INTERNATIONAL STANDARD

ISO/IEC 9796

First edition
1991-09-15

Information technology — Security techniques — Digital signature scheme giving message recovery

Technologies de l'information — Techniques de sécurité — Schéma de signature numérique rétablissant le message

Reference number
ISO/IEC 9796:1991(E)

Contents

Annexes

ISO/IEC Copyright Office • Case postale 56 • CH-1211 Genève 20 • Switzerland
Printed in France

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

International Standard ISO/IEC 9796 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*.

Annexes A, B, C and D are for information only.

Introduction

A digital signature in electronic exchange of information is a counterpart to a handwritten signature in classical mail.

Most digital signature schemes are based upon a particular public-key system. Any public-key system includes three basic operations:

— a process producing pairs of keys: a secret key and a public key;

— a process using a secret key;

— a process using a public key.

In any public-key digital signature scheme, the secret key is involved in a signature process for signing messages, and the public key is involved in a verification process for verifying signatures. A pair of keys for a digital signature scheme thus consists of a "secret signature key" and a "public verification key".

Two types of digital signature schemes are clearly identified.

— When the verification process needs the message as part of the input, the scheme is named a "signature scheme with appendix". The use of a hash-function is involved in the calculation of the appendix.

— When the verification process reveals the message together with its specific redundancy (sometimes called the "shadow of a message"), the scheme is named a "signature scheme giving message recovery".

This International Standard specifies a scheme for digital signature of messages of limited length.

This digital signature scheme allows a minimal resource requirement for verification. It does not involve the use of a hash-function and it avoids the known attacks against the generic algorithm in use.

The message need not be in a natural language. It may be any arbitrary string of bits of limited length. Examples of such messages are cryptographic key materials and the result of hashing another, longer message, which is also called the "imprint of a message". A characteristic example is a structured set of a few strings of bits generated by cryptographic software and hardware, one of these strings coding control information produced within the hardware.

NOTE — The use of this International Standard may involve patented items.

Information technology — Security techniques — Digital signature scheme giving message recovery

1 Scope

This International Standard specifies a digital signature scheme giving message recovery for messages of limited length and using a public-key system.

This digital signature scheme includes

— a signature process using a secret signature key and a signature function for signing messages;

— a verification process using a public verification key and a verification function for checking signatures while recovering messages.

During the signature process, messages to be signed are padded and extended if necessary. Artificial redundancy is then added, depending upon the message itself. No assumption is made as to the possible presence of natural redundancy in the messages. The artificial redundancy is revealed by the verification process. The removal of this artificial redundancy gives message recovery.

This International Standard does not specify the key production process, the signature function and the verification function. Annex A gives an example of a public-key system including key production, signature function and verification function. The various steps of these operations are illustrated by examples in annex B.

Some parameters in the scheme are related to security: this International Standard does not specify the values to be used in order to reach a given level of security. However, this International Standard is specified in such a way as to minimize the required changes in its use if some of these parameters have to be modified.

2 Definitions

For the purposes of this International Standard, the following definitions apply.

2.1 message: String of bits of limited length.

2.2 signature: String of bits resulting from the signature process.

3 Symbols and abbreviations

MP	Padded message
ME	Extended message
MR	Extended message with redundancy
IR	Intermediate integer
Σ	Signature
k_s	Length of the signature in bits
IR'	Recovered intermediate integer
MR'	Recovered message with redundancy
MP'	Recovered padded message
Sign	Signature function under control of the secret signature key
Verif	Verification function under control of the public verification key
mod z	Arithmetic computation modulo z
μ	Nibble
Π	Permutation of the nibbles
m	Byte
S	Shadow of the bytes
$X \parallel Y$	Concatenation of strings of bits X and Y
$X \oplus Y$	Exclusive-or of strings of bits X and Y

NOTES

1 All integers (and all strings of bits or bytes) are written with the most significant digit (or bit or byte) in left position.

2 The hexadecimal notation, with the digits 0 to 9 and A to F, is used in table 1 and in annex B.

4 General overview

The next two clauses specify

— the signature process in clause 5;

— the verification process in clause 6.

Each signing entity shall use and keep secret its own signature key corresponding to its own public verification key.

Messages to be signed shall be padded and extended if necessary. Redundancy is then added according to rules specified in clause 5. From the extended messages with redundancy, signatures shall be computed using the secret signature key as specified in clause 5.

Each verifying entity should know and use the public verification key specific to the signing entity. A signature shall be accepted if and only if the verification process specified in clause 6 is successful.

NOTE — The production and the distribution of keys fall outside the scope of this International Standard.

5 Signature process

Figure 1 summarizes the signature process.

