Selective Forgery of RSA Signatures Using Redundancy

Marc Girault
marc.girault@francetelecom.fr

Jean-François Misarsky
jean-francois.misarsky@francetelecom.fr

CNET CAEN
42, rue des Coutures
B.P. 6243
FR-14066 CAEN Cedex

Abstract: We show the weakness of several RSA signature schemes using redundancy (i.e. completing the message to be signed with some additional bits which are fixed or message-dependent), by exhibiting chosen-message attacks based on the multiplicative property of RSA signature function. Our attacks, which largely extend those of De Jonge and Chaum [DJC], make extensive use of an affine variant of Euclid's algorithm, due to Okamoto and Shiraishi [OS]. When the redundancy consists of appending any fixed bits to the message m to be signed (more generally when redundancy takes the form of an affine function of m), then our attack is valid if the redundancy is less than half the length of the public modulus. When the redundancy consists in appending to m the remainder of m modulo some fixed value (or, more generally, any function of this remainder), our attack is valid if the redundancy is less than half the length of the public modulus minus the length of the remainder. We successfully apply our attack to a scheme proposed for discussion inside ISO.

1   Introduction

Let (P, S) be a RSA [RSA] key pair, where P is the public function and S the secret one. It is well known that the "reciprocal property" (the fact that $P \circ S = S \circ P = Id$, the identity function) and the "multiplicative property" (the fact that $S(xy) = S(x)S(y)$) of RSA lead to potential weaknesses, especially when used for signatures.

The reciprocal property trivially allows to perform an existential forgery: just choose Σ at random and compute $m = P(\Sigma)$; then the pair (m, Σ) is an apparently authentic signed message. The multiplicative property allows a selective forgery by performing a 2-chosen-message attack, i.e. a chosen-message attack requiring two messages. Let m be the message to be signed, choose x as you like in $[1, n\text{-}1]$ and compute

$y = m/x \mod n$ where n is the public modulus; obtain the signatures of x and y and compute the signature of m as the product $S(m) = S(x)S(y) \mod n$.

Different ways exist to eliminate these potential weaknesses. We can either add some redundancy to the message to be signed [ISO1], or use a hash-function in the signature scheme [ISO2], [BR]. The present paper is related to the redundancy solution. This solution is of particular interest when the message is short, because it prevents from specifying and implementing a hash-function (a rather delicate cryptographic challenge), and it allows to construct very compact signed messages, since messages can be recovered from the signatures themselves (and hence need not any longer be transmitted or stored). More precisely, let R be the (invertible) redundancy function. The signature of m is $\Sigma(m) = S[R(m)]$, and the signer only sends $\Sigma(m)$ to the receiver. The latter applies P to $\Sigma(m)$, and verifies that the result complies with the redundancy rule, i.e. is an element of the image set of R. Then he recovers m by discarding the redundancy (i.e. by applying R^{-1}) to this result.

But it has been shown in the past [DJC] that too simple redundancy does not avoid all the chosen-message attacks. For instance, the redundancy defined by appending trailing '0' bits to the message is insufficient because it remains possible, for any m, to construct two integers x and y such that $(m\|0..0) = (x\|0..0)(y\|0..0) \mod n$ (implying $S(m\|0..0) = S(x\|0..0)S(y\|0..0) \mod n$) by using Euclid's algorithm. In the standard ISO/IEC 9796 Part 1 [ISO1], a redundancy function is described, the security of which is assessed as very good. But its expansion rate (at least two) is too high in many applications, e.g. public key certification. As a consequence, there remains a need for a simple/short redundancy function providing adequate security.

The main goal of this paper is to show that a number of attractive redundancy functions, some of which proposed here and there, are subject to a 2-chosen-message attack. It is organized as follows: in section 2, we summarize our results, in section 3, we describe the mathematical tools used by our attacks, in section 4, attacks on valid messages with fixed redundancy, in section 5, attacks on valid messages with fixed and modular redundancy, in section 6, some applications including an attack on a scheme proposed for discussion inside ISO. We explain how to defeat this forgery in section 7 and we conclude in section 8.

