Selective Forgery of RSA Signatures with Fixed-Pattern Padding

Arjen K. Lenstra[1] and Igor E. Shparlinski[2]

[1] Citibank, N.A. and Technical University Eindhoven
1 North Gate Road Mendham, NJ 07945-3104, USA
arjen.lenstra@citicorp.com
[2] Department of Computing, Macquarie University
Sydney, NSW 2109, Australia
igor@comp.mq.edu.au

Abstract. We present a practical selective forgery attack against RSA signatures with fixed-pattern padding shorter than two thirds of the modulus length. Our result extends the practical existential forgery of such RSA signatures that was presented at Crypto 2001. For an n-bit modulus the heuristic asymptotic runtime of our forgery is comparable to the time required to factor a modulus of only $\frac{9}{64}n$ bits. Thus, the security provided by short fixed-pattern padding is negligible compared to the security it is supposed to provide.

1 Introduction

Let N be an RSA modulus, and let n denote its bit length. At Crypto 2001 two attacks were presented against RSA signatures with a fixed-pattern padding shorter than $2n/3$: a practical existential forgery attack that runs in time polynomial in n and a selective forgery attack with unspecified non-polynomial runtime [2], see also [5,6,10] for several previously known results. The attack of [2] can, however, not be guaranteed to work for all selected messages. In this paper we extend the selective forgery attack by presenting a modification that works for all selected messages. It can be shown to have heuristic asymptotic expected runtime

$$e^{(1+o(1))(\log N)^{1/3}(\log\log N)^{2/3}}$$

for $N \to \infty$, where log denotes the natural logarithm. Our modified attack works well in practice. This is illustrated by a successful selective forgery attack against a 1024-bit RSA modulus.

Despite a series of increasingly effective attacks, fixed padding RSA signatures remain adopted by several standards. For more details and additional references see [2]. Although the attacks do not extend to random padding, they can be dangerous if the padding is obtained by applying a 'weak' hash-function to the message.

Let d and e satisfy $ed \equiv 1 \bmod \varphi(N)$, where $\varphi(N)$ is the Euler function. The value of d is *private* while N and e are *public*. We assume that the residue ring of integers modulo N is represented by the elements $\mathbf{Z}_N = \{0, 1, \ldots, N-1\}$.

D. Naccache and P. Paillier (Eds.): PKC 2002, LNCS 2274, pp. 228–236, 2002.

A fixed-pattern padding scheme works by concatenating each ℓ-bit message with the same $(n-\ell)$-bit padding. The resulting padded message, which is an element of $\mathbf{Z}_N$, is then signed by computing its d-th power modulo N. It has been shown in [2] that for $\ell > n/3$ the attacker can generate in deterministic polynomial time four interrelated ℓ-bit messages so that valid signatures on three of them can be used to generate a valid signature on the remaining one. This is an existential chosen message attack because there is no control over the message for which the signature is forged. It was shown in [2] that the attack succeeds for 1024-bit RSA moduli.

It has also been described in [2] how the attack can be modified into a selective forgery attack, i.e., an attack where the message whose signature is forged is selected in advance. However, this modification has not been analyzed in detail. For example, neither the general strategy nor the precise runtime have been described, although it is mentioned that it no longer runs in polynomial time. Actually, the selective forgery from [2] works only occasionally. For most selected messages it will not be successful. In this paper we present some considerations which help to facilitate a selective forgery attack. As a result our selective forgery attack may be expected to work for all selected messages. Our analysis includes an optimal choice of parameters and a heuristic estimate of the asymptotic runtime. The runtime is subexponential, but it is much lower than a brute force factorization attack via the number field sieve. Furthermore, we present an example of a successful selective forgery attack against an unfactored 1024-bit RSA modulus.

In Section 2 we review the attacks from [2] and present some additional observations concerning them. In Section 3 we present our alternative approach to the selective forgery attack and analyze its runtime. In Section 4 an example of a successful selective forgery attack is presented. For ease of reference the notation from [2] is maintained as much as possible.

