On the Big Gap Between $|p|$ and $|q|$ in DSA

Zhengjun Cao

Department of Mathematics, Shanghai University,

Shanghai, China, 200444. caozhj@shu.edu.cn

Abstract We introduce a message attack against DSA and show that the security of DSA is indeed reduced to the following problem, i.e., find $\theta \in \mathbb{Z}_q^*$ such that

$$z = (\hat{g}^{\theta} \bmod p) \bmod q$$

where $\mathrm{Ord}_p(\hat{g}) = q$ and $z \in \mathbb{Z}_q^*$ is randomly chosen by the adversary. Compared with the common key-only attack, i.e., find $x \in \mathbb{Z}_q^*$ such that

$$y = g^x \bmod p$$

the message attack is more effective because of the big gap between $|p|$ (1024-bit) and $|q|$ (160-bit).

Keywords DSA, Schnorr's signature, message attack.

1 Introduction

In 1984, ElGamal [2] proposed a famous cryptographic mechanism, which can be used for both digital signatures and encryption. It is hypothesized that the ElGamal mechanism gets its security from the difficulty of calculating discrete logarithms in a finite field. In the past two decades, many variations of the ElGamal signature have been proposed, including DSA [3] (the digital signature algorithm of U.S.), GOST [4] (the digital signature algorithm of Russia). Another well-known variant of the ElGamal signature scheme is the Schnorr signature scheme [7].

Schneier [8] claimed that DSA is not a derivative of Schnorr, nor even of ElGamal. All three are examples of the general construction of discrete-logarithm-based digital signatures. Then a problem should be raised. Are all the variants equally secure?

We observe that there is a marked difference between many variations of the ElGamal signature and Schnorr's signature. We now only use DSA to illustrate this difference. See the following table.

	PK	Signature	Verification
DSA	(p, q, g, y)	(r, s)	$r = (g^{\mathcal{H}(m)s^{-1}} y^{rs^{-1}} \bmod p) \bmod q.$
Schnorr	(p, q, g, y)	(r, s)	$\mathcal{H}(m \| g^s y^{-r} \bmod p) = r$

It's clear that the latter evaluates the hash function at (m, r, s) rather than just at m, where the digest r is either an input parameter of the hash function $\mathcal{H}$ or the unique output of it.

In this paper, we introduce a message attack against DSA and show that the security of DSA is indeed reduced to the following problem, i.e., find $\theta \in \mathbb{Z}_q^*$ such that

$$z = (\hat{g}^{\theta} \bmod p) \bmod q$$

where $\mathrm{Ord}_p(\hat{g}) = q$ and $z \in \mathbb{Z}_q^*$ is randomly chosen by the adversary. Compared with the common key-only attack, i.e., find $x \in \mathbb{Z}_q^*$ such that

$$y = g^x \bmod p$$

the message attack is more effective because of the big gap between $|p|$ (1024-bit) and $|q|$ (160-bit). The running time of this approach is $O(\frac{\sqrt{q}}{|\mathcal{S}_{m,\rho}|})$, where $\mathcal{S}_{m,\rho} \stackrel{\text{def}}{=} \mathcal{S}_{(\rho)} \cap \mathcal{S}_{(m,\rho)}$,

$$\mathcal{S}_{(\rho)} \stackrel{\text{def}}{=} \left\{\hat{g}^t \bmod p \ : \ 0 \le t \le q-1,\ \hat{g} = g^{\rho} y \bmod p\right\}, \quad |\mathcal{S}_{(\rho)}| = q \approx 2^{160}$$

$$\mathcal{S}_{(m,\rho)} \stackrel{\text{def}}{=} \left\{z + iq \ : \ 0 \le i \le \left[\frac{p-z}{q}\right],\ z = \mathcal{H}(m)\rho^{-1} \bmod q\right\}, \quad |\mathcal{S}_{(m,\rho)}| = \left[\frac{p-z}{q}\right] + 1 \approx 2^{864}$$

We also definitely point out that the Schnorr's signature scheme is free of the attack because it adopts a special self-feedback mode. It evaluates the hash function at (m, r, s) rather than just at m.