Throughout this paper, we call valid message any message m completed with redundancy (i.e. any integer in the form $R(m)$), and bitlength (or length in short) of an integer the number of bits of its binary representation. We denote by $|m|$ the bitlength of m. We also define mb as the maximum bitlength of message accepted in a signature scheme.

2 Our Results

First, we extend the results of De Jonge and Chaum [DJC]: if the redundancy consists in appending any fixed bits to m to be signed, or more generally if redundancy takes the form of an affine function of m, that is when the signature $\Sigma(m)$ of m is computed as $\Sigma(m) = S(\omega m + a)$, for any constant a, any constant ω and message m, then the signature scheme is subject to a chosen-message attack, provided the redundancy is less than half the length of the public modulus used by S and P. De Jonge and Chaum

exhibited similar attacks only in the cases when $a = 0$ (with the same amount of redundancy) or when $\omega = 1$ (with a smaller amount of redundancy).

Next, we study the case of the redundancy obtained in appending to m the remainder of m modulo some fixed value. Then, the signature scheme is still subject to a 2-chosen-message attack, provided the redundancy is less than half the length of the public modulus minus the length of the remainder. In a particular case, it even works when the redundancy is up to half the length of the modulus.

Here, the term "chosen-message attack" means the following: for any arbitrary message m it is possible to construct two messages m_1 and m_2 such that $\Sigma(m_2) / \Sigma(m_1) = \Sigma(m)$ modulo the RSA-modulus used by S. Therefore, by obtaining the signatures of m_1 and m_2, an enemy can forge the signature of m. It must be stressed that m can be entirely selected by the enemy; so this forgery is selective, not only existential.

All the attacks make extensive use of an affine variant of Euclid's algorithm, due to Okamoto and Shiraishi [OS], which is described in the coming section.

3   Basic Tools

In all our attacks, we will face the following problem:

Let n be a positive integer and d, z_0, X, Y, with X and Y "small", four positive integers less than n. Find solutions x and y to:

$$(S) \qquad \begin{cases} dx = y + z_0 \pmod{n} \\ |x| < X \\ |y| < Y \end{cases}$$

3.1   Case of $z_0 = 0$

W. De Jonge and D. Chaum solved this problem [DJC]. There is at least one solution not equal to $(0, 0)$ if $XY > n$. Demonstration of this result uses the "pigeon-hole principle". It is useful to remark [GTV] that finding small x and y satisfying (S) comes to finding a good approximation of the fraction d/n. So, we find such a solution by developing it in continued fractions i.e. applying extended Euclidean algorithm to d and n.

Algorithm EE
- *Input*: n, d, X, Y (with $XY > n$)
- *Output*: nothing or some x such that $|x| < X$ and $|dx \pmod{n}| < Y$
- *Method*: apply extended Euclidean algorithm to d and n; one obtains coefficients l_i and m_i such that:

$$l_i n + m_i d = r_i \tag{1}$$

where the r_i are the successive remainders; output the smallest (in absolute value) m_i such that $n/Y < |m_{i-1}|$ (the case "such an m_i does not exist" is very rare).

• *Proof:* the fractions $|l_i/m_i| = -l_i/m_i$ are in fact the convergents of the development of d/n in continued fractions; hence:

$$|d/n + l_i/m_i| \leq 1/|m_i m_{i+1}| \quad \Rightarrow \quad |dm_i + nl_i| \quad \leq \quad n/|m_{i+1}|$$
$$\Rightarrow \quad |dm_i \,(\mathrm{mod}\, n)| \quad < \quad Y$$

Moreover, $(|m_i| \leq n/Y$ and $XY > n$) implies $|m_i| < X$

3.2 Case of $z_0 \neq 0$

Okamoto and Shiraishi provide in [OS] an extension of extended Euclidean algorithm which very often solves this problem. We use a version of this algorithm to generate solutions.