2 Idea of the Attack

For an ℓ-bit message m the fixed-pattern padding is denoted by $R(m) \in \mathbf{Z}_N$. The signature $s(m)$ is defined as

$$s(m) = R(m)^d \bmod N. \tag{1}$$

Following [2], we define

$$R(m) = \omega \cdot m + a \tag{2}$$

where ω and a are the fixed multiplicative and additive redundancies, respectively. Given a fixed $(n-\ell)$-bit padding Π, left-padding $\Pi|m$ is obtained using $\omega = 1, a = 2^\ell \Pi$ and right-padding $m|\Pi$ using $\omega = 2^{n-\ell}, a = \Pi$. Note that in the former case Π and in the latter case m should be small enough to make sure that $R(m) \in \mathbf{Z}_N$.

Let $m_1, \ldots, m_4$ be four distinct ℓ-bit messages such that

$$R(m_1) \cdot R(m_2) \equiv R(m_3) \cdot R(m_4) \bmod N. \tag{3}$$

With (1) it follows that

$$s(m_3) = \frac{s(m_1) \cdot s(m_2)}{s(m_4)} \bmod N$$

unless the inversion modulo N of $s(m_4)$ fails. The latter possibility is ignored from now on, since it would lead to a factor of N. In any case, if equation (3) is satisfied, then the signature $s(m_3)$ of m_3 can be computed given signatures $s(m_1), s(m_2)$, and $s(m_4)$ of m_1, m_2, and m_4, respectively.

With $P = a/\omega \bmod N$ and (2) congruence (3) is equivalent to

$$P(m_3 + m_4 - m_1 - m_2) \equiv m_1 m_2 - m_3 m_4 \bmod N.$$

With

$$t = m_3, \quad x = m_1 - m_3, \quad y = m_2 - m_3, \quad z = m_3 + m_4 - m_1 - m_2 \tag{4}$$

this becomes

$$Pz \equiv xy - tz \bmod N. \tag{5}$$

For an existential forgery attack, integers t, x, y, z satisfying (5) are constructed so that the corresponding $m_1, \ldots, m_4$ are at most ℓ bits long. We describe the construction from [2] in slightly more detail than can be found in [2]. Let $\frac{P_i}{Q_i}$ denote the i-th continued fraction convergent to P/N. Then

$$\left| \frac{P}{N} - \frac{P_i}{Q_i} \right| \le \frac{1}{Q_i Q_{i+1}}.$$

There is an integer j such that $Q_j < N^{1/3} \le Q_{j+1}$. Let $u = |PQ_j - NP_j|$. Then

$$0 < u \le N/Q_{j+1} < N^{2/3} \quad \text{and} \quad Pz \equiv u \bmod N$$

for an integer z with $|z| < N^{1/3}$, namely either $z = Q_j$ or $z = -Q_j$.

Given z, an integer y is selected with $N^{1/3} \le y \le 2N^{1/3}$ and $\gcd(y, z) = 1$. It follows that t can be found such that $0 \le t < y$ and

$$tz \equiv -u \bmod y.$$

With

$$x = (u + tz)/y \le u/y + z \le 2N^{1/3}$$

and $Pz \equiv u \bmod N$, the integers $t, x, y, z \le 2N^{1/3}$ satisfy congruence (5), as desired. From $|z| < N^{1/3}$ and $y \ge N^{1/3}$ it follows that $y + z > 0$, so that $x + t = (u + t(y + z))/y > 0$ since $u > 0$. Therefore, the messages

$$m_1 = x + t, \quad m_2 = y + t, \quad m_3 = t, \quad m_4 = x + y + z + t$$

(cf. (4)) are positive and about ℓ bits long, assuming that $\ell \approx n/3$. Clearly, this attack runs in polynomial time.

