More Detail for a Combined Timing and Power Attack against Implementations of RSA

Werner Schindler[1] and Colin D. Walter[2]

[1] Bundesamt für Sicherheit in der Informationstechnik (BSI)
Godesberger Allee 185-189, 53175 Bonn, Germany
Werner.Schindler@bsi.bund.de
[2] Comodo Research Laboratory
10 Hey Street, Bradford, BD7 1DQ, UK
Colin.Walter@comodogroup.com

Abstract. Implementations of Montgomery's modular multiplication algorithm (MMM) typically make conditional subtractions in order to keep the output within register or modulus bounds. For some standard exponentiation algorithms such as m-ary, it has been shown that this yields enough information to deduce the value of the exponent. This has serious implications for revealing the secret key in cryptographic applications without adequate counter-measures. Much more detail is provided here about the distribution of output values from MMM when the output is only reduced to keep it within register bounds, about how implementations of sliding windows can be attacked, and about handling errors.

Key Words. RSA cryptosystem, side channel leakage, power analysis, timing attack, Montgomery modular multiplication, exponentiation, statistical decision problem.

1 Introduction

Side-channel leakage occurs through data dependent variation in the use of resources such as time and hardware. The former results from branching in the code or compiler optimisation [4], and the latter from data causing gate switching in the circuitry. Both manifest themselves measurably through overall current variation as well as local or global electro-magnetic radiation (EMR) [5–7]. Information leaking in these ways might be used by an attacker to deduce secret cryptographic keys which are contained securely inside a smart card.

Timing variation can be measured using overall delay during decryption or signing [1, 4, 11, 13]. However, for successful attacks on keys with limited lifespans for which only a bounded number of decryptions with the same key is allowed, some finer detail may become necessary, such as the time variations for executing component procedures. This detail can be seen by observing delays between the characteristic power consumption wave form for loading instructions and data at the start of the procedure. Here this is applied to the version of Montgomery modular multiplication [9] which contains a final conditional subtraction for reducing the output by only enough to fit within register bounds. The aim is to recover a secret RSA [10] exponent from deductions of whether or not this conditional subtraction occurs.

An essential assumption is that there is no blinding of the exponent or randomisation in the exponentiation algorithm so that the same type of multiplicative operations are performed (but with different data) every time the key is used. This enables an attacker to capture the collection of instances of the extra final subtraction for each individual operation. We will show how to determine exponent digits from this data, recover from any errors, and hence obtain the secret key even if the input text (the base of the exponentiation) has been blinded.

The results emphasise the need for care in the implementation of RSA. Indeed, similar results apply to elliptic curve cryptography (ECC) [8] and other crypto-systems based on modular arithmetic. In all cases where a key is re-used, some counter-measures should be employed to ensure that the conditional subtraction is removed, or at least hidden, and that the same sequence of key-dependent operations is not re-used for each exponentiation. In elliptic curve cryptography, standard key blinding adds about 20% to the cost of a point multiplication. The temptation to avoid this overhead should be resisted.

Montgomery modular multiplication [9] is arguably the preferred method for hardware implementation. We treat this algorithm here, but equivalent results should hold for any method for which there are correlations between the data and the multiplication time. The generic timing attack methodology which we adopt was first described in [16]. This approach established the feasibility of recovering the key within its lifetime and also gave the theoretical justification behind the different frequencies of final subtractions observed between squarings and multiplications.

A closer study of such distributions reduces the number of exponentiations required to recover the key. This was done for various settings in [11–13] when the final reduction yields output less than the modulus. However, reducing output only as far as the same length as the modulus is more efficient, and is the case studied here. The mathematics is more involved and so requires numerical approximation methods, but this is easy for an attacker. In addition, more detail is given concerning the recovery of the key when sliding windows exponentiation is employed.

2 The Computational Model

In order to clarify the precise conditions of the attack, each assumption is numbered. First, the computational model requires several assumptions, namely:

i) secret keys are unblinded in each exponentiation;
ii) Montgomery Modular Multiplication (MMM) is used with a conditional subtraction which reduces the output to the word length of the modulus;
iii) the most significant bit of the modulus lies on a word boundary; and
iv) exponentiation is performed using the m-ary sliding windows method [2, 3].

Exponent blinding might be deemed an unnecessary expense if other countermeasures are in place. So we still need better knowledge of how much data leaks out under the first assumption. For a sample of one exponentiation this particular assumption is irrelevant, but then, with data leaked from other sources, the techniques here might be just sufficient to make a search for the key computationally feasible.

As shown below, efficiency is good justification for the second hypothesis. The word length referred to there is the size of inputs to the hardware multiplier, typically 8-, 16- or 32-bit. However, standards tend to specify key lengths which are multiples of a large power of 2, such as 768, 1024, 1536 and 2048 bits. So the third hypothesis is almost certainly the case.

Because of its small memory requirements and greater safety if an attacker can distinguish squares from multiplications, a sliding window version of 4-ary exponentiation is the usual algorithm employed in smartcards. The exponent is re-coded from least to most significant bit. When a bit 0 occurs, it is re-coded as digit 0 with base 2 and when a bit 1 occurs, this bit and the next one are recoded as digit 1 or 3 with base 4 in the obvious way. This representation is then processed left to right, with a single modular squaring of the accumulating result for digit 0, and two modular squarings plus a modular multiplication for digits 1 and 3. According to the digit, the first or pre-computed third power of the initial input is used as the other operand in the multiplications.

An attacker can recover the secret exponent if he can i) distinguish squarings from multiplications, and ii) determine whether the first or third power was used as an operand in the multiplications. The classical square-and-multiply or binary algorithm ($m = 2$) is less secure against this type of attack as the second task is omitted. The attack here treats any m, and applies equally well when the Chinese Remainder Theorem (CRT) is used.

3 Initial Notation

Let R be the so-called “Montgomery factor” associated with an n-bit RSA modulus M. Then the implementation of Montgomery's algorithm satisfies the following specification [15]:

v) For inputs $0 \le A,\ B < R$, MMM generates output $P \equiv A{*}B{*}R^{-1} \bmod M$ satisfying $ABR^{-1} \le P < M{+}ABR^{-1}$ before any final, conditional subtraction.

Clearly this specifies P uniquely. The division by R is the result of shifting the accumulating product down by the multiplier word length for a number of iterations equal to the number of words in A. Since A, B and M will all have the same word length for the applications here, hypothesis (iii) yields

vi) $M < R < 2M$

i.e. R is the smallest power of 2 which exceeds M.

The output bound $P < M{+}R$ implied by (v) means that any overflow into the next word above the most significant of M has value at most 1. When this overflow bit is 1, the conditional subtraction can be called to reduce the output to less than R without fear of a negative result. Hence that bit can be used efficiently to produce an output which satisfies the pre-conditions of (v) for further applications of MMM. This is the case of MMM that is considered here.

Alternatively, the condition $P < M$ is often used to control the final subtraction. This is the standard version, but the test requires evaluating the sign of $P-M$, which is more expensive than just using the overflow bit. This case of MMM was discussed in [12]. A constant time version of MMM is possible by computing and storing both P and $P-M$, and then selecting one or other according to the sign of the latter. However, the subtraction can be avoided entirely if the inputs satisfy $A, B < 2M$ and R satisfies $4M < R$. This, too, is more expensive [15].

As usual, the private and public exponents are denoted d and e. For deciphering (or signing), ciphertext C is converted to plaintext $C^d \bmod M$. The m-ary sliding windows exponentiation algorithm using MMM requires pre-computation of a table containing $C^{(i)} \equiv C^iR \bmod M$ for each odd i with $1 \leq i < m$. This is done using MMM to form $C^{(1)}$ from C and $R^2 \bmod M$, $C^{(2)}$ from $C^{(1)}$, and then iteratively $C^{(i+2)}$ from $C^{(i)}$ and $C^{(2)}$. By our assumptions, the word length of each $C^{(i)}$ is the same as that of M, but it may not be the least non-negative residue. We also define $b = \log_2 m$ and assume it to be integral.

