Introduction to Cryptography: Homework 12

December 15, 2021

Requirements about the delivery of this assignment:

- Submit a pdf-document via Brightspace;
- Make sure that you write both name and student number on all documents (not only in the file name).

Deadline: Monday January 3, 17:00 sharp!

Grading: You can score a total of 100 points for the hand-in assignments. To get full points, please **explain all answers clearly and verify computations, e.g., concerning inverses. When using a computer algebra system to compute inverses, we do not grant any points if you simply query inverses**.

Exercises:

1. **Computing $\varphi(n)$ is as hard as factoring n.** Computing $\varphi(n)$ is usually done by factoring n. We have seen in the lecture that factoring integers is hard. In this exercise, we will see that, for $n = pq$, computing $\varphi(n)$ is as hard as factoring n. Therefore, let $p \neq q$ be prime numbers.

   (a) What are the roots of $X^2 - (p+q)X + pq$ over the reals? [The roots are numbers a such that $a^2 - (p+q)a + pq = 0$.]

   (b) Show that when we know both n and $\varphi(n)$, we can determine $p+q$ and pq. [Hint: Express $p+q$ and pq in terms of n and $\varphi(n)$.]

   (c) Let $n = 631349$ and $\varphi(n) = 629760$. Factor n. [You have to use (a) and (b) here; not use a computer algebra system! You can in fact do this with only pen and paper.]

2. **Using the Chinese Remainder Theorem.** Use the method on slide 16 of slides_13_RSA or Garner's method (slide 17 of slides_13_RSA) to solve the following system of modular equations.

$$\begin{cases} x \equiv 9 \pmod{19}; \\ x \equiv 12 \pmod{23}. \end{cases}$$

3. **RSA signatures versus EC Schnorr signatures.** In this exercise, we are going to compare the computational costs of RSA signatures and EC Schnorr signatures (see Figure 1).

Alice		Bob
$\mathcal{E}, G, q, A, a$		$\mathcal{E}, G, q$ (Alice: A)
$v \xleftarrow{\$} \mathbb{Z}/q\mathbb{Z}$, $V \leftarrow [v]G$ $c \leftarrow \mathrm{h}(\mathcal{E}; G; A; V; m)$		
$r \leftarrow v - ca$	$\xrightarrow{m,(r,V)}$	$c \leftarrow \mathrm{h}(\mathcal{E}; G; A; V; m)$ $V \stackrel{?}{=} [r]G + [c]A$

Figure 1: Elliptic-curve Schnorr signatures

For a fair comparison, we consider a prime modulus $n = p'q'$, where p' and q' are prime numbers of $\frac{3072}{2} = 1536$ bits each, such that n is 3072 bits. We also consider an elliptic curve over a field $\mathbb{F}_p$ with a subgroup of order q generated by G, where p and q are prime numbers of 256 bits. Suppose that the RSA public key (used for signature verification) is $e = 2^{16} + 1$ (i.e., $|e| = 17$), and that the corresponding private key is 3072 bits long, and has $\frac{3072}{2} = 1536$ ones.

(a) Suppose that you compute an RSA signature with the square-and-multiply algorithm. How many squarings and how many multiplications do you need to perform?

(b) Suppose that you verify an RSA signature with the square-and-multiply algorithm. How many squarings and how many multiplications do you need to perform?

To compare the costs, we need to know the costs of multiplications $\mathbf{M}_p$ and squarings $\mathbf{S}_p$ in $\mathbb{F}_p$, multiplications $\mathbf{M}_n$ and squarings $\mathbf{S}_n$ in $\mathbb{Z}/n\mathbb{Z}$, and doublings $\mathbf{D}$ and additions $\mathbf{A}$ (on the elliptic curve). For this exercise, you may assume that $\mathbf{M}_p = \mathbf{S}_p = \mathbf{M}_q = \mathbf{S}_q$, $\mathbf{M}_n = \mathbf{S}_n$, $\mathbf{M}_n \approx 50\mathbf{M}_p$, $\mathbf{D} = 7\mathbf{M}_p$ and $\mathbf{A} = 16\mathbf{M}_p$. You may also assume that we have used the same hash function h for both signature schemes. Furthermore, you may also assume that the costs of computing one addition in $\mathbb{Z}/q\mathbb{Z}$ is negligible compared to computing a multiplication in $\mathbb{Z}/q\mathbb{Z}$.