For a selective forgery attack, congruence (5) is rewritten as

$$(P + m_3)z \equiv xy \bmod N. \tag{6}$$

Given m_3, integers x, y, z satisfying (6) are sought such that the corresponding m_1, m_2, m_4 are no more than ℓ bits long. In [2] it is suggested to compute the continued fraction expansion of $(P + m_3)/N$, resulting in z, u with $|z| < N^{1/3}$ and $0 < u < N^{2/3}$ such that

$$(P + m_3)z \equiv u \bmod N \tag{7}$$

and to write u as the product xy of two integers x and y of about the same size. Since z and u are almost certainly unique (over a random choice of the message m_3 and the value P that follows from the padding) this attack fails if u cannot be factored in the prescribed way. Indeed, it follows from [7, Theorem 21] that with overwhelming probability a randomly selected integer u does not have a divisor in the range $[u^{1/2-\eta(u)}, u^{1/2+\eta(u)}]$ for any function $\eta(u) \to 0$. It is also useful to recall that for any fixed $0 < \delta \leq 1/2$ the density of the integers u having a prime divisor exceeding $u^{1-\delta}$ is positive. More precisely the density is $-\log(1-\delta)$, see [3]. Thus, for almost all messages m_3 the resulting u simply does not split into two factors of about the same size. This point is not mentioned in [2], and neither is it explained how one should go about factoring u. In Section 3 we address both these problems.

3 Improvements

As usual $L_M(\alpha, \gamma)$ denotes any quantity of the form

$$\exp((\gamma + o(1))(\ln M)^{\alpha}(\ln \ln M)^{1-\alpha})$$

for $M \to \infty$. Then factoring u in congruence (7) using the number field sieve takes about

$$L_{N^{2/3}}\left(1/3, (64/9)^{1/3}\right) = L_N\left(1/3, (128/27)^{1/3}\right)$$

see [9]. However, as noted in Section 2, the selective forgery attack fails if u cannot be factored as xy, with x and y of about the same order of magnitude.

We show that by allowing a little bit of 'slackness' and by working with marginally longer messages of size $O(N^{1/3+\varepsilon})$, one can efficiently produce a sequence of u-values based on a single message m_3. Moreover, this allows us to use the elliptic curve factoring method [11] to quickly search for a u with a large smooth part. Overall we obtain a considerable speedup of the approach that uses the number field sieve.

Fix a small positive ε and let $M = \lfloor N^{1/3} \rfloor$. For a random integer k with $0 < k < N^{\varepsilon}$ apply the continued fraction algorithm to $(P + m_3 - kM)/N$ to find v_k, w_k such that

$$0 \leq v_k \leq N^{1/3}, \qquad 0 \leq |w_k| \leq N^{2/3},$$

and

$$(P + m_3 - kM)v_k \equiv w_k \bmod N.$$

It follows that

$$(P + m_3)v_k \equiv w_k + kMv_k \bmod N.$$

Multiplying both sides by -1 if necessary, we obtain the congruence

$$(P + m_3)z_k \equiv u_k \bmod N$$

with $0 \le |z_k| \le N^{1/3}$ and $0 \le u_k \le 2N^{2/3+\varepsilon}$.

Analyzing the continued fraction algorithm we see that unless most of the fractions $(P+m_3-kM)/N$ admit abnormally good approximations, different values of k are likely to produce different pairs (v_k, w_k), and thus different (z_k, u_k). We remark that a fraction A/N has abnormally good approximations if and only if it has a very large partial quotient in its continued fraction expansion. On the other hand, one easily derives from [12, Theorem 5.10, Theorem 5.17, and (5.11)] that 'on average' over all $A \in \mathbf{Z}_N$ with $\gcd(A, N) = 1$, the largest quotient of A/N is $O(\log^2 N)$. This shows how the value u in (7) can be randomized, thereby solving one problem with the selective forgery attack proposed in [2]. It remains to analyze how many values u_k have to be generated before a 'good' one can be recognized quickly.

A positive integer is Y-smooth if all its prime factors are at most Y. Let $\Psi(X, Y)$ denote the total number of Y-smooth numbers up to X. The following estimate is a substantially relaxed and simplified version of (for example) [8, Corollary 1.3]: for a fixed arbitrary $\delta > 0$, $X \ge 10$, and $\alpha \le (\log X)^{1-\delta}$,

$$\Psi(X, X^{1/\alpha}) = X\alpha^{-\alpha+o(\alpha)} \tag{8}$$

for $\alpha \to \infty$.