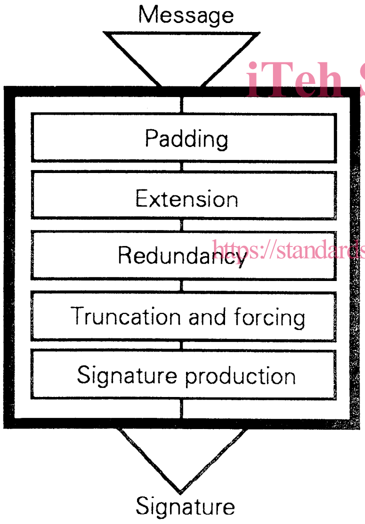

Figure 1 — Signature process

NOTE — A good implementation of the signature process should physically protect the operations in such a way that there is no direct access to the signature function under control of the secret signature key.

5.1 Padding

The message is a string of bits. This string of bits is padded to the left by 0 to 7 zeroes so as to obtain a string of z bytes. Index r, to be used later on, is the number of padded zeroes plus one. Index r is thus valued from 1 to 8.

Consequently, in the padded message denoted by MP, the $8z+1-r$ least significant bits are information bearing.

$$MP = m_z \,\|\, m_{z-1} \,\|\, \ldots \, m_2 \,\|\, m_1$$

$$m_z = (r-1 \text{ padded zeroes}) \,\|\, (9-r \text{ information bits})$$

Number z multiplied by sixteen shall be less than or equal to number k_s+3. Consequently, the number of bits of the message to be signed shall be at most 8 times the largest integer less than or equal to $(k_s+3)/16$.

5.2 Extension

Number t, to be used later on, is the least integer such that a string of $2t$ bytes includes at least k_s-1 bits.

The extended message ME is obtained by repeating the z bytes of MP, as many times as necessary, in order and concatenated to the left, until forming a string of t bytes.

For i valued from 1 to t and j equal to $i-1$ (mod z) plus one (j is therefore valued from 1 to z), the i-th byte of ME equals the j-th byte of MP.

$$ME = \ldots \, m_z \,\|\, \ldots \, m_2 \,\|\, m_1$$
$$\longleftarrow t \text{ bytes} \longrightarrow$$

NOTE — Number z is less than or equal to number t. The equality may occur only if k_s is congruent to 13, 14, 15, 0 or 1 mod 16.

5.3 Redundancy

The extended message with redundancy MR is obtained by interleaving the t bytes of ME in odd positions and t bytes of redundancy in even positions. Altered by index r, the least significant nibble of the $2z$-th byte of MR codes the message length by its value and its position.

For i valued from 1 to t,

— the $(2i-1)$-th byte of MR equals the i-th byte of ME;

— the $2i$-th byte of MR equals the image of the i-th byte of ME according to the shadow S specified in table 1, except for the $2z$-th byte of MR which equals the exclusive-or of index r with the shadow of the z-th byte of ME.

$$MR = \ldots \, S(m_z) \oplus r \,\|\, m_z \,\|\, \ldots \, S(m_2) \,\|\, m_2 \,\|\, S(m_1) \,\|\, m_1$$
$$\longleftarrow 2t \text{ bytes} \longrightarrow$$

NOTE — The computation of the $2t$ bytes of MR (mr_{2t} to mr_1) from the z bytes of MP (mp_z to mp_1) is performed by applying successively the following three formulae for i valued from 1 to t.

$$j := (i-1 \bmod z)+1 \,; \quad mr_{2i-1} := mp_j \,; \quad mr_{2i} := S(mp_j)$$

Finally, the $2z$-th byte is altered by index r.

$$mr_{2z} := r \oplus mr_{2z}$$

5.4 Truncation and forcing

The intermediate integer IR is coded by a string of k_s bits where the most significant bit is valued to 1 and where the k_s-1 least significant bits are those of MR, except for the least significant byte which is replaced. If $\mu_2 \,\|\, \mu_1$ is the least significant byte of MR, then the least significant byte of IR shall be $\mu_1 \,\|\, 6$.

5.5 Signature production

The signature Σ is obtained as a string of k_s bits by applying to IR the signature function under control of the secret signature key.

$$\Sigma = \text{Sign}(\, IR \,)$$

Table 1 — Permutation Π and shadow S

μ	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
$\Pi(\mu)$	E	3	5	8	9	4	2	F	0	D	B	6	7	A	C	1

If nibble μ consists of the bits $a_4\ a_3\ a_2\ a_1$, then under the permutation Π, its image denoted by $\Pi(\mu)$ consists of the bits $a_4 \oplus a_2 \oplus a_1 \oplus 1$; $a_4 \oplus a_3 \oplus a_1 \oplus 1$; $a_4 \oplus a_3 \oplus a_2 \oplus 1$; $a_3 \oplus a_2 \oplus a_1$.

If byte m consists of the nibbles $\mu_2\ \mu_1$, then under the shadow S, its image denoted by $S(m)$ consists of the nibbles $\Pi(\mu_2)\ \Pi(\mu_1)$.