(c) Express the costs of computing an RSA signature in the number of multiplications $\mathbf{M}_p$ and the number of hash computations $\mathbf{h}$.

(d) Express the costs of verifying an RSA signature in the number of multiplications $\mathbf{M}_p$ and the number of hash computations $\mathbf{h}$.

In exercise 4(f) of Assignment 12, you've expressed the costs of computing $[x]G$ in the number of multiplications $\mathbf{M}_p$. If you were not able do this exercise, you may assume that the final answer to (f) was 3830 $\mathbf{M}_p$.

(e) Express the costs of computing a Schnorr signature in the number of multiplications $\mathbf{M}_p$ and the number of hash computations $\mathbf{h}$.

(f) Express the costs of verifying a Schnorr signature in the number of multiplications $\mathbf{M}_p$ and the number of hash computations $\mathbf{h}$.

(g) Repeat (a) and (c), but instead use the Chinese Remainder Theorem to speed up the computation. You may assume that $\mathbf{M}_{p'} = \mathbf{M}_{q'} = \mathbf{S}_{p'} = \mathbf{S}_{q'} \approx 17\mathbf{M}_p$, and d_p and d_q are $\frac{3072}{2} = 1536$ bits long and half of the bits are 1.

Hand in assignments

Reminder: When using a computer algebra system to compute inverses, we do not grant any points for querying inverses! For example you will not get any points when you simply query $9^{-1} \pmod{139}$ and give 31 as an answer.

1. **(30 points) Textbook RSA encryption/decryption.** Alice and Bob want to exchange messages using the textbook RSA encryption scheme with $n = 437$ (note: $437 = 19 \cdot 23$).

   (a) Compute $\varphi(437)$. 5 pt

   (b) What is $\#(\mathbb{Z}/437\mathbb{Z})^*$? 5 pt

   (c) Assume that Alice's public key is $A = (n, e) = (437, 7)$. Bob uses Alice's public key to encrypt the message $m = 104$. Compute this ciphertext. 5 pt

   (d) Compute Alice's private key d. 10 pt

   (e) Alice's receives the ciphertext $C' = 384$. Decrypt C' using Alice's private key. 5 pt

2. **(45 points) Using CRT on textbook RSA decryption.** Alice and Bob encrypt data to each other using textbook RSA encryption. As in hand in assignment 1, Alice's public key is $(n, e) = (437, 7)$. The ciphertext Alice receives from Bob is $C = 384$.

   (a) Determine Alice's private keys d_p and d_q. [Hint: $d_p \equiv e^{-1} \pmod{p-1}$. Maybe have a look at the Euler's totient theorem.] 14 pt

   (b) Compute $m_p \equiv c_p^{d_p} \pmod{p}$ and $m_q \equiv c_q^{d_q} \pmod{q}$. [Hint: $c_p \equiv C \pmod{p}$.] 14 pt

   (c) Retrieve the plaintext from m_p and m_q. [Hint: Apply CRT to m_p and m_q.] 17 pt

3. **(25 points) Textbook RSA encryption and textbook RSA signatures.**

   (a) Is textbook RSA encryption IND-CPA secure? Explain your answer! 10 pt

   (b) In the lecture, we saw that when Eve intercepts two messages $m_1 \neq m_2$ signed by Alice with s_1, respectively s_2, she can create a forgery by setting $m_3 = m_1 \cdot m_2 \pmod{n}$ and $s_3 = s_1 \cdot s_2 \pmod{n}$. Show that Eve can create a forgery when only intercepting one message m_1 signed by Alice with s_1. 15 pt