FURTHER ATTACKS ON SERVER-AIDED RSA CRYPTOSYSTEMS

JAMES MCKEE AND RICHARD PINCH

Abstract. Lim and Lee [5] describe protocols for server-aided RSA digital signatures involving moduli N with special structure: $N = pq$ where p and q are both of order $N^{1/2}$, and $p - 1$ and $q - 1$ have a large common factor β. We describe a method to factor such numbers in time $O\big(N^{1/4}/\beta\big)$ and show that this renders the proposed system insecure.

1. Introduction

Lim and Lee [5] describe protocols for server-aided RSA digital signatures involving moduli N with special structure: $N = pq$ where $p-1$ and $q-1$ have a large common factor β. As usual, p and q are both of order $N^{1/2}$.

The authors claim that "there exists no known algorithm for factoring N (for $|N| \geq 512$) with knowledge of β of size $64 \sim 80$." We shall show that this claim is incorrect: we describe a method to factor such numbers in time $O\big(N^{1/4}/\beta\big)$ which renders the proposed system insecure.

2. The proposed cryptosystem

服务器辅助的RSA签名计算

Lim and Lee [5] discuss server-aided RSA signature computation in the situation where the client does not have the computational power to form an RSA signature $y = x^d \bmod N$, where $N = pq$ and d are known to the client and x is an arbitrary challenge from the server. An example might be a smart-card with restricted computing power as client interacting with an EFTPOS terminal as server.

In such protocols it is assumed that the client will have to use the server to perform certain computations: in this case, modular exponentiations. A number of proposals have been made and attacked.

The proposal in [5] involves use of a blinding factor $r^{-g} \bmod N$, where r and g are random, in order to avoid certain attacks on the protocol by malicious servers. They propose that, in order to speed up precomputation of this blinding factor, the factors p and q of N should be chosen so that $p-1$ and $q-1$ have a common prime factor β, and that the values of r should be randomly chosen from among powers of α, where α is an element of order β in the multiplicative group modulo N. The values of α and β are to be kept secret. It is proposed that for p and q of size around 256 bits, the size of β should be in the range 64 to 80 bits.

3. The first stage of the attack

Assume that $N = pq$ where p and q have size ~ 256 bits and that a prime β of size $64 \sim 80$ bits divides both $p-1$ and $q-1$.

We first note that $N \equiv 1 \bmod \beta$. We partially factor $N-1$ by one of the methods whose running time depend on the size of the prime factor to be extracted, such as

Pollard's $p-1$ or ρ methods [7], [8] or Lenstra's elliptic curve method [3]. Current experience suggests that a prime factor of up to 45 decimal digits can routinely be extracted within a few days computation. Since the proposed size of β is at most 25 decimal digits, we may assume that β is known. Indeed we would need to choose β of size well in excess of 150 bits to guard against this stage, but then applying Pollard's ρ method with the "random" map $x \mapsto x^{N-1} + 3 \bmod N$ would split N in $O\left(\sqrt{p/\beta}\right)$ steps, since there are at most $(p-1)/\beta + 1$ values of $x^{N-1} \bmod p$. With β in excess of 150 bits, $\sqrt{p/\beta} \sim 2^{53}$, and N is vulnerable.

4. The second stage of the attack

We know that $p \equiv q \equiv 1 \bmod \beta$. We can now employ a variant of a method of Lehmer described in [6]. Write $p = x\beta + 1$ and $q = y\beta + 1$, so that $N = xy\beta^2 + (x+y)\beta + 1$. Then $(N-1)/\beta = xy\beta + (x+y) = u\beta + v$ where u and $0 \le v < \beta$ are known and x, y are unknown. We have $x + y = v + c\beta$, $xy = u - c$, where c is the (unknown) carry in expressing $(N-1)/\beta$ in base β.

Finding x and y is equivalent to finding c, since given c we know $x+y$ and xy, and x, y are obtained as the roots of the quadratic equation $(Z-x)(Z-y) = Z^2 - (x+y)Z + xy$.

The discriminant of this quadratic must be a square,

$$(x-y)^2 = (x+y)^2 - 4xy = (v + c\beta)^2 - 4(u-c) = \beta^2c^2 + (2\beta v + 4)c + v^2 - 4u$$

so that a possible candidate for c can be tested quickly: indeed, congruences modulo small primes should suffice to eliminate the majority of the incorrect values.

