Advanced Cryptography — Midterm Exam

Serge Vaudenay

4.5.2010

- all documents are allowed
- a pocket calculator is allowed
- communication devices are not allowed
- answers to the exercises must be provided on a separate sheet
- readability and style of writing will be part of the grade
- do not forget to put your name on your copy!

1 RC4 Biases

The RC4 pseudorandom number generator is defined by a state and an algorithm which update the state and produces an output byte. In RC4, a state is defined by

- two indices i and j in $\mathbf{Z}_{256}$;
- one permutation S of $\mathbf{Z}_{256}$.

By abuse of notation we write $S(x)$ for an arbitrary integer x as for $S(x \bmod 256)$. The state update and output algorithm works as follows:

1: $i \leftarrow i + 1$
2: $j \leftarrow j + S(i)$
3: exchange the values at position $S(i)$ and $S(j)$ in table S
4: output $z_i = S(S(i) + S(j))$

Q.1 Assume that the initial S is a random permutation with uniform distribution and that i and j are set to 0.

Q.1a What is the probability that $[S(1) \neq 2 \text{ and } S(2) = 0]$?

Q.1b If $S(1) \neq 2$ and $S(2) = 0$ hold, show that the second output z_2 is always 0.

Q.1c In other cases, show that $z_2 = 0$ with probability close to $\frac{1}{256}$.
Hint: a 2-line heuristic argument is fine for this question (and this question only).

Q.1d Deduce $p = \Pr[z_2 = 0]$. What do you think of this probability?

Q.2 Let N be an integer. Let now consider a random generator which generate a byte Z such that $\Pr[Z = 0] = p \gg \frac{1}{N}$ and $\Pr[Z = x] = \frac{1-p}{N-1}$ for $0 < x < N$. In every question below, treat the general case then apply it to an example with $N = 256$ and $p = \frac{2}{N}$.

Q.2a Describe a best distinguisher between Z and an unbiased generator based on n samples? Show that there is one making the output 1 if and only if $\frac{k}{n} \geq \tau$ where k is the number of occurrences of $Z = 0$ in the n samples, and

$$\tau = \frac{1}{1 + \frac{\log(pN)}{\log \frac{1-\frac{1}{N}}{1-p}}}$$

is a threshold.
Treat the general case then compute τ in the example.

Q.2b Give a simpler formula to estimate τ when $\frac{1}{N} \ll 1$ and $p \ll 1$.
Treat the general case then compute the estimate in the example.

Q.2c What is the best advantage of a distinguisher when limited $n = 1$?
Treat the general case then compute the advantage in the example.

Q.2d What is the best advantage of a distinguisher when limited $n = 2$? (Assume that $\tau \leq \frac{1}{2}$.)
Treat the general case then compute the advantage in the example.
Hint: reduce to computing $\Pr[k = 0]$.

Q.2e What is the best advantage of a distinguisher when limited $n = \lfloor \frac{1}{\tau} \rfloor$?
Treat the general case then compute the advantage in the example.

Q.2f Show that the Chernoff information between the two distributions is

$$C = -\tau \log_2 \frac{p}{\tau} - (1-\tau) \log_2 \frac{1-p}{1-\tau}$$

where $\alpha = \frac{1/N}{p}$, $\beta = \frac{1-1/N}{1-p}$.
Treat the general case then compute the Chernoff information in the example.
Hint: show that $C = -\log_2 \min \left(p\alpha^\lambda + (1-p)\beta^\lambda \right)$ and that the minimum is reached when

$$\left(\frac{\alpha}{\beta}\right)^\lambda = \frac{1-p}{p} \times \frac{1}{\frac{1}{\tau} - 1}$$

and deduce the optimal λ to compute C.
Hint[2]: if you are afraid of manipulating ugly formulae, consider skipping this question.

Q.2g Deduce an approximation for the required number of samples to distinguish the two distributions.
Treat the general case then compute this number in the example. Discuss the validity of the approximation.

Q.2h Compute the Squared Euclidean imbalance between the two distributions and compare with the Chernoff information.
Treat the general case then compute the SEI in the example. Discuss the validity of the approximation for the Chernoff information.

2 Breaking RSA with Low d Exponent

In this exercise we assume some RSA public key (N, e) and a secret key d such that $ed \bmod \varphi(N) = 1$. We let $pq = N$ be the factorization of N into primes. We assume that p and q are roughly of same length, i.e. $\frac{1}{c}\sqrt{N} \leq p, q \leq c\sqrt{N}$ for some $c \geq 1$ (e.g. $c = 2$). We assume that d is short so that $d \leq N^\alpha$ with $\alpha < \frac{1}{4}$. We will assume $N^\alpha \leq \frac{1}{c}N^{\frac{1}{4}}$. The objective of the exercise is to show that we can recover d from N and e in polynomial time.

Q.1

Q.1a Show that there exists an integer k such that $ed = k(N - p - q + 1) + 1$.

Q.1b Show that $\gcd(k, d) = 1$ and

$$0 \leq \frac{k}{d} - \frac{e}{N} = \frac{k}{d} \times \frac{p + q - 1 - \frac{1}{k}}{N}$$

Q.1c Deduce that if d is such that $d \leq \frac{1}{c}N^{\frac{1}{4}}$ and $c \geq 3$, then $0 \leq \frac{k}{d} - \frac{e}{N} \leq \frac{2}{3d^2}$.
Hint: show that $\frac{p+q}{N} \leq \frac{2}{3d^2}$ and use the result from Q.1b.