4 The Threat Model

The security threat model is simply that:

vii) an attacker can observe and record every occurrence of the MMM conditional subtraction over a number of exponentiations with the same key.

Section 1 provided the justification for this. The attack still works, although less efficiently, if manufacturers' counter-measures reduce certainty about the occurrence or not of the subtraction.

Unlike many attacks in the past, we make no assumptions about the attacker having control or knowledge of input to, or output from, the exponentiation. Although he may have access to the ciphertext input or plaintext output of a decryption, random nonces and masking should be employed to obscure such knowledge so that he is unable to use occurrences of the conditional subtraction to determine whether a square or multiply occurred. It is therefore assumed that

viii) the attacker can neither choose the input nor read either input or output.

Indeed, even without masking this is bound to be the case for exponentiations when the Chinese Remainder Theorem has been used. Lastly, as the attacker may have legitimate access to the public key $\{M, e\}$, it is assumed that

ix) the correctness of a proposed value for the private exponent d can be checked.

The assumptions so far mean that the same sequence of multiplicative operations is carried out for every decryption. So, from a sample of N decryptions with the same key, the attacker can construct an array $Q = (q_{i,j})$ whose elements are 1 or 0 depending on whether or not the ith MMM of the jth decryption includes the conditional subtraction. The elements $q_{i,j}$ are called *extra reduction* or *er*-values. Similarly, initialisation gives a matrix $Q' = (q'_{i,j})$: if C_j is the input to the jth decryption then the er-value $q'_{i,j}$ is associated with the calculation of $C_j^{(i)} \equiv {C_j}^i R \bmod M$ for the digit i.

5 Some Limiting Distributions

The timing attack here was first reported in [16], but precise MMM output bounds now allow a much more accurate treatment of the probabilities and hence more reliable results. More exact figures should make it easier to determine the precise strength of an implementation against the attack. Important first aims are to determine the probability of extra reductions and to establish the probability distribution function for the MMM outputs P. To this end two reasonable assumptions are made about such distributions:

x) The ciphertext inputs C behave as realizations of independent random variables which are uniformly distributed over $[0, M)$.

xi) For inputs A and B to MMM during an exponentiation, the output prior to the conditional subtraction is uniformly distributed over the interval $[ABR^{-1}, M{+}ABR^{-1})$.

Assumption (x) is fulfilled if, for example, the ciphertext is randomly chosen (typical for RSA key exchange) or message blinding has been performed as proposed in [4]. Here it is also a convenient simplification. In fact, there may be some initial non-uniformity (with respect to the Euclidean metric) which might arise as a result of content or formatting, such as with constant prefix padding. We return to this topic in Remark 1.

In (xi), the multiples of M subtracted from the partial product P during the execution of MMM are certainly influenced by *all* bits of A, B and M. However, the probability for the final conditional subtraction is essentially determined by the *topmost* bits of the arguments.

Before the formal treatment we illustrate the limit behaviour of the distributions which will be considered and provide the informal reasoning behind their construction. In order to exhibit the difference between squares and multiplications software was written to graph the probability of the extra subtraction in the limit of three cases, namely the distribution of outputs after a long sequence of i) squarings of a random input, ii) multiplications of independent random inputs, and iii) multiplications of a random input by the same fixed constant A. The result is illustrated in Figure 1. In the second case, two independent values are chosen from the kth distribution in the sequence. They are used as the inputs to MMM and the output generates the $k{+}1$st distribution. In all three cases the convergence is very rapid, with little perceptible change after 10 or so iterations.

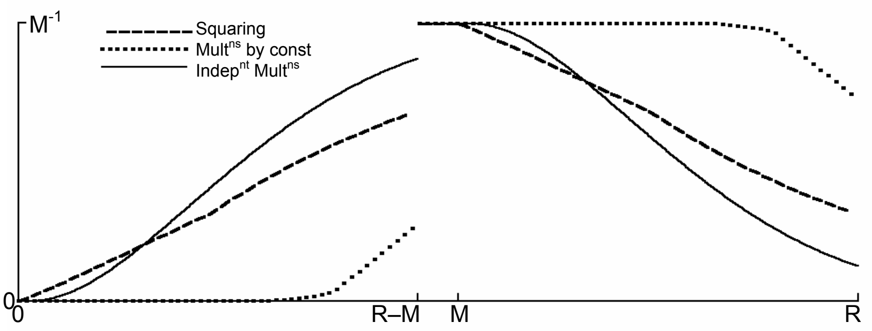

Fig. 1. Probability (vertical axis) of output in the range $0..R{-}1$ (horizontal axis) after a number of squarings, independent multiplications or multiplications by a constant A for the case $M = 0.525R$ and $A = 1.5M$.

The density functions have three distinct sections which correspond to the output belonging to one of the intervals $[0, R{-}M)$, $[R{-}M, M)$ or $[M, R)$. This is because outputs with a residue in $[R{-}M, M)$ have only one representative in $[0, R)$ and so, under hypothesis (xi), they occur in $[R{-}M, M)$ with probability M^{-1} whereas outputs with a residue in $[0, R{-}M)$ have two representatives in $[0, R)$ with combined probability M^{-1}. Note the discontinuities in the functions at $R{-}M$ and that probabilities tend to 0 as the output approaches 0.

Because M is large, the discrete probabilities are approximated accurately by the density functions of continuous random variables. If f is the limiting density function of a sequence of independent multiplications, then

1) $\int_0^R f(x)dx = 1$
2) $f(x) = 0$ for $x < 0$ and $R < x$
3) $f(x) = M^{-1}$ for $R{-}M \le x \le M$
4) $f(x){+}f(x{+}M) = M^{-1}$ for $0 \le x \le R{-}M$
5) $f(x) = \frac{1}{M}\int_0^x f(y)dy + \frac{1}{M}\int_x^R \int_0^{Rx/y} f(y)f(z)dz\,dy$ for $0 \le x \le R{-}M$

Properties 1 to 4 are already clear, and apparent in the figure. Property 5 encapsulates the restrictions imposed by MMM. To establish it, consider outputs x in the interval $[0, R{-}M)$. These do not involve a final conditional subtraction, and so, by (xi), they are derived from MMM inputs y and z for which $[yzR^{-1}, M{+}yzR^{-1})$ contains x. As the distribution is assumed to be uniform on this interval, there is a 1 in M chance of obtaining x if $yzR^{-1} \le x < M{+}yzR^{-1}$, i.e. if $R(x{-}M)/y < z \le Rx/y$. Since $R(x{-}M)/y \le R(R{-}2M)/y < 0$, the lower bound on feasible z is 0 and the upper bound is $\min\{R, Rx/y\}$. Thus,

$$f(x) = \frac{1}{M}\int_0^R \int_0^{\min\{R,Rx/y\}} f(y)f(z)dz\,dy.$$

The integral over y can be split into the sub-intervals $[0, x]$ and $[x, R]$ in order to separate the cases of the upper limit on z being R or Rx/y. This yields

$$f(x) = \frac{1}{M}\int_0^x \int_0^R f(y)f(z)dz\,dy + \frac{1}{M}\int_x^R \int_0^{Rx/y} f(y)f(z)dz\,dy$$

in which the first integral simplifies to give the expression in property 5.

These properties determine f completely. Although an algebraic solution does not exist, numerical approximations are easy to obtain. The other two density functions satisfy the same properties 1 to 4. The analogues to property 5 are:

5′) $f(x) = \frac{1}{M}\int_0^{\sqrt{Rx}} f(y)dy$ for $0 \le x \le R{-}M$

for the limit of consecutive squarings, and

5″) a) $f(x) = \frac{1}{M}\int_0^{Rx/A} f(y)dy$ for $0 \le x \le A \le R{-}M$

b) $f(x) = \frac{1}{M}$ for $A \le x \le R{-}M$

for the limit of consecutive multiplications by a constant A.

