c Show that

$$1 - (1 - H(P_0, P_1)^2)^n \leq d(P_0^{\otimes n}, P_1^{\otimes n}) \leq \sqrt{1 - (1 - H(P_0, P_1)^2)^{2n}}$$

Q.4d Consider that the advantage of the best distinguisher using n samples is an incresing function of n that we extend over the real numbers. Let $n_{0.5}$ be the value of n for which the advantage is 0.5. Show that

$$\frac{0.20}{-\log_2(1 - H(P_0, P_1)^2)} \leq n_{0.5} \leq \frac{1}{-\log_2(1 - H(P_0, P_1)^2)}$$

HINT: $\log_2 \frac{3}{4} \approx -0.4150$.

Q.5 Compare $n_{0.5}$ from Q.3 and Q.4d for the example of Q.2.