2 Description of DSA

In August of 1991, the U.S. National Institute of Standards and Technology (NIST) proposed a digital signature algorithm (DSA). The DSA has become a U.S. Federal Information Processing Standard (FIPS 186) called the Digital Signature Standard (DSS), and is the first digital signature scheme recognized by any government. DSS was selected by NIST, in cooperation with the NSA, to be the digital authentication standard of the U.S. government. The standard was issued in May 1994 [3].

The signature mechanism requires a hash function $\mathcal{H} : \{0,1\}^* \longrightarrow \mathbb{Z}_q^*$ for some integer q. The DSS explicitly requires use of the Secure Hash Algorithm (SHA-1). It's universally believed that the security of the DSA relies on two distinct but related discrete logarithm problems. One is the logarithm problem in $\mathbb{Z}_p^*$. The other is the logarithm problem in the cyclic subgroup of order q.

The description of DSA can be related as follows.

Public key p : 512-bit to 1024-bit prime. q : 160-bit prime factor of $p-1$. g : a base element of order q mod p. y : $= g^x \bmod p$.

Private Key $x \in \mathbb{Z}_q^*$ (a 160-bit number).

Signing(1) Select a random secret integer $k \in \mathbb{Z}_q^*$. (2) Compute $r = (g^k \bmod p) \bmod q$, $s = k^{-1}(\mathcal{H}(m) + xr) \bmod q$. (3) The signature for message m is the pair (r, s).

Verifying Accept it if and only if

$$(g^{\mathcal{H}(m)s^{-1}} y^{rs^{-1}} \bmod p) \bmod q = r$$

where s^{-1} is computed in $\mathbb{Z}_q^*$.

3 Common attacks

It's well-known that there are two common attacks against DSA [8]:

(1) Key-only attack against k or x. We know each signature requires a new value of k, and the value must be chosen randomly. If Eve ever recovers a k that Alice used to sign a message, perhaps by exploiting some properties of the random-number generator that generated k, she can recover Alice's private key x. If Eve ever gets two messages signed using the same k, even if she doesn't know what it is, she can recover x. And with x, Eve can generate undetectable forgeries of Alice's signature. In any implementation of the DSA, a good random-number generator is essential to the system's security.

(2) Dangers of a common modulus. A common modulus too easily becomes a tempting target for cryptanalysis. It is still too early to tell much about different DSS implementations, but there is some cause for concern.

Here is a more complete description of what is referred to as "key-only attack against k":

1. Given a signature (r, s) on a message m, find k such that $r = (g^k \bmod p) \bmod q$.
2. From k, recover the private key $x = (ks - H(m))/r \bmod q$.

4 A new message attack and its effectiveness

4.1 Basic idea

Now we present a more effective attack against DSA. We should stress that it's a message attack instead of a key-only attack.

By the verifying phase in DSA, we have:

$$r = \left(g^{\mathcal{H}(m)s^{-1}} y^{rs^{-1}} \bmod p\right) \bmod q$$

Suppose that $r = \left(g^{\alpha} y^{\beta} \bmod p\right) \bmod q$

Hence, set

$$
\begin{aligned}
(g^{\alpha} y^{\beta})^{s} &= g^{\mathcal{H}(m)} y^{(g^{\alpha} y^{\beta} \bmod p)} \bmod p \\
\alpha s &= \mathcal{H}(m) \bmod q \\
\beta s &= (g^{\alpha} y^{\beta} \bmod p) \bmod q
\end{aligned}
$$

Thus, $\mathcal{H}(m) = \alpha\beta^{-1} \left(g^{\alpha} y^{\beta} \bmod p\right) \bmod q$.

Given a message m, it suffices for the adversary to find $\alpha, \beta \in \mathbb{Z}_q^*$ such that

$$\mathcal{H}(m) = \alpha\beta^{-1} \left(g^{\alpha} y^{\beta} \bmod p\right) \bmod q \qquad (*)$$

He then computes

$$r = \left(g^{\alpha} y^{\beta} \bmod p\right) \bmod q, \qquad s = r\beta^{-1} \bmod q$$

The resulting signature for the message m is (r, s).