Although there are differences between the distributions for squaring and multiplication, they are substantially the same. Figure 1 illustrates the largest possible differences, which occur for M close to $\frac{1}{2}R$. For M close to R they are essentially all equal to the same uniform distribution. The distributions which lead to these limiting cases are considered in more detail in the next section.

6 Conditional Probabilities

As before, a prime $'$ is used on quantities relating to the initialisation phase, and unprimed identifiers relate to the computation stage. The attacker wants to estimate the types of the Montgomery multiplications in the computation phase on basis of the observed extra reduction (*er*-) values $q'_{j,k}$ and $q_{j,k}$ within the initialization and computation phases respectively. These types will be denoted using the text characters of the set $\mathcal{T} := \{\text{`}S\text{'}, \text{`}M_1\text{'}, \ldots, \text{`}M_{m-1}\text{'}\}$ where 'S' corresponds to a square, and 'M_i' to multiplication by the precomputed table entry $C^{(i)}$ associated with digit i of the re-coded exponent. In this section we derive explicit formulas for the probabilities of these events given the observed er-values. We begin with some definitions. It is convenient to normalise the residues mod M to the unit interval through scaling by a factor M.

Definition 1. *A realization of a random variable X is a value assumed by X. For sample number k, the er-values $q'_{j,k}$ and $q_{j,k}$ are defined by $q'_{j,k} := 1$ if computation of table entry ${C_k}^{(j)}$ in the initialisation phase requires an extra reduction (this includes both $q'_{1,k}$ and $q'_{2,k}$), and similarly $q_{j,k} := 1$ if the* jth *Montgomery multiplication in the computation phase requires an extra reduction. Otherwise $q'_{j,k} := 0$ and $q_{j,k} := 0$. As abbreviations, let $\boldsymbol{q}'_k := (q'_{1,k}, \ldots, q'_{m-1,k})$ and $\boldsymbol{q}_{i,\ldots,i+f-1;k} := (q_{i,k}, \ldots, q_{i+f-1,k})$. For $A \subseteq B$ the indicator or characteristic function $1_A\colon B \to \mathbb{R}$ is defined by $1_A(x) := 1$ if $x \in A$ and 0 otherwise. Further, for $\gamma := M/R$, let $\chi\colon [0, 1+\gamma^{-1}) \to [0, \gamma^{-1})$ be given by $\chi(x) := x$ if $x < \gamma^{-1}$ and $\chi(x) := x - 1$ otherwise; that is, $\chi(x) := x - 1_{x \ge \gamma^{-1}}$. Lebesgue measure is denoted by λ.*

Lemma 1. (i) $\frac{\mathrm{MMM}(A,B)}{M} = \chi\left(\frac{A}{M}\frac{B}{M}\frac{M}{R} + \frac{ABM^*(\mathrm{mod}\ R)}{R}\right)$
where $M^* = (-M)^{-1}(\mathrm{mod}\ R)$.

(ii) The extra reduction in MMM is necessary exactly when -1 is subtracted by the application of χ.

This lemma follows immediately from the definition of Montgomery's multiplication algorithm. Assertion (xi) is equivalent to saying that the second summand in the right-hand side of Lemma 1(i) is uniformly distributed on the unit interval $[0,1)$. For a formal justification, the ideas from the proof of Lemma A.3(iii) in [13] can be adjusted in a straightforward way. To formulate a mathematical model we need further definitions which mirror the operations which comprise the exponentiation:

Definition 2. *Assume $T(i) \in \mathcal{T}$, $i = 1, 2, \ldots$, describes the sequence of multiplicative operations in the exponentiation. Let $F := \{i \mid 1 \le i \le m{-}1,\ i \text{ odd}\}$. Suppose the random variables V'_i $(i \in F \cup \{2\})$ and $V_1, V_2, \ldots$ are independent and equidistributed on the unit interval $[0,1)$. Define the random variables S'_i $(i \in F \cup \{0,2\})$ so that S'_0 assumes values in $[0,1)$ and*

$$
\begin{aligned}
S'_1 \quad &:= \quad \chi\left(S'_0(R^2 \,(\bmod M)/M)\gamma + V'_1\right) && (1)\\
S'_2 \quad &:= \quad \chi\left({S'_1}^2\gamma + V'_2\right) && (2)\\
S'_{2i-1} \quad &:= \quad \chi\left(S'_{2i-3}S'_2\gamma + V'_{2i-1}\right) \qquad \textit{for } 1 < i \le \frac{m}{2} && (3)
\end{aligned}
$$

Similarly, define $S_0 := S'_r$ where $r \in F$ is the left-most digit of the secret exponent d after recoding (cf. Sect. 2) and, for $i \ge 1$, let

$$
S_i \;:=\; \begin{cases} \chi\left(S_{i-1}^2\gamma + V_i\right) & \textit{if } T(i) = \text{‘}S\text{’} \\ \chi\left(S_{i-1}S'_j\gamma + V_i\right) & \textit{if } T(i) = \text{‘}M_j\text{’} \end{cases} \tag{4}
$$

Lastly, define $\{0,1\}$-valued random variables $W'_1, \ldots, W'_{m-1}$ and $W_1, W_2, \ldots$ by

$$
\begin{aligned}
W'_1 \quad &:= \quad 1_{S'_1 < S'_0(R^2 \,(\bmod M)/M)\gamma} && (5)\\
W'_2 \quad &:= \quad 1_{S'_2 < {S'_1}^2\gamma} && (6)\\
W'_{2i-1} \quad &:= \quad 1_{S'_{2i-1} < S'_{2i-3}S'_2\gamma} \qquad \textit{for } 1 < i \le \frac{m}{2} && (7)
\end{aligned}
$$

$$
W_i \;:=\; \begin{cases} 1_{S_i < S_{i-1}^2\gamma} & \textit{if } T(i) = \text{‘}S\text{’} \\ 1_{S_i < S_{i-1}S'_j\gamma} & \textit{if } T(i) = \text{‘}M_j\text{’} \end{cases} \tag{8}
$$

Thus the distribution of S_i describes the random behaviour of the output from the ith multiplication, V'_i and V_i correspond to the variation described in assumption (xi), and W'_i, W_i are the associated distributions of final subtractions recorded in Q' and Q:

Mathematical Model. We interpret the components of the er-vector $\boldsymbol{q}'_k = (q'_{1,k}, \ldots, q'_{m-1,k})$, row subscripts in $F \cup \{2\}$, as realizations of the random variables $W'_1, \ldots, W'_{m-1}$ with S'_0 having the uniform distribution of the normed (random) input C_k/M to the kth exponentiation. Similarly, we interpret $q_{1,k}, q_{2,k}, \ldots$ as realizations of the random variables $W_1, W_2, \ldots$.

Consequently, we have to study the stochastic processes $W_1', W_2', W_3', W_5', \ldots, W_{m-1}'$ and $W_1, W_2, \ldots$. However, the situation is much more complicated than in the case of the standard Montgomery algorithm considered in [12] because the random variables $S_1, S_2, \ldots$ are neither independent nor identically distributed. Their distribution depends on the sequence of operations $T(1), T(2), \ldots$.