Algorithm OS

• *Input:* n, d, X, Y (with $XY > n$), z_0

• *Output:* nothing or some x such that $|x| < X$ and $|dx - z_0 \,(\mathrm{mod}\, n)| < Y$

• *Method:* apply extended Euclidean algorithm to d and n; introduce a sequence y_i whose first term y_0 is z_0 and following ones are defined by:

$$y_i = y_{i-1} - q_i' r_i \tag{2}$$

where q_i' is the quotient in the division of y_{i-1} by r_i; introduce also the sequence k_i whose first term k_0 is zero and the following ones are defined by:

$$k_i = k_{i-1} + q_i' m_i \tag{3}$$

Output k_i such that $n/Y < |k_i| < X$ and $|k_i| y_i \leq n$

• *Proof:* let the sequence h_i whose first term h_0 is zero and following ones defined by:

$$h_i = h_{i-1} + q_i' l_i \tag{4}$$

Then,

$$h_i n + k_i d = (h_{i-1} + q_i' l_i) n + (k_{i-1} + q_i' m_i) d$$
$$= h_{i-1} n + k_{i-1} d + q' (l_i n + m_i d)$$

(1) and (2) imply:

$$h_i n + k_i d = h_{i-1} n + k_{i-1} d + (y_{i-1} - y_i)$$

Then,

$$h_i n + k_i d = 0 + (y_0 - y_1) + (y_1 - y_2) + \ldots + (y_{i-1} - y_i)$$
$$= \qquad\qquad y_0 - y_i \qquad\qquad \Rightarrow k_i d \,(\mathrm{mod}\, n) = z_0 - y_i$$

By taking output's conditions on k_i into account, we have:

$$|k_i| < X \quad \text{and} \quad y_i \leq n/|k_i| < Y$$

Remark: to increase the number of solutions when $z_0 \neq 0$, you can combine one solution (x, y) found by algorithm OS with a solution (x', y') given by algorithm EE for the same system with $z_0 = 0$.

4 Valid Messages with Fixed Redundancy

Recall that redundancy function takes the form of an affine function of message. The signature $\Sigma(m)$ of m is computed as $\Sigma(m) = S(\omega m + a)$ for any constant a, any constant ω and message m. De Jonge and Chaum already studied multiplicative attacks on schemes using fixed redundancy. But their results were restricted to $a = 0$ (and any value of ω) or $\omega = 1$ (and any value of a). Moreover, their attack is valid if the redundancy takes up less than half of the bits in the modulus n when $a = 0$, and otherwise if the redundancy takes up less than one third of the bits in the modulus n.

Our method extend this results: the signature scheme is subject to a chosen-message attack for any value of a and ω, provided that the redundancy takes up less than half of the bits in a valid message.

In this section, we describe our attack on right-padded redundancy scheme, left-padded redundancy scheme, then on a more general scheme. Proof and efficiency are only given in the general case.

4.1 Right-Padded Redundancy Scheme

Let a be a fixed pattern of bits, and $\omega = 2^{|a|}$.
We denote by $\mathscr{E}$ the set of messages:
$$\mathscr{E} = \{\text{integers } m \text{ such that } 0 \leq m < n/\omega\}$$
and by $\mathscr{E}'$ the set of valid messages:
$$\mathscr{E}' = \{\omega m + a \text{ such that } m \in \mathscr{E}\}$$
Example: an element of $\mathscr{E}'$ has this form:

Message m	...1010010100001110101

Attack:
- Choose a message $m \in \mathscr{E}$ of which you want to forge a signature.
- Set

$$z_0 = \frac{a}{\omega}\left[1 - (\omega m + a)\right](\bmod n) \tag{5}$$

- Solve

$$(\omega m + a)x = y + z_0 (\bmod n) \tag{6}$$

with x and y elements of $\mathscr{E}$ by using algorithm OS. You obtain, very often, a solution if the range of m is larger than $\sqrt{n}$ (i.e. the number of bits of redundancy a is less than half of the bits of modulus n). See 4.3 for more details.
- By replacing z_0 by its expression (5) in the latter equation (6), you can easily prove that $(\omega m + a)(\omega x + a) = (\omega y + a) \pmod{n}$. If you get signatures of y and x (i.e. if you get $S(\omega y + a)$ and $S(\omega x + a)$), then you deduce the signature of m by dividing $S(\omega y + a)$ by $S(\omega x + a)$ modulo n.