4.2 Effectiveness

Compared with the common key-only attack, i.e., find $x \in \mathbb{Z}_q^*$ such that

$$y = g^{x} \bmod p \qquad (1)$$

the Eq.$(*)$ is less intractable than Eq.(1).

In fact, the Eq.$(*)$ is reduced to

$$\mathcal{H}(m) = \rho \left((g^{\rho} y)^{\beta} \bmod p\right) \bmod q$$

if take $\alpha \equiv \rho\beta \bmod q$, where ρ is selected by the adversary in advance. Since the order of g relative to p is q and $y \equiv g^{x} \bmod p$, the above equation can be written as follows

$$\mathcal{H}(m) = \rho \left(\hat{g}^{\beta} \bmod p\right) \bmod q$$

where $\mathrm{Ord}_p(\hat{g}) = q$. It's clear that we only need to consider the following problem

$$z = \left(\hat{g}^{\theta} \bmod p\right) \bmod q \qquad (2)$$

for a given $z = \mathcal{H}(m)\rho^{-1} \bmod q$. That means it only needs to find $\theta \in \mathbb{Z}_q^*$ satisfying an arbitrary equation listed below

$$
\begin{aligned}
\hat{g}^{\theta} \bmod p &= z \\
\hat{g}^{\theta} \bmod p &= z + q \\
&\vdots \\
\hat{g}^{\theta} \bmod p &= z + \left[\frac{p - z}{q}\right] q
\end{aligned}
$$

Define the following two sets

$$\mathcal{S}_{(\rho)} \stackrel{\text{def}}{=} \left\{\hat{g}^t \bmod p \ : \ 0 \le t \le q-1, \ \hat{g} = g^\rho y \bmod p\right\}$$

$$\mathcal{S}_{(m,\rho)} \stackrel{\text{def}}{=} \left\{z + iq \ : \ 0 \le i \le \left[\frac{p-z}{q}\right], \ \ z = \mathcal{H}(m)\rho^{-1} \bmod q\right\}$$

Note that $|\mathcal{S}_{(\rho)}| = q \, (\approx 2^{160})$, $|\mathcal{S}_{(m,\rho)}| = \left[\frac{p-z}{q}\right] + 1 \, (\approx 2^{1024-160})$. Thus $|\mathcal{S}_{(m,\rho)}| \gg |\mathcal{S}_{(\rho)}|$.

Define the following set

$$\mathcal{S}_{m,\rho} \stackrel{\text{def}}{=} \mathcal{S}_{(\rho)} \cap \mathcal{S}_{(m,\rho)}$$

Then the number of solutions to Eq.(2) is $|S_{m,\rho}|$. Thus the running time of this approach is $O(\frac{\sqrt{q}}{|\mathcal{S}_{m,\rho}|})$.

The full version of the above message attack can be described as follows.

A message attack against DSA

Input: $p, q, g, y, \mathcal{H}; m$
Output: (r, s)
1. Pick a random $\rho \in \mathbb{Z}_q^*$.
2. $z \leftarrow \mathcal{H}(m)\rho^{-1} \bmod q$.
3. Use Pollard's rho algorithm to compute $\theta \in \mathbb{Z}_q^*$ such that $z = \left((g^\rho y)^\theta \bmod p\right) \bmod q$.
4. $r \leftarrow z, \ \ s \leftarrow r\theta^{-1} \bmod q$.