From the sequence of u_k values we are interested in those u_k that have a factor exceeding $0.5\sqrt{u_k}$ that is $N^{1/\alpha}$-smooth with

$$\alpha = c(\log N)^{1/3}(\log\log N)^{-1/3}, \tag{9}$$

for a constant c.

We remark that this choice of α optimizes (up to the value of the constant c which we choose later) the trade-off between the number of trials before a 'good' u_k is found and the complexity of finding an $N^{1/\alpha}$-smooth factor of such numbers using the elliptic curve factoring method [11].

Thus $\log\alpha = (1/3 + o(1))\log\log N$ for $N \to \infty$. According to (8) one may expect that there are

$$\Psi(N^{1/3}, N^{1/\alpha}) = \Psi(N^{1/3}, (N^{1/3})^{3/\alpha}) = N^{1/3}(\alpha/3)^{-\alpha/3+o(\alpha)}$$

integers $s \in [(N/2)^{1/3}, N^{1/3}]$ that are $N^{1/\alpha}$-smooth. Also, the number of primes $p \in [N^{1/3+\varepsilon}, 2N^{1/3+\varepsilon}]$ is proportional to $N^{1/3+\varepsilon}/\log N$. Thus, for such s and p, the number of products sp is

$$N^{2/3+\varepsilon}\alpha^{-\alpha/3+o(\alpha)}(\log N)^{-1}.$$

With (9) this becomes

$$N^{2/3+\varepsilon} L_N(1/3, -c/9).$$

It follows that we may expect that one among $L_N(1/3, c/9)$ numbers u_k has an $N^{1/\alpha}$-smooth part that exceeds $0.5\sqrt{u_k}$. Note that $L_N(1/3, c/9) < N^{\varepsilon}$ for any fixed $\varepsilon > 0$ and $N \to \infty$. Using the elliptic curve factoring method [11] the $N^{1/\alpha}$-smooth part of u_k can be found in heuristic expected time

$$L_{N^{1/\alpha}}\left(1/2, \sqrt{2}\right) = L_N\left(1/3, 2\sqrt{1/3c}\right).$$

Therefore the total complexity of finding a 'good' u_k is

$$L_N\left(1/3, c/9\right) L_N\left(1/3, 2\sqrt{1/3c}\right) = L_N\left(1/3, c/9 + 2\sqrt{1/3c}\right).$$

This is minimized for $c = 3$ giving $L_N(1/3, 1)$ for the total heuristic expected runtime. Note that $L_N(1/3, 1)$ is substantially faster than

$$L_N(1/3, (128/27)^{1/3}) \approx L_N(1/3, 1.68),$$

the runtime of the approach that attempts to use the number field sieve to factor u directly. The latter approach is not always successful because u may not split into two factors of about equal size.

Because

$$L_N(1/3, 1) = L_{N^{9/64}}\left(1/3, (64/9)^{1/3}\right) \tag{10}$$

one may be tempted to expect that our selective forgery attack against 1024-bit moduli is easier than factoring 150-bit moduli using the number field sieve. However, (10) is an asymptotic result, useful for understanding the asymptotic growth rate of the runtime of our method, not to obtain absolute runtimes. To illustrate this, implementations of the number field sieve factoring algorithm use very fast sieving-based smoothness tests. Using the elliptic curve factoring method as smoothness test instead leads to the same heuristic asymptotic runtime for the number field sieve, but doing so would make it much slower in practice. Our selective forgery attack is in theory based on the elliptic curve method, but uses in practice a combination of trial division and the elliptic curve method – much faster sieving based smoothness tests do not seem to apply.

Nevertheless, and as shown in Section 4, our method is very practical. On average it turns out that a selective forgery attack against 1024-bit moduli can be expected to be easier than factoring moduli of about 250 bits using the number field sieve. Factoring 250-bit moduli is currently considered to be a triviality. Consequently, obtaining a 1024-bit selective forgery is a simple matter too. This practical result was obtained using a moderately efficient implementation of the elliptic curve method. With more careful coding it should not be hard to improve upon the practical performance of our method.