4.2 Left-Padded Redundancy Scheme

Let a' be a fixed pattern of bits, and $\beta = 2^{mb}$. We denote by $\mathcal{E}$ the set of messages:
$$\mathcal{E} = \{\text{integers } m \text{ such that } 0 \le m < \beta\}$$
and by $\mathcal{E}'$ the set of valid messages:
$$\mathcal{E}' = \{m + a'\beta \text{ such that } m \in \mathcal{E}\}$$
Example: an element of $\mathcal{E}'$ has this form :

01001001010101111...	Message m

Attack:
- Choose a message $m \in \mathcal{E}$ of which you want to forge a signature.
- Set

$$z_0 = a'\beta[1 - (m + a'\beta)](\mathrm{mod}\ n) \qquad (7)$$

- Solve

$$(m + a'\beta)x = y + z_0(\mathrm{mod}\ n) \qquad (8)$$

with x and y elements of $\mathcal{E}$ by using algorithm OS. You obtain, very often, a solution if the range of m is larger than $\sqrt{n}$ (i.e. the number of bits of redundancy a is less than half of the bits of modulus n). See 4.3 for more details.
- By replacing z_0 by its expression (7) in the latter equation (8), you can easily prove that $(m + a'\beta)(x + a'\beta) = (y + a'\beta)$ (mod n). If you get signatures of y and x (i.e. if you get $S(y + a'\beta)$ and $S(x + a'\beta)$), then you deduce the signature of m by dividing $S(y + a'\beta)$ by $S(x + a'\beta)$ modulo n.

4.3 Generalization

Let a be the lower bound to a valid message, b be the upper bound to a valid message ($a \le m < b < n$), ω a multiplicative constant. Consequently, we can define $\mathcal{E}$ as the set of messages:
$$\mathcal{E} = \{\text{integers } m \text{ such that } 0 \le m < (b - a) / \omega\}$$
and $\mathcal{E}'$ as the set of valid messages:
$$\mathcal{E}' = \{\omega m + a \text{ such that } m \in \mathcal{E}\}$$

Attack:
- Choose a message $m \in \mathcal{E}$ of which you want to forge a signature.
- Set

$$z_0 = \frac{a}{\omega}[1 - (\omega m + a)](\mathrm{mod}\ n) \qquad (9)$$

- Solve

$$(\omega m + a)x = y + z_0(\mathrm{mod}\ n) \qquad (10)$$

with x and y elements of $\mathcal{E}$ by using algorithm OS. You obtain, very often, a solution if the range of m is larger than $\sqrt{n}$ (i.e. the number of bits of redundancy, multiplicative and additive, is less than half of the bits of modulus n).

- By replacing z_0 by its expression (9) in the latter equation (10), you can easily prove that $(\omega m + a)(\omega x + a) = (\omega y + a) \pmod{n}$. If you get signatures of y and x (i.e. if you get $S(\omega y + a)$ and $S(\omega x + a)$), then you deduce the signature of m by dividing $S(\omega y + a)$ by $S(\omega x + a)$ modulo n.

Proof: let x and y be a couple of solutions:

$$(\omega m + a)x = y + z_0 \pmod{n}$$

$$(\omega m + a)\omega x = \omega y + \omega\left[\frac{a}{\omega}[1 - (\omega m + a)]\right] \pmod{n}$$

$$(\omega m + a)\omega x = \omega y + a - a(\omega m + a) \pmod{n}$$

$$(\omega m + a)(\omega x + a) = (\omega y + a) \pmod{n}$$

Efficiency: algorithm OS gives a solution if $XY > n$ (see 3.2.), i.e. if:

$$\frac{(b-a)}{\omega}\frac{(b-a)}{\omega} > n$$

Thus, a solution is obtained when the range of m, i.e. $\frac{(b-a)}{\omega}$, is larger than $\sqrt{n}$ or when:

$$\log_2\left(\frac{b-a}{\omega}\right) > \log_2\left(\sqrt{n}\right)$$

$$\log_2(n) - \log_2\left(\frac{b-a}{\omega}\right) < \frac{1}{2}\log_2(n)$$

i.e. the number of bits of redundancy, multiplicative and additive redundancy, is less than half of the bits of modulus n.