Definition 3. *(i) For $i \in F{\cup}\{2\}$, $w \in \{0,1\}$ and indices of the components ranging over $F{\cup}\{0,2\}$, let the subset $\mathcal{D}'(i;w) \subseteq [0,1){\times}[0,\gamma^{-1})^{\frac{1}{2}m+1}$ be given by those vectors $(s_0', \ldots, s_{m-1}')$ for which $v_1', \ldots, v_{m-1}' \in [0,1)$ exist such that $s_0', \ldots, s_{m-1}', v_1', \ldots, v_{m-1}'$ satisfy* (1)-(3) *in place of $S_0', \ldots, S_{m-1}', V_1', \ldots, V_{m-1}'$ and, additionally, the component s_i' and its predecessor must result in w when inserted into whichever of* (5), (6) *or* (7) *describes W_i'. Thus, for example, by* (6) *the vectors $(s_0', \ldots, s_{m-1}')$ in $\mathcal{D}'(1;1)$ satisfy $s_1' < s_0'(R^2(\mathrm{mod}\, M))/R$, and by* (7) *those in $\mathcal{D}'(2i{-}1;1)$ satisfy $s_{2i-1}' < s_{2i-3}'s_2'\gamma$.*
(ii) For $i \leq j \leq i{+}f{-}1$, $w \in \{0,1\}$ and $t \in \mathcal{T}$, define $\mathcal{D}_f(i,j;w,t) \subseteq [0,\gamma^{-1})^{f+1}$ to be the subset of vectors $(s_{i-1}, \ldots, s_{i+f-1})$ for which there are $v_i, \ldots, v_{i+f-1} \in [0,1)$ such that $s_{i-1}, \ldots, s_{i+f-1}, v_i, \ldots, v_{i+f-1}$ satisfy (4) *in place of $S_{i-1}, \ldots, S_{i+f-1}, V_i, \ldots, V_{i+f-1}$ with the assumption that $T(j) = t$ and, additionally, the component s_j and its predecessor must result in w when inserted into the instance of* (8) *which describes W_j for $T(j) = t$. Thus, for example, the elements of $\mathcal{D}_f(i,j;1,\text{‘}M_k\text{’})$ satisfy the constraint $s_j < s_{j-1}s_k'\gamma$.*

Remark 1. As already mentioned in Sect. 5, there may also be scenarios of practical interest which imply distributions of S_0' other than the uniform distribution of assertion (x). For signing with constant prefixed padding, for instance, the random variable S_0' can be assumed to have a Dirac (= single-point) distribution due to the definition of S_1'. Generally speaking, the distribution of S_0' influences the conditional density $g(\cdot|\cdot)$ in Lemma 2(i) and hence implicitly the conditional probabilities in Theorem 1 and the optimal decision strategy. Then it is necessary to adjust the derivation of $g(\cdot|\cdot)$ to the concrete distribution of S_0' but otherwise the remaining steps in this and the forthcoming sections pass through identically.

The (conditional) probability of a given *er*-vector can be described using the sets of Definition 3:

Lemma 2. *(i) Assume the random variable S_0' is equidistributed on $[0,1)$. Then the conditional distribution of the random vector $(S_1', \ldots, S_{m-1}')$ on $[0,1)^{\frac{1}{2}m+1}$ under the condition $W_1'{=}w_1'$, …, $W_{m-1}'{=}w_{m-1}'$ has Lebesgue probability density*

$$g(s_1', \ldots, s_{m-1}' \mid w_1', \ldots, w_{m-1}') :=$$

$$\frac{\int_0^1 1_{\cap_{i\in F\cup\{2\}}\mathcal{D}'(i;w_i')}(s_0', s_1', \ldots, s_{m-1}')\, ds_0'}{\int_{[0,1)\times[0,\gamma^{-1})^{\frac{1}{2}m+1}} 1_{\cap_{i\in F\cup\{2\}}\mathcal{D}'(i;w_i')}(s_0', s_1', \ldots, s_{m-1}')\, ds_0' ds_1' \cdots ds_{m-1}'}. \qquad (9)$$

(ii) The distribution of S_{i-1} has a Lebesgue density, say h_{i-1}. Moreover,

$$\mathrm{Prob}(W_i = w_i, \ldots, W_{i+f-1} = w_{i+f-1} \mid W'_1 = w'_1, \ldots, W'_{m-1} = w'_{m-1}) \ = \tag{10}$$
$$\int\limits_{[0,\gamma^{-1})^{\frac{1}{2}m+f+2}} g(s'_1, \ldots, s'_{m-1} \mid w'_1, \ldots, w'_{m-1}) \cdot h_{i-1}(s_{i-1}) \times$$
$$\times\ 1_{\bigcap_{u=i}^{i+f-1} \mathcal{D}_f(i,u;w_i,T(u))}(s_{i-1}, \ldots, s_{i+f-1})\, ds'_1 \cdots ds'_{m-1} ds_{i-1} ds_i \cdots ds_{i+f-1}.$$

(iii) If $S_0 = S'_r$ then

$$\mathrm{Prob}(W_1 = 1 \mid W'_1 = w'_1, \ldots, W'_{m-1} = w'_{m-1}) \ = \tag{11}$$
$$\int\limits_{[0,\gamma^{-1})^{\frac{1}{2}m+1}} g(s'_1, \ldots, s'_{m-1} \mid w'_1, \ldots, w'_{m-1}) \cdot \max\{0, 1-\gamma^{-1}+{s'_r}^2\gamma\}\, ds'_1 \cdots ds'_{m-1}.$$

Proof. We first note that $\{(s'_0, \ldots, s'_{m-1}) \in [0,1) \times [0,\gamma^{-1})^{\frac{1}{2}m+1} \mid W'_1 = w'_1, \ldots, W'_{m-1} = w'_{m-1}\} = \bigcap_{i \in F \cup \{2\}} \mathcal{D}'(i, w'_i)$. For the moment let $\Psi \colon [0,1)^{m/2+2} \to [0,1) \times [0,\gamma^{-1})^{m/2+1}$ be given by $\Psi(s'_0, v'_1, \ldots, v'_{m-1}) := (s'_0, s'_1, \ldots, s'_{m-1})$. By this we mean that the coordinates s'_i of image points are determined by instances of the equation (3) of the form $s'_i = \chi\left(s'_{i-2}s'_2\gamma + v'_i\right)$ when $i > 2$ and the similar ones from equations (1) and (2) for $i = 1, 2$. The mapping Ψ is injective (though not surjective) and differentiable almost everywhere with Jacobian 1. As $\Psi(S'_0, V'_1, \ldots, V'_{m-1}) = (S'_0, S'_1, \ldots, S'_{m-1})$ and since the random variables $S'_0, V'_1, \ldots, V'_{m-1}$ are independent and equidistributed on $[0,1)$ the transformation theorem (applied to the inverse Ψ^{-1} where it is differentiable) implies that the random vector $(S'_0, S'_1, \ldots, S'_{m-1})$ has constant density on the image $\Psi([0,1)^{m/2+2})$. The definition of conditional probabilities and computing the marginal density with respect to s'_0 proves (9). The first assertion of (ii) follows immediately from Lemma 3 (in the Appendix) with $G = \mathbb{R}$, $\mu = \lambda$, $\nu = \lambda\,|_{[0,1)}$ and τ (depending on $T(i{-}1)$) denoting the distribution of $S_{i-2}^2\gamma$ or $S_{i-2}S'_j\gamma$ for a particular index $j{\in}F$. Equation (10) can be verified in a similar way to (9). For fixed $s'_0, \ldots, s'_{m-1}$ we define $\Psi_s \colon [0,\gamma^{-1}) \times [0,1)^f \to [0,\gamma^{-1})^{f+1}$ by $\Psi(s_{i-1}, v_i, v_{i+f-1}) := (s_{i-1}, \ldots, s_{i+f-1})$. Again, Ψ_s is almost everywhere differentiable with Jacobian 1, and the transformation theorem completes the proof of (ii) as the random variables $S_{i-1}, V_i, \ldots, V_{i+f-1}$ are independent and equidistributed on $[0,\gamma^{-1})$ or $[0,1)$, resp. (Note that $\Psi_s^{-1}(s_{i-1}, *) = (s_{i-1}, *)$.) The first Montgomery multiplication in the computation phase is a squaring. Hence $\mathrm{Prob}(W_1 = 1) = \mathrm{Prob}({S'_r}^2\gamma + V_1 \ge \gamma^{-1})$. As ${S'_r}^2\gamma \in [0,\gamma^{-1})$ this proves (iii). □

Remark 2. In the Appendix, Theorem 2 (i) to (iv) and (v) respectively consider the fictional situations that the computation phase consists only of squarings or multiplications by a fixed table entry. The distributions of the $S_1, S_2, \ldots$ then converge respectively to $f \cdot \lambda_{[0,\gamma^{-1})}$ and $f_{(s'_j)} \cdot \lambda_{[0,\gamma^{-1})}$ for fixed $s'_j \in [0,\gamma^{-1})$. These were graphed in Figure 1 for $\gamma = 0.525$ and $C^{(j)} = 1.5\gamma R$. In fact, the density $h_{i-1}(\cdot)$ depends on $T(1), \ldots, T(i{-}1)$ and $s'_1, \ldots, s'_{m-1}$ (cf. Sect. 10).