Correctness.

$$\begin{aligned} r = z &= \left((g^\rho y)^\theta \bmod p\right) \bmod q \\ &= \left(g^{\theta\rho} y^\theta \bmod p\right) \bmod q \\ &= \left(g^{\theta\mathcal{H}(m)r^{-1}} y^{\theta r^{-1} r} \bmod p\right) \bmod q \\ &= \left(g^{\mathcal{H}(m)s^{-1}} y^{rs^{-1}} \bmod p\right) \bmod q \end{aligned}$$

Remark 1. Note that the adversary may fail to find $\theta \in \mathbb{Z}_q^*$ such that

$$\mathcal{H}(m) = \rho \left((g^\rho y)^\theta \bmod p\right) \bmod q$$

for a given message m and $\rho \in \mathbb{Z}_q^*$. This means the forgery is Selective Forgery (*an adversary is able to create a valid signature for a particular message or class of messages*)[6]. But we should point out that a given message m can always be padded as $\hat{m}$ so that $\mathcal{H}(\hat{m}) = \rho \left((g^\rho y)^\theta \bmod p\right) \bmod q$. Thus the forgery is almost universal [6].

4.3 Comparison

Compared with the common key-only attacks against DSA, our message attack has some dramatic advantages. We refer to the following table.

	Key-only attack against x	Key-only attack against k	Our message attack
target	$x \in \mathbb{Z}_q^*$	$k \in \mathbb{Z}_q^*$	$\theta \in \mathbb{Z}_q^*$
equation	$g^x = y \mod p$	$r = (g^k \bmod p) \bmod q$	$\mathcal{H}(m)\rho^{-1} = \left((g^\rho y)^\theta \bmod p\right) \bmod q$
modulus size	p: 1024-bit	p: 1024-bit, q: 160-bit	p: 1024-bit, q: 160-bit
the number of solutions	1	$\lvert\mathcal{S}_{(r)}\rvert$	$\lvert\mathcal{S}_{m,\rho}\rvert$
running time	$O(\sqrt{q})$	$O(\frac{\sqrt{q}}{\lvert\mathcal{S}_{(r)}\rvert})$	$O(\frac{\sqrt{q}}{\lvert\mathcal{S}_{m,\rho}\rvert})$

where

$$\mathcal{S}_{(r)} \stackrel{\text{def}}{=} \left\{g^t \bmod p \,:\, 0 \le t \le q-1\right\} \bigcap \left\{r + iq \,:\, 0 \le i \le \left[\frac{p-r}{q}\right]\right\}$$

$$\mathcal{S}_{m,\rho} \stackrel{\text{def}}{=} \left\{\hat{g}^t \bmod p \,:\, 0 \le t \le q-1,\ \hat{g} = g^\rho y \bmod p\right\} \bigcap \left\{z + iq \,:\, 0 \le i \le \left[\frac{p-z}{q}\right],\ z = \mathcal{H}(m)\rho^{-1} \bmod q\right\}$$

In the key-only attack against k, we should stress that the adversary must know the signature (r, s) on a given message m. The number of solutions only depends on the given value r. In our message attack, however, the adversary can randomly choose the pair (m, ρ). The number of solutions depends on either ρ or $\mathcal{H}(m)$.

Note that the lower bound of $|\mathcal{S}_{m,\rho}|$ is of great importance in our message attack. Regretfully, it appears very hard to make a pure theoretical analysis of it at present.

5 Schnorr's signature and its advantage

5.1 Description

It's well known that DSA is related to Schnorr's signature. This technique employs a subgroup of order q in $\mathbb{Z}_p^*$, where p is some large prime number. The method also requires a hash function $\mathcal{H} : \{0,1\}^* \longrightarrow \mathbb{Z}_q$.

Public key p : a large prime. q : a large prime factor of $p-1$. g : a base element of order q mod p. y : $= g^x \bmod p$.

Private Key $x \in \mathbb{Z}_q^*$.

Signing (1) Select a random secret integer $k \in \mathbb{Z}_q^*$. (2) Compute $e = g^k \bmod p$, $r = \mathcal{H}(m||e)$, $s = xr + k \bmod q$. (3) The signature for message m is the pair (r, s).