Remarks :

- If ω is a power of two upper than $2^{|a|}$ then it is the right-padded redundancy scheme (see 3.1).
- If $\omega = 1$ and a is a multiple of 2^{mb} then it is the left-padded redundancy scheme (see 3.2).
- Note that with an appropriate choice of ω and a, it is a scheme with the message in the middle :

1101010101...	Message m	...00010101101
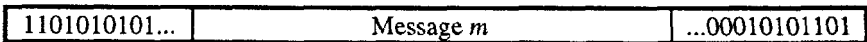
5   Valid Messages With Fixed And Modular Redundancy

The expression "modular redundancy" is used to indicate a redundancy obtained with a modular operation. We denote this modular redundancy by the function $H(x)$. In this section, we consider a modular redundancy of u bits in length.

We consider three cases: first of all, the particular case $H(m) = m \pmod{2^u + 1}$, a modular redundancy of u bits (except if $H(m) = 2^u$, an event of probability nearly equal to 0). Next $H(m) = m \pmod{2^u + v}$ where v is a negative integer greater than or

equal to -2^{u-1}, and last $H(m) = (m \pmod{2^u + v)}) \oplus Mask$ where v is a negative integer greater than or equal to -2^{u-1} and $Mask$ is a u-bit fixed string. We denote the message m concatenated with $H(m)$ by:

$$\Phi(m) = m \parallel H(m) \tag{11}$$

Let a and ω be integers less than n, α the length of message, and $\mathscr{E}$ the set of messages:

$$\mathscr{E} = \{m \text{ such that } 0 \leq m < 2^\alpha\}$$

Then, the set of valid messages is:

$$\mathscr{E}' = \{\omega\Phi(m) + a, \text{ with } m \in \mathscr{E}\}$$

Example: if ω is a power of two, then an element of $\mathscr{E}'$ has this form :

01011....	Message m (α bits)	$H(m)$ (u bits)	...0110

5.1 $H(m) = m \pmod{2^u + 1}$

We can also write

$$m = q(2^u + 1) + r \tag{12}$$

with q the quotient and r the remainder of Euclidean division of m by $(2^u + 1)$. Hence $\Phi(m) = [q(2^u + 1) + r] 2^u + r$ and finally we obtain:

$$\Phi(m) = \psi(m)(2^u + 1) \tag{13}$$

with

$$\psi(m) = q2^u + r \tag{14}$$

Consequently, a new definition of the set of valid message is possible :

$$\mathscr{E}' = \{\omega'\psi(m) + a \text{ with } m \in \mathscr{E}\}$$

with $\omega' = \omega(2^u + 1)$.
Our attack uses this new definition.

Attack:
• Choose a message m of which you want to forge a signature.
• Set

$$z_0 = \frac{a}{\omega'}\left[1 - (\omega'\psi(m) + a)\right] \pmod{n} \tag{15}$$

• Solve

$$(\omega'\psi(m) + a)x = y + z_0 \pmod{n} \tag{16}$$

with x and y positive integers less than $2^{\alpha+u}/(2^u + 1)$ by using algorithm OS. You obtain, very often, a solution if the number of bits of the message, α, is upper than half of the length of modulus n.
• By replacing z_0 by its expression (15) in the latter equation (16), you can easily prove that:

$$(\omega'\psi(m) + a)(\omega'x + a) = (\omega'y + a) \pmod{n}$$

But the definition of function ψ, (13), and the fact that $\Phi(m) < 2^{\alpha+u}$, imply the existence of a message m s.t. $\psi(m) = t$ when t is less than $2^{\alpha+u}/(2^u + 1)$. Consequently, there are two messages m_1 and m_2 such that $\psi(m_1) = x$ and $\psi(m_2) = y$. Finally, if you

get signatures of m_1 and m_2 (i.e. if you get $S(\omega'\psi(m_2) + a)$ and $S(\omega'\psi(m_1) + a)$), then you deduce the signature of m by dividing $S(\omega'\psi(m_2) + a)$ by $S(\omega'\psi(m_1) + a)$ modulo n.