4 Example

We present a selective forgery attack against $N = \text{RSA-1024}$, the as yet unfactored 1024-bit challenge modulus from RSA Laboratories:

$$\begin{aligned} N &= \text{RSA-1024} \\ &= \texttt{C05748BB FB5ACD7E 5A77DC03 D9EC7D8B B957C1B9 5D9B2060} \\ &\quad \texttt{90D83FD1 B67433CE 83EAD737 6CCFD612 C72901F4 CE0A2E07} \\ &\quad \texttt{E322D438 EA4F3464 7555D62D 04140E10 84E999BB 4CD5F947} \\ &\quad \texttt{A7667400 9E231854 9FD102C5 F7596EDC 332A0DDE E3A35518} \\ &\quad \texttt{6B9A046F 0F96A279 C1448A91 51549DC6 63DA8A6E 89CF8F51} \\ &\quad \texttt{1BAED645 0DA2C1CB}, \end{aligned}$$

see `http://www.rsa.com/rsalabs/challenges/factoring/numbers.html`.

With $\omega = 1$ and $a = 2^{1023} + 2^{365}$ and

$$\begin{aligned} m_3 = \quad & \texttt{167148 0115C7FF 50D924CC 6DD0B4EE AA7C04FD E74073D7} \\ & \texttt{8D010BB2 8DB1B371 C8D2A0E1 EE09EA3E D721BCCE} \end{aligned}$$

(the hexadecimal representation of the first 103 digits of π) a search of a few hours on a 600 MHz PIII laptop using a commercially available implementation of the elliptic curve factoring method produced:

$$\begin{aligned} R(m_1) &= \texttt{80000000 00000000 00000000 00000000 00000000 00000000} \\ &\quad \texttt{00000000 00000000 00000000 00000000 00000000 00000000} \\ &\quad \texttt{00000000 00000000 00000000 00000000 00000000 00000000} \\ &\quad \texttt{00000000 00000000 00000000 00000000 00000000 00000000} \\ &\quad \texttt{00000000 00000000 00002000 12CDBE43 3BF454BD CE9C1D5C} \\ &\quad \texttt{6BEB3D7C DC937495 8CAB854E 56EE8476 F1FF524D 3C5E8E25} \\ &\quad \texttt{5D60E809 04C3DDB8}, \\ R(m_2) &= \texttt{80000000 00000000 00000000 00000000 00000000 00000000} \\ &\quad \texttt{00000000 00000000 00000000 00000000 00000000 00000000} \\ &\quad \texttt{00000000 00000000 00000000 00000000 00000000 00000000} \\ &\quad \texttt{00000000 00000000 00002000 075BC9B1 A93BA55B DC35329E} \\ &\quad \texttt{B66E4BC1 915568BA EEEDC419 E3114231 626E8C21 9DF736BE} \\ &\quad \texttt{12312CF1 C92314A0}, \\ R(m_3) &= \texttt{80000000 00000000 00000000 00000000 00000000 00000000} \\ &\quad \texttt{00000000 00000000 00000000 00000000 00000000 00000000} \\ &\quad \texttt{00000000 00000000 00000000 00000000 00000000 00000000} \\ &\quad \texttt{00000000 00000000 00002000 00167148 0115C7FF 50D924CC} \\ &\quad \texttt{6DD0B4EE AA7C04FD E74073D7 8D010BB2 8DB1B371 C8D2A0E1} \\ &\quad \texttt{EE09EA3E D721BCCE}, \end{aligned}$$

$$R(m_4) = \texttt{80000000 00000000 00000000 00000000 00000000 00000000}$$
$$\texttt{00000000 00000000 00000000 00000000 00000000 00000000}$$
$$\texttt{00000000 00000000 00000000 00000000 00000000 00000000}$$
$$\texttt{00000000 00000000 00002000 1AA5C13B 5A416F97 8EC675C4}$$
$$\texttt{A924CE38 3A44C314 C4DF7204 E5D5197F D2E1363B 98D81A66}$$
$$\texttt{5AF68931 6B749CB9}.$$

Because $R(m_1), \ldots, R(m_4)$ satisfy (3), the signature $s(m_3)$ on the preselected message m_3 can be forged, as desired, if $s(m_1)$, $s(m_2)$, and $s(m_4)$ are known.