Theorem 1 quantifies the probabilities for the different type vectors of the Montgomery multiplications $i, \ldots, i{+}f{-}1$ given the observed extra reductions:

Theorem 1. *Let* $\theta = (\omega_i, \ldots, \omega_{i+f-1}) \in \mathcal{T}^f$. *If* $T(i) = \omega_i,\ \ldots,\ T(i{+}f{-}1) = \omega_{i+f-1}$ *then let* $p_\theta\left((\boldsymbol{q}_{i,\ldots,i+f-1;k})_{1\le k\le N} \mid (\boldsymbol{q}'_k)_{1\le k\le N}\right)$ *denote the conditional probability for the er-vectors* $(\boldsymbol{q}_{i,\ldots,i+f-1;k})_{1\le k\le N}$ *if* $(\boldsymbol{q}'_k)_{1\le k\le N}$ *were observed in the initialization phase. Then,*

$$p_\theta((\boldsymbol{q}_{i,\ldots,i+f-1;k})_{1\le k\le N} \,|\, (\boldsymbol{q}'_k)_{1\le k\le N}) \approx \prod_{k=1}^{N} \int\limits_{[0,\gamma^{-1})^{\frac{1}{2}m+f+2}} g(s'_1, \ldots, s'_{m-1} \,|\, q'_{1,k}, \ldots, q'_{m-1,k}) \ \times$$

$$\times\, h_{i-1}(s_{i-1}) \cdot 1_{\cap_{u=i}^{i+f-1} \mathcal{D}_f(i,u;w_u,\omega_u)}(s_{i-1}, \ldots, s_{i+f-1})\, ds'_1 \ldots ds'_{m-1} ds_{i-1} \ldots ds_{i+f-1} \quad (12)$$

If r *is the left-most block (i.e. digit) of the secret exponent after recoding then*

$$\mathrm{Prob}\left((q_{1,k})_{1\le k\le N} \mid (\boldsymbol{q}'_k)_{1\le k\le N}\right) \approx \quad (13)$$
$$\prod_{k=1}^{N} \int\limits_{[0,\gamma^{-1})^{\frac{1}{2}m+1}} g(s'_1, \ldots, s'_{m-1} \mid q'_{1,k}, \ldots, q'_{m-1,k}) \max\{0, 1{-}\gamma^{-1}{+}{s'_r}^2\gamma\}\, ds'_1 \ldots ds'_{m-1}.$$

Proof. According to our mathematical model we interpret the observed er-vectors $\boldsymbol{q}'_k$ and $\boldsymbol{q}_{i,\ldots,i+f-1;k}$ as realizations of random variables $W'_{1,k}, \ldots, W'_{m-1,k}$ and $W_{i,k}, \ldots, W_{i+f-1,k}$ respectively, where the latter correspond to $T(i) = \omega_i$, ..., $T(i{+}f{-}1) = \omega_{i+f-1}$. Theorem 1 is an immediate consequence of Lemma 2 and the mathematical model. □

7 A priori Distribution

In Section 9 we determine an optimal decision strategy for simultaneous guessing of the types $T(i), \ldots, T(i{+}f{-}1)$ of the ith,...,$(i{+}f{-}1)$th Montgomery multiplications. It seems to be reasonable for the attacker to choose the hypothesis θ within the set $\Theta \subseteq \mathcal{T}^f$ of all admissible hypotheses which is the most likely one, given the observed er-vectors. In the sliding window exponentiation scheme a multiplication using a particular table entry is preceded by at least $b = \log_2 m$ squarings. Consequently, for the chosen f,

$$\Theta = \theta_0 \cup \{\theta_{k,j} \mid 1 \le k \le f; 1 \le j \le m{-}1 \text{ for odd } j\} \qquad \text{if } f \le b{+}1 \qquad (14)$$

where

$\theta_0 :=$ (‘S’$, \ldots,$ ‘S’) means $T(i) =$ ‘S’, ..., $T(i{+}f{-}1) =$ ‘S’ and
$\theta_{k,j} :=$ (‘S’$, \ldots,$ ‘S’, ‘M_j’, ‘S’$, \ldots,$ ‘S’) means
$T(i{+}k{-}1) =$ ‘M_j’ but $T(v) =$ ‘S’ for $v \neq i{+}k{-}1$.

However, the admissible hypotheses occur with different probabilities. The optimal decision strategy in Section 9 exploits this fact. In the present section we determine a distribution η on Θ which approximates the exact distribution of the admissible hypotheses and which depends on the secret key. We call η the (approximate) *a priori* distribution, and $\eta_{k,j}$ denotes the approximate probability that $(T(i),\ldots,T(i{+}f{-}1)) = \theta_{k,j}$ for randomly chosen i.

When re-coding, the secret exponent d is divided into blocks (digits) of length 1 and b. If d is assumed to be random then both block lengths should occur with the same frequency, and the average block length is about $(b{+}1)/2$. Hence we should expect about $n/(b{+}1)$ blocks of length b and $n/(b{+}1)$ blocks of length 1 where n is the bit length of the modulus M. Consequently, about $2n/(b{+}1)m$ blocks of length b should equal any given odd exponent digit j. Thus we expect this many vectors $(T(i),\ldots,T(i{+}f{-}1))$ of type $\theta_{k,j}$ for $1 \le k \le f$. As there are about $n + n/(b{+}1) = (b{+}2)n/(b{+}1)$ Montgomery multiplications (including squares) we set

$$\eta_{1,1} \;:=\; \cdots \;:=\; \eta_{f,m-1} \;:=\; \frac{n(b{+}1)}{(b{+}1)\frac{1}{2}m(b{+}2)n} \;=\; \frac{1}{\frac{1}{2}m(b{+}2)} \qquad \text{and}$$

$$\eta_0 \;:=\; 1 - \frac{\frac{1}{2}mf}{(b{+}2)\frac{1}{2}m} \;=\; \frac{b{+}2{-}f}{b{+}2} \qquad \text{if } f \le b{+}1. \tag{15}$$

8 Error Detection and Correction

It seems to be unhelpful to consider error detection and error correction strategies before the decision strategy itself has been derived. However, the optimal decision strategy considers the different types of possible error. Roughly speaking, it tries to avoid estimation errors but 'favours' those kinds of errors which are easier to detect and correct than others. The following example illuminates the situation.

Example 1. Let $b = 2$ (i.e. $m = 4$), and assume that the secret exponent d is given by $\ldots|0|01|0|01|0|\ldots$. The correct type sequence is then given by
$\ldots$,'S','S','S','M_1','S','S', 'S','M_1','S',$\ldots$
whereas the following a), b) and c) are possible estimation sequences:
a) $\ldots$,'S','S','S','M_1','S','M_3','S','M_1','S',$\ldots$
b) $\ldots$,'S','S','S','S', 'S','S', 'S','M_1','S',$\ldots$
c) $\ldots$,'S','S','S','M_3','S','S', 'S','M_1','S',$\ldots$

Each of the subsequences a), b), and c) contains exactly one false guess. The error in a) ('M_3') is obvious as the number of squarings between two multiplications must be at least $b = 2$. This *type-a error* ('M_j' instead of 'S') is usually easy to detect if its occurrence is isolated, i.e. if there are no further type-a or *type-b errors* ('S' instead of 'M_j') within a neighbourhood of the error. Then one or at most two positions remain for which exactly one guess is false, and we call the type-a error *locally correctable.* A type-a error is not locally correctable if it occurs within a long series of squarings or if bursts of type-a and type-b errors occur. Then we call it a *global* type-a error. The type-b errors and *type-c errors* ('M_i' instead of 'M_j') illustrated in sequences b) and c) are less obvious. If all type-a errors have been corrected the attacker knows the number of type-b errors (since the number of squarings equals the bit length d minus 1) but not their positions. In particular, type-b and type-c errors are global errors. Of course, to correct type-b and type-c errors it is reasonable first to change those guesses where the respective decisions have been “close” and then to check the new exponent estimator (cf. Sect. 10).