5.2 $H(m) = m \pmod{2^u + v}$

Let
$$m = q(2^u + v) + r \tag{17}$$
where q and r are respectively the quotient and the remainder of the Euclidean division of m by $(2^u + v)$. Thus:
$$\Phi(m) = q(2^u + v)2^u + r(2^u + 1) \tag{18}$$
Given that $v \neq 1$, it follows that we cannot apply the latter method (5.1) to reduce the number of variables. Consequently, we will rather fix the value of either the quotient or the remainder. We choose to fix r because its range is shorter than the range of q. Hence, the modular redundancy is fixed as well.

Attack:
• Choose a message m of which you want to forge a signature.
• Choose r_1 and r_2 two positive integers less than $2^u + v$.
• Set
$$a_1 = r_1(2^u + 1)\omega + a$$
$$a_2 = r_2(2^u + 1)\omega + a \tag{19}$$
$$z_0 = \frac{1}{\omega 2^u(2^u + v)}\left[(\omega\Phi(m) + a)a_1 - a_2\right]$$

• Solve
$$(\omega\Phi(m) + a)q_1 = q_2 \cdot z_0 \pmod{n} \tag{20}$$
with q_1 and q_2 positive integers less than, respectively, $(2^\alpha - r_1)/(2^u + v)$ and $(2^\alpha - r_2)/(2^u + v)$, by using algorithm OS. You obtain, very often, a solution if the number of bits of the message, α, minus the number of bits of redundancy, u, is upper than half of the length of modulus n.
• Set
$$m_1 = q_1(2^u + v) + r_1 \tag{21}$$
and
$$m_2 = q_2(2^u + v) + r_2 \tag{22}$$
The set of possible values of q_1, r_1, q_2, r_2, implies that $m_1 \in M$ and $m_2 \in M$. By replacing z_0, a_1, a_2, by their expressions (19) in the solved equation (20), you obtain, after a brief calculation :
$$(\omega\Phi(m) + a)(\omega\Phi(m_1) + a) = (\omega\Phi(m_2) + a) \pmod{n}$$
Finally, you deduce the signature of m by dividing $S(\omega\Phi(m_2) + a)$ by $S(\omega\Phi(m_1) + a)$ modulo n.

5.3 $H(m) = (m \pmod{2^u + v}) \oplus Mask$

We denote by *Mask* a u-bit fixed string and by $\oplus$ the function exclusive OR.

We apply the same method as previously, but we introduce a new function:
$$C(r) = r2^u + (r \oplus Mask) \qquad (23)$$
Thus we obtain:
$$\Phi(m) = q(2^u + v)2^u + C(r) \qquad (24)$$
Since during the development of the attack the two remainders r_1 and r_2 are fixed, $C(r_1)$ and $C(r_2)$ are also fixed and the mask does not generate any extra difficulty.

Attack:
- Choose a message m of which you want to forge a signature.
- Choose r_1 and r_2 two positive integers such that they are less than $2^u + v$.
- Set

$$a_1 = C(r_1)\omega + a$$
$$a_2 = C(r_2)\omega + a \qquad (25)$$
$$z_0 = \frac{1}{\omega 2^u (2^u + v)}\left[(\omega\varphi(m) + a)a_1 - a_2\right]$$