5 Conclusion

In this paper we have extended the existential forgery attack against short fixed-pattern padding RSA signatures from [2] to a practical selective forgery attack. Here 'short' means that the fixed-pattern padding is shorter than two thirds of the modulus length. The heuristic asymptotic runtime of our method was shown to be $L_N(1/3, 1)$. It thus provides an example where a runtime of $L_N(1/3, 1)$ is achieved using the elliptic curve factoring method. As noted in [1, Section 4.2], where an earlier example is given, this is rare, as such runtimes are usually associated with Coppersmith's discrete logarithm algorithm for finite fields of fixed small characteristic [4] and the number field sieve [9].

It remains an open question if short fixed-pattern padding RSA signatures can be selectively forged in polynomial time. Neither is it known if attacks exist against longer fixed-pattern padding RSA signatures. It is quite natural to try to build multiplicative relations including more than four signatures (which could be a way to attack longer paddings), however at the moment it is not clear how to approach this. Until these issues are settled, research into the practical malleability of fixed-pattern padding RSA signatures remains an interesting subject because it may shed new light on the properties of RSA, still the world's foremost public key system.

Acknowledgment

The authors thank Allan Steel for his assistance with the elliptic curve factoring method.

References

1. D. Boneh, R.J. Lipton, 'Algorithms for black-box fields and their application to cryptography' *Proc. Crypto'96, Santa Barbara*, Lect. Notes in Comp. Sci., vol 1109, Springer-Verlag, Berlin, 1996, 283–297.
2. E. Brier, C. Clavier, J.-S. Coron and D. Naccache, 'Cryptanalysis of RSA signatures with fixed-pattern padding', *Proc. Crypto'01, Santa Barbara*, Lect. Notes in Comp. Sci., vol. 2139, Springer-Verlag, Berlin, 2001, 433–439.

3. S.D. Chowla and J. Todd, 'The density of reducible integers', *Canad. J. Math.*, **1** (1949) 297–299.
4. D. Coppersmith, 'Fast evaluation of logarithms in fields of characteristic two', *IEEE Trans. Inform. Theory* **30** (1984) 587–594.
5. M. Girault and J.-F. Misarsky, 'Selective forgery of RSA signatures using redundancy', *Proc. Eurocrypt'97, Konstanz*, Lect. Notes in Comp. Sci., vol. 1233, Springer-Verlag, Berlin, 1997, 495–507.
6. M. Girault and J.-F. Misarsky, 'Cryptoanalysis of countermeasures proposed for repairing ISO 9796', *Proc. Eurocrypt'00, Bruges*, Lect. Notes in Comp. Sci., vol. 1807, Springer-Verlag, Berlin, 2000, 81–90.
7. R.R. Hall and G. Tenenbaum, *Divisors*, Cambridge Univ. Press, 1988.
8. A. Hildebrand and G. Tenenbaum, 'Integers without large prime factors', *J. de Théorie des Nombres de Bordeaux*, **5** (1993) 411–484.
9. A.K. Lenstra and H.W. Lenstra, Jr., (Editors), *The developments of the number field sieve*, Lect. Notes in Mathematics, vol. 1554, Springer-Verlag, Berlin, 1993.
10. J.-F. Misarsky, 'A multiplicative attack using LLL algorithm on RSA signatures with redundancy, ', *Proc. Crypto'97, Santa Barbara*, Lect. Notes in Comp. Sci., vol. 1294, Springer-Verlag, Berlin, 1997, 221–234.
11. H.W. Lenstra, Jr., 'Factoring integers with elliptic curves', *Ann. of Math.*, **126** (1987) 649–673.
12. H. Niederreiter, *Random number generation and Quasi–Monte Carlo methods*, SIAM Press, 1992.