6 Verification process

Figure 2 summarizes the verification process.

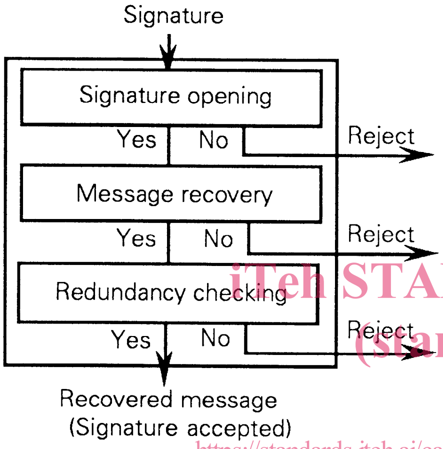

Figure 2 — Verification process

6.1 Signature opening

The signature Σ is transformed into the recovered intermediate integer IR' by applying to Σ the verification function under control of the public verification key.

$$IR' = \mathrm{Verif}(\ \Sigma\ )$$

The signature Σ shall be rejected if IR' is not a string of k_s bits where the most significant bit is valued to 1 and where the least significant nibble is valued to 6.

6.2 Message recovery

The recovered message with redundancy MR' is the string of $2t$ bytes where the $1-k_s$ (mod 16) most significant bits are null and where the k_s-1 least significant bits are those of IR', except for the least significant byte which is replaced. According to the permutation Π specified in table 1, if $\mu_4 \parallel \mu_3 \parallel \mu_2 \parallel 6$ are the four least significant nibbles of IR', then the least significant byte of MR' shall be $\Pi^{-1}(\mu_4) \parallel \mu_2$.

$$MR' = m_{2t} \parallel m_{2t-1} \parallel \dots\ m_2 \parallel m_1$$

NOTE — The strings MR and MR' may be unequal. The string MR' consists of the k_s–1 least significant bits of MR padded by 0 to 15 zeroes in the most significant bits.

From the $2t$ bytes of MR', t sums are computed. According to the shadow S specified in table 1, the i-th sum equals the exclusive-or of the $2i$-th byte with the shadow of the $(2i-1)$-th byte.

$$m_{2i} \oplus S(m_{2i-1})$$

The signature Σ shall be rejected if the t sums are null.

Number z is recovered as the position of the first non-null sum. The recovered padded message MP' is the string of the z least significant bytes in odd positions in MR'.

$$MP' = m_{2z-1} \parallel m_{2z-3} \parallel \dots\ m_{2i-1} \parallel \dots\ m_3 \parallel m_1$$

Index r is recovered as the value of the least significant nibble of the first non-null sum.

The signature Σ shall be rejected if index r is not valued from 1 to 8, and also if the r–1 most significant bits of MP' are not all null.

$$m_{2z-1} = (r\text{–}1 \text{ padded zeroes}) \parallel (9\text{–}r \text{ information bits})$$

The message is recovered as the string of the $8z+1-r$ least significant bits of MP'.

6.3 Redundancy checking

The signature Σ shall be accepted if and only if the k_s-1 least significant bits of MR' are equal to the k_s-1 least significant bits of another extended message with redundancy computed from the recovered padded message MP' according to 5.2 and 5.3.

Annex A
(informative)

Example of a public-key system for digital signature

A.1 Definitions

Modulus: Integer constructed as the product of two primes.

Public verification key: Modulus and verification exponent.

Secret signature key: Signature exponent.

A.2 Symbols and abbreviations

RR	Representative element
IS	Resulting integer
n	Modulus
k	Length of the modulus in bits
p, q	Prime factors of the modulus
v	Verification exponent
s	Signature exponent
$\mathrm{lcm}(a, b)$	Least common multiple of integers a and b
$(a \mid n)$	Jacobi symbol of a with respect to n

NOTE — Let p be an odd prime, and let a be a positive integer. The Legendre symbol of integer a with respect to prime p is defined by the following formula.

$$(a \mid p) = a^{(p-1)/2} \bmod p$$

When integer a is not a multiple of p, then the Legendre symbol of integer a with respect to prime p is valued to either +1 or −1 depending on whether integer a is or is not a square modulo p.

The Legendre symbol of multiples of p with respect to prime p is null.

Let n be an odd positive integer, and let a be a positive integer. The Jacobi symbol of integer a with respect to integer n is the product of the Legendre symbols of integer a with respect to the prime factors of n.

Therefore if $n = p\,q$, then $(a \mid n) = (a \mid p)\,(a \mid q)$.

The Jacobi symbol of any integer a with respect to any integer n may be efficiently computed without the prime factors of n.

A.3 Key production

A.3.1 Public verification exponent

Each signing entity shall select a positive integer v as its public verification exponent.

The public verification exponent may be standardized in specific applications.