- Solve

$$(\omega\Phi(m) + a)q_1 = q_2 - z_0 (\text{mod } n) \qquad (26)$$
with q_1 and q_2 positive integers less than, respectively, $(2^\alpha - r_1)/(2^u + v)$ and $(2^\alpha - r_2)/(2^u + v)$, by using algorithm OS. You obtain, very often, a solution if the number of bits of the message, α, minus the number of bits of redundancy, u, is upper than half of the length of modulus n.
- Set

$$m_1 = q_1(2^u + v) + r_1 \qquad (27)$$
and
$$m_2 = q_2(2^u + v) + r_2 \qquad (28)$$
The set of possible values of q_1, r_1, q_2, r_2, implies that $m_1 \in M$ and $m_2 \in M$. By replacing z_0, a_1, a_2, by their expressions (25) in the solved equation (26), you obtain, after a brief calculation :
$$(\omega\Phi(m) + a)(\omega\Phi(m_1) + a) = (\omega\Phi(m_2) + a) (\text{mod } n)$$
Finally, you deduce the signature of m by dividing $S(\omega\Phi(m_2) + a)$ by $S(\omega\Phi(m_1) + a)$ modulo n.

Remark: since this attack does not depend on the exact expression of $C(r)$, it can be performed against any modular redundancy in the form:
$$H(m) = H'[m (\text{mod } 2^u + v)], \text{ for any function } H'.$$

6 Applications

We applied our results to a part of the project on digital signature schemes giving message recovery ISO/IEC JTC 1/SC 27 [ISO]. It was a Working Draft (WD), i.e. one of the first stages of the development of International Standards. After, when the working group is satisfied with the specified solution, the next step is the Committee Draft (CD), which is submitted to a ballot. Successive Committee Drafts may be

considered until consensus is reached on the technical content. Once consensus has been attained, the text is finalized for submission as a Draft International Standard (DIS). Once a DIS has been approved, the final text is published as an International Standard (IS).

Part 2 of this project aims at defining a signature scheme allowing short certificates, which is convenient for smart cards. Like ISO/IEC 9796 [ISO], it is supposed to avoid the known attacks against RSA [GQLS]. In a particular case, this project uses a simplified hash-function $H(m) = 2(m \pmod{2^{79}+1})$ to define the modular redundancy. Structure of a valid message :

Adaptation bits	More-data bit	Padding Field	Data Field	Check Field	Adaptation nibble
Fixed: 2 bits	Fixed: 1 bit	Variable: 1 or more bits	Variable	Fixed: 80 bits	Fixed: 4 bits
01	0	0, 1 or more bits set to 0 followed by 1 bit set to 1	Message	Modular redundancy	0110

We implemented algorithms OS and EE in C-language on a PC computer to obtain our results. With a message m of 384 bits, $H(m) = 2(m \pmod{2^{79}+1})$, and a 512-bit RSA-modulus to define this scheme, we found nearly 40 solutions with algorithm OS and nearly 4000 solutions by the means of a simple combination with results of algorithm EE. This result can certainly be improved if all possible combinations are considered. When the length of message is 425-bit long, we found 60 or so with OS and about 8800 with OS combined with EE.

We have modified the function $H(m)$ to study the efficiency of our algorithm. With $H(m) = Mask \oplus 2(m \pmod{2^{79}+1})$ and $Mask =$ BBBBBBBBBBBBBBBBBBBB, we found, when the length of message is 384 bits nearly 16 solutions with OS and nearly 670 with OS and EE. When the length of message is 425 bits, we found 23 or so with OS and about 1720 with OS combined with EE. As previously, the number of solutions can certainly be expanded.

Remark: in the first case, we obtain more solutions than in the second one because the redundancy is not fixed. In fact, using $H(m) = 2(m \pmod{2^{79}+1})$ is like using the particular modular redundancy defined in 5.1. Here $u = 80$ and

$$\Phi(m) = [q(2^{79} + 1) + r] \, 2^{80} + 2r$$

with q the quotient and r the remainder of Euclidean division of m by $(2^{79} + 1)$. Finally we obtain:

$$\Phi(m) = \psi(m)(2^{80} + 2) \text{ with } \psi(m) = q2^{79} + r$$

and the attack described in 5.1 can be applied.