9 The Optimal Decision Strategy

The preliminary work has now been done. Here the pieces are assembled to derive an optimal decision strategy for guessing the types of f consecutive Montgomery multiplications simultaneously when $1 \leq f \leq b{+}1$. Therefore, we interpret the estimation of $T(i), \ldots, T(i{+}f{-}1)$ as a statistical decision problem.

Roughly speaking, in a statistical decision problem the statistician (here the attacker) observes a sample $\omega \in \Omega$ (here the observed extra reduction vectors $(\boldsymbol{q}'_k, \boldsymbol{q}_{i,\ldots,i+f-1,k})_{1\leq k\leq N}$) which he interprets as a realization of a random variable X. The distribution p_θ of X depends on the unknown parameter $\theta \in \Theta$ which has to be guessed (here $\theta = (T(i), \ldots, T(i{+}f{-}1))$). The decision strategy clearly depends on the observation ω but also considers the *a priori* distribution η (cf. Sect. 7) which quantifies the likeliness of the possible parameters and the ‘damage’ caused by the possible guessing errors. The ‘damage’ is quantified by the loss function $s(\theta, a)$ where $\theta \in \Theta$ denotes the true parameter whereas $a \in \Theta$ stands for a potential guess. In our case the loss function corresponds to the effort which is necessary for the detection, localization and correction of wrong guesses (cf. Sects. 8 and 10). Of course, correct decisions do not cause any loss, i.e. $s(\theta, \theta) = 0$ for all $\theta \in \Theta$. A decision strategy is optimal if its expected loss attains a minimum.

Optimal Decision Strategy. Let the *a priori* distribution η be given by (15). Let $\tau_{opt}((\boldsymbol{q}'_k, \boldsymbol{q}_{i,\ldots,i+f-1,k})_{1\leq k\leq N}) := a^*$ if the sum

$$\sum_{\theta\in\Theta} s(\theta, a') p_\theta\left((\boldsymbol{q}_{i,\ldots,i+f-1;k} \mid \boldsymbol{q}'_k)_{1\leq k\leq N}\right) \eta(\theta) \tag{16}$$

is minimal for $a' = a^*$ (i.e. the attacker picks $a^* \in \Theta$ when he observes the vector $(\boldsymbol{q}'_k, \boldsymbol{q}_{i,\ldots,i+f-1,k})_{1\leq k\leq N}$). Then τ_{opt} is optimal among all decision strategies which estimate $T(i), \ldots, T(i{+}f{-}1)$ simultaneously.

Proof. The proof of the analogous assertion in Section 7 of [12] can be re-applied here almost literally. The conditional probabilities $p_\theta(\cdot|\cdot)$ were computed in Theorem 1. □

10 Experimental Results

Although theoretical considerations were not significantly more difficult for $f \geq 1$, in this section we fix $f{=}1$ in order to simplify the calculations (cf. [12], §8). Thus the types of the particular Montgomery multiplications are guessed separately. The restriction on f means it is convenient to attack a sliding window exponentiation scheme, and $m = 4$ (i.e. $b = 2$) was chosen for a simulation. Extra reduction values $q'_{1,k}, \ldots, q'_{m-1,k}$ and $q_{1,k}, q_{2,k}, \ldots$ were obtained from the simulation using pseudo-randomly chosen moduli M and inputs $C_1, \ldots, C_N$.

We first determined the optimal decision strategy given in Section 9. We used the loss function given by $s(\text{'}S\text{'}, \text{'}M_j\text{'}) = 1$ (type-a error), $s(\text{'}M_j\text{'}, \text{'}S\text{'}) = 1.5$ (type-b error) and $s(\text{'}M_i\text{'}, \text{'}M_j\text{'}) = 2.5$ for $i \neq j$ (type-c error). Equation (15) gives the *a priori* distribution $\eta(\text{'}S\text{'}) = 0.75$ and $\eta(\text{'}M_1\text{'}) = \eta(\text{'}M_3\text{'}) = 0.125$. Next, we computed approximations as follows for the density $h_{i-1(k)}$ for each of the three alternatives $\theta = \text{'}S\text{'}$, $\theta = \text{'}M_1\text{'}$ and $\theta = \text{'}M_3\text{'}$. First, Theorem 2 was applied to obtain the 'pure' limit densities f (cf. Thm. 2(iii)) and $f_{(s)}$ (cf. Thm. 2(v)). The iterates of the densities $f_{up}(x) := 1_{[0,1)}(x)$ and $f_{low}(x) := 1_{[\gamma^{-1}-1,\gamma^{-1})}$ (cf. Thm. 2(iv)) squeeze the respective limit distribution. The convergence is monotonic and exponentially fast (Thm. 2(iv)). If $T(i) = \text{'}M_j\text{'}$ then at least two squarings had been carried out just before. As the convergence to f is exponentially fast we assumed $h_{i-1(k)} \approx f$ in that case. For the hypothesis $T(i) = \text{'}S\text{'}$ we set $h_{i-1(k)} := \eta_0 \cdot f + \eta_1 \cdot f_{(s_1)} + \eta_3 \cdot f_{(s_3)}$ with $\eta_1 = \eta_3 = 0.125$ and $\eta_0 = 0.75$, and s_j denoting the ratio of table entry j divided by the modulus M. Then we put the pieces together, determined the conditional probabilities $p_{\text{'}S\text{'}}(\cdot)$, $p_{\text{'}M_1\text{'}}(\cdot)$ and $p_{\text{'}M_3\text{'}}(\cdot)$ using Theorem 1, and so derived the optimal decision strategy.

Table 1. Average number of errors per 100 guesses with $b = 2$, $f = 1$.

M/R	N	type-a	global type-a	type-b	type-c
0.99	350	0.53	0.11	0.29	0.67
0.99	400	0.37	0.07	0.21	0.04
0.85	400	0.74	1.58	0.12	0.06
0.85	450	0.54	0.11	0.62	0.03
0.85	500	0.44	0.08	0.03	0.25
0.70	700	1.24	0.19	0.22	0.35

Applying this optimal decision strategy we obtained guesses $\widetilde{T}(1), \widetilde{T}(2), \ldots$. A large number of simulation runs gave the results in Tables 1 to 3. The "type-a" column in Table 1 covers all errors of type a, namely both the locally correctable and global type-a errors. In a first step the attacker corrects the locally correctable type-a errors. Usually, he knows a reference equation $y^d \equiv x(\text{mod} M)$, e.g. a signature. Using this he can check whether a guess $\widetilde{d}$ for d is correct. As already observed in Section 8, the number of global errors is relevant for the practical feasibility of the attack. This is the sum of the type-b, type-c and global type-a errors. Table 2 gives the percentage of trials for which the number of such global errors is no more than a given bound. For example, at most one global error occurred in 76% of the trials for the parameter set $M/R \approx 0.85$, $N = 500$, $n = 512$. Clearly, for the sake of efficiency the attacker first tries to change those guesses for which the decision has been 'close'.

A successful attack on a 512-bit exponent requires about 680 correct guesses. For this, about $2 \cdot 680 = 1360$ hypotheses have to be rejected. To each rejected

Table 2. Number of global errors.

M/R	N	n	0	≤ 1	≤ 2	≤ 3
0.99	350	512	10%	31%	49%	64%
0.99	400	512	16%	46%	62%	78%
0.85	400	512	19%	43%	60%	71%
0.85	450	512	33%	62%	80%	90%
0.85	500	512	46%	76%	90%	97%
0.70	700	512	35%	60%	71%	76%

hypothesis about the types of a sequence of f=1 multiplicative operations, we assign the ratio between the expected loss if this hypothesis had been chosen divided by the expected loss for the hypothesis chosen in this decision. The rejected hypotheses are ordered using these ratios, that with the smallest ratio (i.e. the most likely alternative) first. If the estimator $\widetilde{d}$ is false the attacker replaces one, two or three guesses respectively by those from the rejected list, beginning with the first, i.e. that with lowest ratio.