In the remaining part of the exercise, we will consider an arbitrary rational number x such that there exist integers μ and ν such that $\gcd(\mu, \nu) = 1$ and $0 \leq \frac{\mu}{\nu} - x \leq \frac{2}{3\nu^2}$. We will show that we can build an algorithm making from x a list of rational numbers containing $\frac{\mu}{\nu}$ in polynomial time in the bitlengths of x. (Note that the bitlength of a rational number is the cumulated bitlength of its numerator and denominator.)

Q.2 Under the assumption that this algorithm is found, deduce that we can recover d in polynomial time.

Q.3 How can we factor N from e and d?

In what follows we forget about RSA and its settings. We only consider the positive rational number x. Given a sequence of integers (or real numbers in Q.4a) $a_0, a_1, \ldots$ such that $a_i > 0$ for all i and an integer n, we define the following notation:

$$[a_0, a_1, \ldots, a_n] = a_0 + \cfrac{1}{a_1 + \cfrac{1}{\ldots + \cfrac{1}{a_n}}}$$

Q.4 Let u and v be the sequences defined by

$$\begin{array}{lll} u_n = a_n u_{n-1} + u_{n-2} & u_{-1} = 1 & u_{-2} = 0 \\ v_n = a_n v_{n-1} + v_{n-2} & v_{-1} = 0 & v_{-2} = 1 \end{array}$$

Q.4a Show that $[a_0, a_1, \ldots, a_n] = \frac{u_n}{v_n}$ for all n.
Hint: first show that $[a_0, a_1, \ldots, a_{n-1}, x] = \frac{xu_{n-1}+u_{n-2}}{xv_{n-1}+v_{n-2}}$ for all real number x and all n.
Hint^2: try with $x = a_n + \frac{1}{x'}$.

Q.4b Show that $u_n v_{n-1} - u_{n-1} v_n = -(-1)^n$ for all n.

Q.4c From now on, we assume that the a_i's are integers. Deduce that $\gcd(u_n, v_n) = 1$ and that $\frac{u_n}{v_n} - \frac{u_{n-1}}{v_{n-1}} = \frac{-(-1)^n}{v_n v_{n-1}}$ for all n.

Q.5 Let $x \geq 0$ be a real number. We define the $a_0, a_1, \ldots$ sequence of *integers* which can be either finite of infinite as follows:

1: let $r \leftarrow x$ and $n \leftarrow -1$

2: **loop**
3: let $n \leftarrow n + 1$
4: let $a_n = \lfloor r \rfloor$
5: exit if $r = a_n$
6: let $r \leftarrow \frac{1}{r - a_n}$
7: **end loop**

We define the sequences u and v from a as before.

Q.5a Show that for all n, x is between $\frac{u_n}{v_n}$ and $\frac{u_{n-1}}{v_{n-1}}$.
Hint: show that $[a_0, a_1, \ldots, a_n, r] = x$ at every iteration of the loop.

Q.5b Show that when x is rational, the algorithm terminates and $x = [a_0, \ldots, a_n]$ when it stops.
Hint: show that $\frac{1}{\nu v_n} \leq \left| x - \frac{u_n}{v_n} \right| \leq \frac{1}{v_{n-1} v_n}$ if $x \neq \frac{u_n}{v_n}$.

Q.5c Deduce that the algorithm terminates if and only if x is rational.

Q.5d Show that every positive rational number can be written $[a_0, a_1, \ldots, a_n]$ with a_i positive integers, $a_i \neq 0$ for $i > 0$, and $a_n \geq 2$ in the $n > 0$ case.

Q.6

Q.6a Show that

$$[a_0, \ldots, a_{n-1}, a_n + \delta] - [a_0, \ldots, a_{n-1}, a_n] = \frac{\delta(-1)^n}{v_n(v_n + \delta v_{n-1})}$$

Hint: sorry, no hint here.

Q.6b Assume that $\mathsf{gcd}(\mu, \nu) = 1$. We denote $x' = [a'_0, \ldots, a'_n]$ the result from the algorithm with $x' = \frac{\mu}{\nu}$ instead of x. Prove that if $0 \leq x - \frac{\mu}{\nu} \leq \frac{2}{3\nu^2}$, then $a_i = a'_i$ for $i < n$ and $a_n = a'_n - (n + 1 \bmod 2)$.
Hint: skip this question.

Q.6c Deduce that if $0 \leq \frac{\mu}{\nu} - x \leq \frac{2}{3\nu^2}$ there exists n such that $\frac{\mu}{\nu} = [a_0, \ldots, a_n + (n+1 \bmod 2)]$.
Hint: assume you did the previous question.

Q.7 By observing that v_n grows faster than a Fibonacci sequence, show that if x is rational, the number of iterations of the previous algorithm is linearly bounded in terms of the bitlength of x.

Q.8

Q.8a Wrap up: show that if x is rational and if there exists μ and ν such that $0 \leq \frac{\mu}{\nu} - x \leq \frac{2}{3\nu^2}$, we can make from x a list of rational numbers containing $\frac{\mu}{\nu}$ in polynomial time.
Hint: just write the algorithm with an explanation about the odd n case.

Q.8b Show that the following algorithm breaks RSA within a linear number of iterations.

1: let $r \leftarrow \frac{e}{N}$, $n \leftarrow -1$
2: let $v_{-1} = 0$, $v_{-2} = 1$
3: let $\rho \leftarrow \mathsf{random}$
4: **loop**
5: let $n \leftarrow n + 1$
6: let $a_n = \lfloor r \rfloor$
7: let $v_n = a_n v_{n-1} + v_{n-2}$
8: let $r \leftarrow \frac{1}{r - a_n}$
9: let $d \leftarrow v_n + v_{n-1} \times (n + 1 \bmod 2)$
10: **if** $\rho^{ed-1} \bmod N = 1$ **then**

11: print d, factor N, and exit
12: **end if**
13: **end loop**