The Art of Cryptography: Integral Lattices, summer 2010

PROF. DR. JOACHIM VON ZUR GATHEN, DANIEL LOEBENBERGER

9. Exercise sheet
Hand in solutions until Sunday, 20 June 2010, 23:59h.

For this exercise sheet you need a running implementation of the nearest hyperplane algorithm. On our course page you find such an implementation in Matlab/MuPAD which you may use.

Exercise 9.1 (The hidden number problem). (10 points)

You are given the prime $p = 12157665459056928919$, i.e. $\ell = \lceil 5\sqrt{\log_2 p} \rceil = 40$ and $n = \lfloor \sqrt{\log_2 p}/2 \rfloor = 3$. As in the lecture let $\varrho_p(x)$ denote for $x \in \mathbb{Z}$ the balanced representative of $x \bmod p$. The input for your hidden number problem is 10

$$t = (5595231179371318634, 3331525485394863766, 11472294169172514772)$$

and

$$v = (5668021504761479021, -1752142242764252526, 18459420707638161 23).$$

You know that there is $u \in \mathbb{Z}_p$ such that $|v_i - \varrho_p(ut_i)| \leq p2^{-\ell}$ for $1 \leq i \leq n$. Find it.

Exercise 9.2 (The dark side of the HNP: Attacking DSA). (19 points)

The *digital signature algorithm* is one of the main standards for digital signatures. It is defined as follows: $p, q \geq 3$ are prime numbers, q is a divisor of $p-1$. For a rational number z and $m \geq 1$ we denote by $R_p(z)$ the unique integer y with $0 \leq y < m$, such that $y \equiv z \pmod{m}$, provided the denominator of z is relatively prime to m. For simplicity the message m is an element of $\mathbb{F}_q$ (even though one would in real life employ a so-called hash function that maps an arbitrary message to $\mathbb{F}_q$). Let $g \in \mathbb{F}_p$ have multiplicative order q, i.e. q is the smallest integer for which we have $g^q = 1$. p, q, g and m are publicly known. The signer's secret key is an element $\alpha \in \mathbb{F}_q^\times$. This key is typically set up once and then used for a long time. The signature scheme is completetly broken if one can reconstruct it (since it would allow anyone to sign on behalf of the owner of the key α). Now in order to sign a message we select randomly a temporary secret $k \in \mathbb{F}_q^\times$ and compute

$$\begin{aligned} r(k) &= R_q(R_p(g^k)) \\ s(k,m) &= R_q(k^{-1}(m + \alpha r(k))) \end{aligned}$$

The pair $(r(k), s(k,m))$ is the DSA signature of the message m using the secret key α and the temporary secret k. It turns out that it is extremely important to keep all information about k secret! We will see now that if we are given the ℓ least significant bits $a := k \bmod 2^\ell$ of the temporary secret $k = b \cdot 2^\ell + a$ (for various k) we will be able to reconstruct the secret key α:

(i) Show that by the definition of the DSA signature we have 2

$$\alpha r(k) = s(k,m)k - m \text{ in } \mathbb{Z}_q.$$

(ii) Show that for $s(k,m) \neq 0$ we can write this as 3

$$\alpha r(k) 2^{-\ell} s(k,m)^{-1} = (a - s(k,m)^{-1} m) 2^{-\ell} + b \text{ in } \mathbb{Z}_q$$

2 (iii) Define the following two elements:

$$\begin{aligned} t(k,m) &= R_q(2^{-\ell} r(k) s(k,m)^{-1}) \\ v(k,m) &= R_q(2^{-\ell}(a - s(k,m)^{-1} m)) \end{aligned}$$

Argue that the attacker can easily compute these two values.

2 (iv) Show that we have $|\alpha t(k,m) - v(k,m)| < q 2^{-\ell}$

3 (v) Explain what we need to do in order to find the secret key α.

7 (vi) On the website you find in the file `dsa-challenge.txt` a real world example (with parameter-sizes that are actually used in the standard) of six signatures of the message $m = 100$ using the DSA standard. For each signature you know the 64 least significant bits of the temporary secrets k used. Find the secret key α.