Table 3 gives the average rank of the lowest correct hypothesis which has been rejected for a given number of global errors. For instance, if there were three global errors, and the correct guesses for them were ranked 3, 29 and 53 in the list, then the rank of the lowest would be 53. The second row of the table says that if there are exactly three global errors for parameters $M/R \approx 0.99$, $N = 400$, $n = 512$ then 57 is the average rank of the lowest correct hypothesis which was initially rejected. If the lowest rank is ≤ 100 (which would normally be the case for an average of 57), the correction requires at most $\binom{100}{3} = 161700$ evaluations of the reference equation (neglecting the unsuccessful efforts to correct exactly 1 or 2 global errors). This is clearly computationally feasible.

Table 3. Average rank of the last correct hypothesis for 1, 2 or 3 global errors.

M/R	N	n	1	2	3
0.99	350	512	31	66	63
0.99	400	512	30	25	57
0.85	400	512	39	57	55
0.85	450	512	22	37	59
0.85	500	512	24	57	70
0.70	700	512	63	132	271

The conditional probabilities (Section 6), and hence the optimal decision strategy, only depend on the ratio $\gamma = M/R$. In our simulations we assumed that the attacker knows this ratio. However, our attack is also feasible if the attacked device uses the Chinese Remainder Theorem (CRT). Then the attacker uses the extra reductions within the initialization phase and known squarings from the computation stage to estimate the parameter γ. The moduli M for the exponentiations are the prime factors of the RSA modulus M'=p_1p_2. For the secret RSA key d', the attacker guesses the exponents actually used, namely $d =$

$d'(\mathrm{mod}\,(p_i{-}1))$ for $i = 1$ or 2. If the guess $\widetilde{d}$ for d is correct then $\gcd(C^{d'}(\mathrm{mod}\, M') - C^{\widetilde{d}}(\mathrm{mod}\, M'), M') = p_i$ and, similarly, $\gcd(C - C^{\widetilde{de}}(\mathrm{mod}\, M'), M') = p_i$. Since the parameter γ has to be estimated the error probability for a singular decision increases somewhat (cf. [12], first paragraph of Sect.10). However, since CRT involves an exponent of only half the length, the number of wrong guesses per exponentiation will be smaller and so the attack more likely to be successful.

11 Conclusion

A timing attack on RSA implementations has been given further detail in the more complex situation of modular reductions being driven by a register length bound rather than a modulus bound. Graphs of the limiting distributions were drawn illustrating one source of the attack. Another source was the combination of exact conditional probabilities for the modular reductions with statistical decision theory for treating sequences of modular multiplications. This reduces the sample size necessary to deduce the secret RSA key from side channel leakage. The resulting powerful methods reduce the number of errors far enough for their correction to be computationally feasible for keys with standard lengths using data obtained well within the normal lifespan of a key.

References

1. J.-F. Dhem, F. Koeune, P.-A. Leroux, P. Mestré, J.-J. Quisquater & J.-L. Willems, *A practical implementation of the Timing Attack*, Proc. CARDIS 1998, J.-J. Quisquater & B. Schneier (editors), LNCS **1820**, Springer-Verlag, 2000, pp. 175–190.
2. D. E. Knuth, The Art of Computer Programming, vol. 2, *Seminumerical Algorithms*, 2nd Edition, Addison-Wesley, 1981, pp. 441–466.
3. Ç. K. Koç, *Analysis of Sliding Window Techniques for Exponentiation*, Computers and Mathematics with Applications **30**, no. 10, 1995, pp. 17–24.
4. P. Kocher, *Timing attack on implementations of Diffie-Hellman, RSA, DSS, and other systems*, Proc. Crypto 96, N. Koblitz (editor), LNCS **1109**, Springer-Verlag, 1996, pp. 104–113.
5. P. Kocher, J. Jaffe & B. Jun, *Differential Power Analysis*, Advances in Cryptology – Crypto '99, M. Wiener (editor), LNCS **1666**, Springer-Verlag, 1999, pp. 388–397.
6. R. Mayer-Sommer, *Smartly Analyzing the Simplicity and the Power of Simple Power Analysis on Smartcards*, Cryptographic Hardware and Embedded Systems (Proc CHES 2000), C. Paar & Ç. Koç (editors), LNCS **1965**, Springer-Verlag, 2000, pp. 78–92.
7. T. S. Messerges, E. A. Dabbish, R. H. Sloan, *Power Analysis Attacks of Modular Exponentiation in Smartcards*, Cryptographic Hardware and Embedded Systems (Proc CHES 99), C. Paar & Ç. Koç (editors), LNCS **1717**, Springer-Verlag, 1999, pp. 144–157.
8. V. Miller, *Use of Elliptic Curves in Cryptography*, Proc. CRYPTO '85, H. C. Williams (editor), LNCS **218**, Springer-Verlag, 1986, pp. 417–426.
9. P. L. Montgomery, *Modular Multiplication without Trial Division*, Mathematics of Computation **44**, no. 170, 1985, pp. 519–521.

10. R. L. Rivest, A. Shamir & L. Adleman, *A Method for obtaining Digital Signatures and Public-Key Cryptosystems*, Comm. ACM **21**, 1978, pp. 120–126.
11. W. Schindler, *A Timing Attack against RSA with Chinese Remainder Theorem*, Cryptographic Hardware and Embedded Systems (Proc CHES 2000), C. Paar & Ç. Koç (editors), LNCS **1965**, Springer-Verlag, 2000, pp. 109–124.
12. W. Schindler, *A Combined Timing and Power Attack*, Public Key Cryptography (Proc PKC 2002), P. Paillier & D. Naccache (editors), LNCS **2274**, Springer-Verlag, 2002, pp. 263–279.
13. W. Schindler, *Optimized Timing Attacks against Public Key Cryptosystems*, Statistics & Decisions **20**, 2002, pp. 191–210.
14. C. D. Walter, *Montgomery Exponentiation Needs No Final Subtractions*, Electronics Letters **35**, no. 21, October 1999, pp. 1831–1832.
15. C. D. Walter, *Precise Bounds for Montgomery Modular Multiplication and Some Potentially Insecure RSA Moduli*, Proc. CT-RSA 2002, B. Preneel (editor), LNCS **2271**, Springer-Verlag, 2002, pp. 30–39.
16. C. D. Walter & S. Thompson, *Distinguishing Exponent Digits by Observing Modular Subtractions*, Topics in Cryptology – CT-RSA 2001, D. Naccache (editor), LNCS **2020**, Springer-Verlag, 2001, pp. 192–207.

Appendix

Lemma 3. *Let τ and ν denote probability measures on a locally compact abelian group G. If μ is a Haar measure on G and ν has a μ-density then the convolution product $\tau * \nu$ also has a μ-density.*

Proof. For any measurable $B \subseteq G$ we have $\tau * \nu(B) = \int_G \nu(B{-}x)\,\tau(dx)$. If $\mu(B) = 0$ then $\mu(B{-}x) = 0$ and hence $\nu(B{-}x) = 0$. This proves the lemma. □

Theorem 2. *Suppose $\gamma \in (0.5, 1)$, and let F and $\chi\colon [0, \gamma^{-1}+1) \to \mathbb{R}$ be defined as in Section 6. Assume further that $U_0, V_1, V_2, \ldots$ denote independent random variables where U_0 assumes values on $[0, \gamma^{-1})$, while $V_1, V_2, \ldots$ are equidistributed on the unit interval $[0, 1)$. Finally, let $U_{n+1} := \chi(U_n^2\gamma + V_{n+1})$ for all $n \in \mathbb{N}$.*
(i) Regardless of the distribution of U_0, for each $n \geq 1$ the distribution μ_n of U_n has a Lebesgue density f_n. Moreover, $f_n(x) = 1$ for $x \in [\gamma^{-1}{-}1, 1)$ and $f_n(x) + f_n(x{+}1) = 1$ for $x \in [0, \gamma^{-1}{-}1)$.
(ii) For $n \geq 2$ we have

$$f_{n+1}(x) = \int_0^{\sqrt{x\gamma^{-1}}} f_n(u)\,du \qquad \text{for } x \in [0, \gamma^{-1}{-}1). \tag{17}$$