Verifying Accept it if and only if

$$\mathcal{H}(m||g^s y^{-r} \bmod p) = r$$

5.2 An advantage

A marked difference between DSA and Schnorr's signature is that the resulting signature (r, s) in the latter should be input into the hash function $\mathcal{H}(\cdot)$, but in DSA the hash function $\mathcal{H}(\cdot)$ is evaluated only as a function of the message m. That is to say, Schnorr's signature adopts the self-feedback mode. Thus an adversary is forced to search for a digest r so that

$$\mathcal{H}(m||g^s y^{-r} \bmod p) = r \qquad \text{for given } p, g, y, m, \mathcal{H}(\cdot) \tag{3}$$

According to some general cryptographic assumptions on $\mathcal{H}(\cdot)$, the problem is very hard although Wang et al. [11] have claimed that SHA-1 is prone to collision.

The challenge in (3) is introduced in Schnorr's signature. In the signature scheme, the resulting signature datum r is tightly bound with the given message m and another signature datum s. An adversary cannot disassemble computational assignments because the output and input of $\mathcal{H}(\cdot)$ should be generated *synchronously*. It's definitely impossible to mount a message attack against it. Regretfully, neither r nor s in DSA is bound to m in this way. The relations among r, s, m are too loose to prevent an adversary from launching a message attack.

In short, Schnorr's signature is more closely related to DLP than DSA. We refer to [9, 10].

6 Conclusion

The exponent q suggested by the NSA cryptographers has a comparatively short length (160-bit) while the modulus p is of 1024-bit length. By the message attack introduced above, we conclude that DSA is not free of the attack although the experiment on it is still in the works.

In 2004, Koblitz and Menezes [5] remarked that:

> It is amazing that none of the DSA opponents noticed that when the Schnorr's signature was modified, the equivalence with discrete logarithms was lost. In other words, there is no argument known that shows that the ability to forge DSA signatures implies the ability to find discrete logs. This reductionist security failure is a much more serious matter than any of the issues that the anti-DSA people raised in 1992 [1]. It is also surprising that apparently none of the NSA cryptographers noticed this possible objection to DSA. If they had, they could have easily fixed it (without any significant loss of efficiency) by having the signer evaluate the hash function at (m, r) rather than just at m.

Intuitively, if the signer evaluates the hash function at (m, r) rather than just at m, the security of DSA becomes more robust.

References

[1] D. Branstad and M. Smid, Responses to comments on the NIST proposed digital signature standard, Crypto'92, LNCS 740, Springer-Verlag, pp. 76-88.

[2] T. ElGamal, A public key cryptosystem and a signature scheme based on discrete logarithms, CRYPTO'84 (LNCS 196), Springer-Verlag, pp. 10-18.

[3] National Institute of Standards and Technology, NIST FIPS PUB 186, Digital Signature Standard, U.S. Department of Commerce, May 1994.

[4] Government Committee of the Russia, GOST R 34.10-94, Gosudarstvennyi Standard of Russian Federation, Information technology, Cryptographic Data Security. Produce and check procedures of Electronic Digital Signature based on Asymmetric Cryptographic Algorithm. 1994. (In Russian)

[5] N. Koblitz and A. Menezes, Another look at provable security, to appear in J. Cryptology; available from http://eprint.iacr.org/2004/152.

[6] A. Menezes, P. Oorschot, S. Vanstone. Handbook of Applied Cryptography, CRC Press, 1996.

[7] C. Schnorr. Efficient signature generation for smart cards, CRYPTO'89, Springer-Verlag, pp. 239-252.

[8] B. Schneier. Applied Cryptography Protocols, algorithm, and source code in C (Second Edition), John Wiley & Sons, Inc. 1996.

[9] D. Pointcheval and J. Stern, Security proofs for signature schemes, Eurocrypt'96, LNCS 1070, Springer-Verlag, pp. 387-398.

[10] P. Paillier and D. Vergnaud, Discrete-log-based signatures may not be equivalent to discrete log, Asiacrypt 2005, LNCS 3788, Springer-Verlag, pp. 1-20.

[11] X. Wang, Y. Yin, H. Yu. Finding Collisions in the Full SHA-1. CRYPTO 2005, LNCS 3621, Springer-Verlag, pp. 17-36.