7 How To Defeat This Forgery

At Eurocrypt'96 Rump Session, we proposed three solutions to repair the previous schemes :

- Introduce the quotient q of Euclidean division of m by $(2^u + v)$

$$H(m) = r \times q \pmod{2^u + v}$$

506

This definition of *H* implies that we cannot isolate *q* and *r* in the expression of *m* concatenated with *H(m)*. The principle of our attack cannot be used here.
- Append to *m* its remainders modulo two different values, $2^{w/2} + v$ and $2^{w/2} + w$ with $v \neq w$. Two different moduli increase the link between message and redundancy, there is an interdependence between the different quotients and remainders. One of them cannot be fixed to use our attack. Simple values can be chosen, e.g. $v = -1$ and $w = 0$.
- Split the message into different parts and keep a simple redundancy. This method increases the number of variables and OS cannot be used to solve $mx = y \pmod{n}$. The latter solution is used in ISO/IEC 9796-3 [ISO3], Working Draft, December 1996, which replaces ISO/IEC JTC 1/SC 27 [ISO].
Remark: one of the authors has recently discovered a multiplicative attack using lattice basis reduction and only the first solution is valid.

8   Conclusion

We have shown the weakness of many attractive redundancy functions for the purpose of RSA digital signatures. We successfully applied our attack to an ISO Working Draft [ISO] and a modified version using a redundancy function with mask. Thus, we showed that some redundancy function may be inappropriate, even when it is message-dependent and even when it involves non-arithmetic operations. Afterwards, we have proposed new redundancy functions, which apparently cannot be attacked by our techniques. Nevertheless a further research showed that two of them can be attacked by a LLL-based method.

Acknowledgments

We would like to thank Louis Guillou for many fruitful discussions about RSA signature schemes and for stimulating this research. We are grateful to Luc Vallée for help on the C-language and for lending of his big number library. We also thank the referees for their useful comments on the previous version of the paper, which helped improve the quality of this paper.

References

[BR]   M. Bellare, P. Rogaway, "The Exact Security of Digital Signatures - How to Sign with RSA and Rabin", Eurocrypt'96 Proceedings, Lecture Notes In Computer Science, Vol.1070, U. Maurer ed., Springer-Verlag, 1996.
[DJC]   W. De Jonge, D. Chaum, "Attacks on some RSA Signatures", Advances in Cryptology, Crypto'85 Proceedings, Lecture Notes In Computer Science, Vol.218, Springer-Verlag, Berlin, 1986, pp. 18-27.
[GQLS]   L.C. Guillou, J.J. Quisquater, P. Landrock, C. Shaer, "Precautions taken against various potential attacks in ISO/IEC DIS 9796, Digital signature scheme giving message recovery", Eurocrypt'90 Proceedings, Lecture Notes in Computer Science, Vol.473, Springer-Verlag, pp 465-473.

[GTV]   M. Girault, P. Toffin, B. Vallée, "Computation of approximation L-th roots modulo n and application to cryptography", Proc. of Crypto'88, LNCS 403, Springer-Verlag, 1988, pp.100-117.

[ISO]   ISO/IEC JTC 1/SC 27, "Digital signature schemes giving message recovery; Part 2: Mechanisms using a hash function", Working Draft, January 1996.

[ISO1]   ISO/IEC 9796-1, "Digital signature schemes giving message recovery; Part 1: Mechanisms using redundancy".

[ISO2]   ISO/IEC 9796-2, "Digital signature schemes giving message recovery; Part 2: Mechanisms using a hash-function".

[ISO3]   ISO/IEC 9796-3, "Digital signature schemes giving message recovery; Part 3: Mechanisms using a check-function".

[OS]   T. Okamoto and A. Shiraishi, "A fast signature scheme based on quadratic inequalities", Proc. of the 1985 Symposium on Security and Privacy, Apr.1985, Oakland, CA.

[RSA]   R.L. Rivest, A. Shamir and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems", CACM, Vol. 21, n°2, Feb. 1978, pp. 120-126.