On the Security of Lenstra's DSA Variant

Serge Vaudenay

Ecole Normale Supérieure — CNRS
Serge.Vaudenay@ens.fr

Abstract

At the Asiacrypt'96 Conference, Lenstra presented a DSA variant which offers some workload advantage for the signer. In this paper, we show some instances of it are not secure unless special care is taken.

1 Lenstra's DSA variant

In the DSA setting, we are given a prime p, a 160-bit prime factor q of $p-1$ and a qth primitive root of unity g. A signer owns a secret key x and a public key $y = g^x \bmod p$. Signing a hashed message h consists in computing

$$r = g^k \bmod p \bmod q \quad \text{and} \quad s = \frac{h - xr}{k} \bmod q$$

where k is a random number (see [1]).

In order to avoid inversion modulo q (namely the computation of $1/k$), Lenstra proposed a pseudorandom generation algorithm able to efficiently generate a pair $(k, 1/k \bmod q)$ in a clever way [2]. Although Lenstra proposed several variants, we consider a precise one which is fairly representative.

In this variant, the signer picks five random 32-bit integers $t_1, \ldots, t_5$ and flip a coin b. If $b = 0$, the signer let

$$k = \frac{t_1 t_2 t_3}{t_4 t_5} \bmod q$$

and if $b = 1$, the signer takes

$$k = \frac{t_1 t_2}{t_3 t_4 t_5} \bmod q$$

We first observe that the number of random bits required seems to fit to the length of k, namely 160 bits.

2 The attack

Our main argument is the following. There are many special (bad) values k which have many possible t_i's representations. This makes the probability the signer picks one of them high, which allows an opponent to crack the k value then to recover the secret key x from s.

Namely, for any 32-bit integer k, we can represent it with any fraction ka/a where a is a random 64-bit integer. Now there are many ways to represent $ka = t_1t_2t_3$ and $a = t_4t_5$. Let assume k is splitted into $k = k_1k_2k_3$. The number of possible splitted $a = t_1t_2t_3$ is equal to $2^{96}/k$ times the number of possibilities to split a into t_4t_5, which is equal to the number of 32-bit factors of a. Since the probability that a 32-bit number d divides a is $1/d$, the average number of 32-bit factors is approximated by $\log 2$. So we obtain the number of representation of a 32-bit integer is roughly

$$\binom{34}{2} \times 2^{96} \times \log 2 \approx 2^{104.6}$$

If we consider all the inverse k of a 32-bit integer, we obtain that the probability of picking such a bad k is $2^{-55.4}$.

So if we think and opponent precomputed all the r values which correspond to a bad k in a hash table, we he get a new signature, the probability it discloses x is $2^{-55.4}$. This probability may be still improved if we consider other bad k's like values obtain with t_i's that have many small factors which can be rearranged in many ways.

3 Thwarting the attack

In Lenstra's paper, there is an unexplained indication which suggests to use t_i's with large prime factors, so that the number of rearrangement is small.

Concretely, if the t_i's are all prime and pairwise distinct, the number of representation of a possible k can only be 14 or 28 depending whether the two representations are possible or not. This correesponds to the number of permutations of the t_i's. The problem is that the number of random bits is too small. This may still be a good way to generate k's uniformly in a special subset.

4 Further work

The probability estimates in this paper maybe improved. Basically, it would be interesting to study all bad k values and find out how random t_i's are bad. For a more positive point of view, the systematic study of the distribution of the generated k's may gives a proof of security. We encourage further work in this direction.

References

[1] U.S. Department of Commerce, National Institute of Standards and Technology. Digital Signature Standard. Federal Information Processing Standard Publication 186, 1994.

[2] Generating standard DSA signatures without long inversion. In *Advances in Cryptology ASIACRYPT'96*, Kyongju, Korea, Lectures Notes in Computer Science 1163, pp. 57–64, Springer-Verlag, 1996.