The range of possible values for c is given by $c\beta \le x + y$: since we are assuming that p and q are of comparable sizes, we have x and y about $\sqrt{N}/\beta$, so that there are of the order of $C = \sqrt{N}/\beta^2$ values of c to test.

For the sizes proposed, $N < 2^{512}$ and $\beta > 2^{64}$, this means that of the order of 2^{128} values of c need to be tested: for $\beta \sim 2^{80}$, this is reduced to 2^{96}. By themselves, these values are too large. We next show how to reduce them significantly.

5. The third stage of the attack

模N的乘群的指数

We observe that $\lambda(N)$, the exponent of the multiplicative group modulo N, is $\lambda(N) = \operatorname{lcm}\{p-1, q-1\} = \operatorname{lcm}\{x\beta, y\beta\}$ and so $\lambda(N)$ divides $xy\beta$. Let a be prime to N. We have

$$a^{u\beta} = a^{xy\beta + c\beta} \equiv a^{c\beta} \bmod N$$

so putting $b = a^\beta$ we have $b^u \equiv b^c$. This equation determines c, which is of magnitude $C = \sqrt{N}/\beta^2$, modulo the order of b in the multiplicative group. With high probability the order of b will be nearly as large as xy, which is of magnitude N/β^2. Hence a solution c to $b^c \equiv b^u \bmod N$ with $c \le C$ is very likely to be the correct value.

We now solve this equation by the "baby-step giant-step" method of Shanks [10]. Let D be an integer larger than $\sqrt{C}$ and form the lists

$$b^0, b^D, b^{2D}, \ldots, b^{D^2} \bmod N$$

and

$$b^u, b^{u-1}, \ldots, b^{u-D} \bmod N.$$

We can sort these lists and find a common value $b^{rD} \equiv b^{u-s}$ in time $O(D^{1+\epsilon})$. Then we recover c as $rD+s$. A low-storage alternative is to use Pollard's λ method [9].

This method will factor N in time $O(C^{1/2+\epsilon})$, that is, in time dominated by $N^{1/4}/\beta$. In the cases under consideration, this will be at most 2^{64} for β of size 64 bits and reduces to 2^{48} for β of size 80 bits. These timings are too low for security and the cryptosystem is insecure.

6. Conclusion

We have described a method for factoring numbers $N = pq$ where p and q are both of order $N^{1/2}$, and $p-1$ and $q-1$ have a common factor β: the method runs in time $O(N^{1/4}/\beta)$ given β. For the parameters suggested in [5], the modulus can be factored with work at most $O(2^{64})$ and the system is insecure.

References

[1] Henri Cohen (ed.), *Algorithmic number theory*, Lecture notes in Computer Science, vol. 1122, Springer–Verlag, 1996, Proceedings, second international symposium, Talence, France, May 1996.

[2] Don Coppersmith (ed.), *Advances in cryptology* — Crypto *'95*, Lecture notes in Computer Science, vol. 963, Berlin, Springer–Verlag, 1995.

[3] Hendrik W. Lenstra jr, *Factoring integers with elliptic curves*, Annals of Math. **126** (1987), 649–673.

[4] Don J. Lewis (ed.), *Number theory institute 1969*, Proceedings of symposia in pure mathematics, vol. 20, Providence RI, Amer. Math. Soc., 1971.

[5] Chae Hoon Lim and Pil Joong Lee, *Security and performance of server-aided RSA computation protocols*, In Coppersmith [2], pp. 70–83.

[6] James F. McKee and Richard G.E. Pinch, *Old and new deterministic factoring algorithms*, In Cohen [1], pp. 217–224.

[7] John M. Pollard, *Theorems on factorization and primality testing*, Proc. Cambridge Philos. Soc. **76** (1974), 521–528.

[8] ———, *A Monte Carlo method for factorization*, BIT **15** (1975), 331–334.

[9] ———, *Monte Carlo methods for index computation (mod p)*, Math. Comp. **32** (1978), 918–924.

[10] Daniel Shanks, *Class number, a theory of factorization and genera*, In Lewis [4], pp. 415–440.

Pembroke College, Oxford
E-mail address: mckee@maths.ox.ac.uk

Queens' College, Cambridge
E-mail address: rgep@cam.ac.uk