(iii) Regardless of the distribution of U_0 we have

$$\|f_{n+1} - f_n\|_\infty := \sup_{x \in [0,\gamma^{-1})} |f_{n+1}(x) - f_n(x)| \;\leq\; \left(\gamma^{-1}{-}1\right)^{n-1} \|f_2 - f_1\|_\infty \tag{18}$$

for all $n \geq 2$. In particular, the sequence $f_1, f_2, \ldots$ converges uniformly to a probability density $f\colon [0, \gamma^{-1}) \to [0, 1]$ which does not depend on the distribution

of U_0. To be precise, $f(x) = 1$ for $x \in [\gamma^{-1}{-}1, 1)$, $f(x) + f(x{+}1) = 1$ for $x \in [0, \gamma^{-1}{-}1)$, and f is the unique solution of (17) possessing these properties, i.e. (17) holds with f in place of f_n and f_{n+1}.
(iv) If $f_1 \le f$ on $[0, \gamma^{-1}{-}1)$ then also $f_n \le f$ for all $n \ge 1$. If $f_1 \le f_2$ then the sequence $f_1, f_2, \ldots$ is monotonically increasing on $[0, \gamma^{-1}{-}1)$. Similarly, $f_1 \ge f$ on $[0, \gamma^{-1}{-}1)$ implies $f_n \ge f$ for all $n \ge 1$, and if $f_1 \ge f_2$ then the sequence $f_1, f_2, \ldots$ is monotonically decreasing on $[0, \gamma^{-1}{-}1)$. If the distribution of U_0 has a Lebesgue density f_0 then the assertions of (iv) even hold for $f_0, f_1, \ldots$ in place of $f_1, f_2, \ldots$.
(v) Let $U'_0 := U_0$ and $U'_{n+1} := \chi(U'_n s\gamma + V_{n+1})$ for all $n \in \mathbb{N}$ where $s \in [0, \gamma^{-1})$ is fixed. Then all assertions from (i) to (iv) can be transferred almost literally to this case. In particular, for $x \in [0, \gamma^{-1}{-}1)$ we have

$$f_{n+1}(x) = \begin{cases} \int_0^{x/s\gamma} f_n(u)\,du & \text{if } x < s \\ 1 & \text{otherwise.} \end{cases} \tag{19}$$

The limit distribution $f_{(s)}$ of $f_1, f_2, \ldots$ is the unique solution of (19) with $f_{(s)}(x){=}1$ for $x \in [\gamma^{-1}{-}1, 1)$ and such that $f_{(s)}(x) + f_{(s)}(x{+}1) = 1$ for $x \in [0, \gamma^{-1}{-}1)$.

Proof. To prove the first assertion of (i) we apply Lemma 3 with $G = \mathbb{R}$ and $\mu = \lambda$ while τ is the distribution of $U^2_{n-1}\gamma$ and ν the restriction of the Lebesgue measure to $[0, 1)$. If $\gamma^{-1}{-}1 \le a < b \le 1$ we have $U_n \in [a, b)$ iff $U^2_{n-1}\gamma + V_n \in [a, b) \cup [a{+}1, b{+}1)$. As U_{n-1} and V_n are independent, and χ coincides with the reduction modulo 1 on this union and V_n is equidistributed on $[0, 1)$, this proves the second assertion of (i). However, for any $0 \le a < b \le \gamma^{-1}{-}1$, we have $U_n \in [a, b) \cup [a{+}1, b{+}1)$ iff $U^2_{n-1}\gamma + V_n \in [a, b) \cup [a{+}1, b{+}1) \cup [a{+}2, b{+}2)$. From $\bmod 1 = \bmod 1 \circ \chi$ we obtain the final assertion of (i) as V_n is equidistributed on $[0, 1)$. For $x \in [0, \gamma^{-1}{-}1)$ the pre-image $\chi^{-1}(x)$ equals x and hence

$$\begin{aligned} \text{Prob}(U_{n+1} \le x) = \int_0^x f_{n+1}(u)\,du \; &= \int_0^{\sqrt{x\gamma^{-1}}} f_n(u)\text{Prob}(u^2\gamma + V_n \le x)\,du \\ &= \int_0^{\sqrt{x\gamma^{-1}}} f_n(u)(x - u^2\gamma)\,du \\ &= x\int_0^{\sqrt{x\gamma^{-1}}} f_n(u)\,du - \gamma\int_0^{\sqrt{x\gamma^{-1}}} f_n(u)u^2\,du. \end{aligned}$$

If f_n is continuous at $\sqrt{x\gamma^{-1}}$ differentiating the left- and the right-hand side verifies assertion (ii) for $f_{n+1}(x)$. We denote the density of $U_1^2\gamma + V_2$ by h_2 for the moment. Applying the convolution formula for densities (to that of $U_1^2\gamma$ and V_2) yields $|h_2(x + s) - h_2(x)| \le \text{Prob}(U_1^2\gamma \in [-s, 0] \cup [1, 1 + s])$ for $|s| < 0.5$. As $[-s, 0] \cup [1, 1 + s]$ converges to $\{0, 1\}$ the right-hand probability converges to 0 as $s \to 0$, i.e. h_2 is continuous at x. As x was arbitrary this proves the continuity of h_2, and f_2 has at most two discontinuity points in $(0, \gamma^{-1})$, namely $\gamma^{-1}{-}1$ and 1. This proves (ii) for $n = 1$ since densities may be defined arbitrarily on zero sets. For arbitrary n, assertion (ii) follows by induction. Clearly,

$$\begin{aligned}\|f_{n+1}-f_n\|_\infty &= \sup_{x\in[0,\gamma^{-1}-1)} |f_{n+1}(x)-f_n(x)|\\ &= \sup_{x\in[0,\gamma^{-1}-1)} \left|\int_0^{\sqrt{x\gamma^{-1}}} (f_n(u)-f_{n-1}(u))\,du\right|\\ &\le \sup_{y,z\in[0,\gamma^{-1}-1)} \left|\int_y^z (f_n(u)-f_{n-1}(u))\,du\right| \;\le\; \left(\gamma^{-1}{-}1\right)\|f_n-f_{n-1}\|_\infty.\end{aligned}$$

The first inequality follows from the fact that $f_n(x)=f_{n-1}(x)=1$ on $[\gamma^{-1}{-}1,1)$ while $f_n(x)+f_n(x{+}1)=1$ for $x\in[0,\gamma^{-1}{-}1)$. Equation (18) follows by induction. Similarly, one concludes $\|f_n-f_n^*\|_\infty \le (\gamma^{-1}{-}1)^{n-2}\|f_2-f_2^*\|_\infty$ for arbitrary sequences $f_1,f_2,\ldots$ and $f_1^*,f_2^*,\ldots$. This shows the uniqueness of f and, as $f_1,f_2,\ldots$ converges uniformly, f fulfils (17). If, in addition, $f_1(x)\le f_2(x)$ for all $x\in[0,\gamma^{-1}{-}1)$ then by induction we obtain

$$\begin{aligned}f_{n+1}(x)-f_n(x) &= \int_0^{\sqrt{x\gamma^{-1}}} (f_n(u)-f_{n-1}(u))\,du\\ &= \int_{[0,\sqrt{x\gamma^{-1}})\cap[0,\gamma^{-1}-1)\cap\{u:u>\sqrt{x\gamma^{-1}}-1\}} (f_n(u)-f_{n-1}(u))\,du \;\ge\; 0.\end{aligned}$$

Replacing f_n by f we obtain the first assertion of (iv), and the second part can be shown similarly. Assertion (v) can be verified in same way as (i) to (iv). □