NOTE — Values 2 and 3 may have some practical advantages.

A.3.2 Secret prime factors and public modulus

Each signing entity shall secretly and randomly select two distinct odd primes p and q subject to the following conditions.

— If v is odd, then p–1 and q–1 shall be coprime to v.

— If v is even, then $(p-1)/2$ and $(q-1)/2$ shall be coprime to v. Moreover, p and q shall not be congruent to each other mod 8.

The public modulus n is the product of the secret prime factors p and q.

$$n = p\,q$$

The length of the modulus is k. Number k shall equal k_s+1.

NOTES

1 Some additional conditions on the choice of primes may well be taken into account in order to deter factorization of the modulus.

2 Some forms of the modulus simplify the modulo reduction and need less table storage. These forms are

$\mathrm{F}_{x,\,y,\,-}$: $n = 2^{64x} - c$ of length : $k = 64x$ bits,

$\mathrm{F}_{x,\,y,\,+}$: $n = 2^{64x} + c$ of length : $k = 64x+1$ bits,

where : $1 \le y \le 2x$ and $c < 2^{64x-8y} < 2c$.

In the negative forms, all the bits of the y most significant bytes are valued to one, up to a quarter of the length of the modulus.

In the positive forms, after a single most significant bit valued to one, all the bits of the y most significant bytes are valued to zero, up to a quarter of the length of the modulus.

A.3.3 Secret signature exponent

The secret signature exponent is the least positive integer s such that sv–1 is a multiple of

— $\mathrm{lcm}(p-1, q-1)$ if v is odd;

— $\frac{1}{2}\mathrm{lcm}(p-1, q-1)$ if v is even.

A.4 Signature function

The intermediate integer IR is a string of $k-1$ bits computed as described in 5.4.

The representative element of IR with respect to n is denoted by RR.

— If v is odd, then RR is IR.

— If v is even and if $(IR \mid n) = +1$, then RR is IR.

— If v is even and if $(IR \mid n) = -1$, then RR is $IR/2$.

NOTE — If v is even, then the Jacobi symbol of RR with respect to n is forced to +1.

RR shall be raised to the power s modulo n. The signature Σ is either the result or its complement to n, the least one.

$$\Sigma = \min \{ RR^s \bmod n, n-(RR^s \bmod n) \}$$

This defines the signature function "Sign".

$$\Sigma = \mathrm{Sign}(IR)$$

A.5 Verification function

The signature Σ is a positive integer less than $n/2$ which shall be raised to the power v modulo n for obtaining the resulting integer IS.

The recovered intermediate integer IR' is then defined by the following decoding.

— If IS is congruent to 6 mod 16, then IR' is IS.

— If $n-IS$ is congruent to 6 mod 16, then IR' is $n-IS$.

Moreover, when v is even,

— if IS is congruent to 3 mod 8, then IR' is $2IS$;

— if $n-IS$ is congruent to 3 mod 8, then IR' is $2(n-IS)$.

The signature Σ shall be rejected in all the other cases, and also if IR' does not lie in the range from 2^{k-2} to $2^{k-1}-1$.

This defines the verification function "Verif".

$$IR' = \mathrm{Verif}(\Sigma)$$

Annex B
(informative)

Illustrative examples related to annex A

The hexadecimal notation is used.

B.1 Examples with public exponent three

B.1.1 Key production

The public verification exponent *v* is 3.

Therefore the secret prime factors are both congruent to 2 mod 3.

p = BA09106C 754EB6FE BBC21479 9FF1B8DE
1B4CBB7A 7A782B15 7C1BC152 90A1A3AB

q = 1 6046EB39 E03BEAB6 21D03C08 B8AE6B66
CFF955B6 4B4F48B7 EE152A32 6BF8CB25

The public modulus *n* of 513 bits is of the form $2^{512} + c$, with $2c > 2^{384} > c$ (form $F_{x,\,y,\,+}$ with $x = 8$ and $y = 16$).

n = p q = 1 00000000 00000000 00000000 00000000
BBA2D15D BB303C8A 21C5EBBC BAE52B71
25087920 DD7CDF35 8EA119FD 66FB0640
12EC8CE6 92F0A0B8 E8321B04 1ACD40B7

The secret signature exponent *s* is $(n–p–q+3)/6$.

s = 2AAAAAAA AAAAAAAA AAAAAAAA AAAAAAAA
C9F0783A 49DD5F6C 5AF651F4 C9D0DC92
81C96A3F 16A85F95 72D7CC3F 2D0F25A9
DBF1149E 4CDC3227 3FAADD3F DA5DCDA7

B.1.2 Length of the variables

Number *z* is a positive integer less than or equal to *k*+2 divided by 16. Number *t* is the largest integer less than or equal to *k*+13 divided by 16.

Consequently, when number *k* is 513,

— number *z* is valued from 1 to 32, the messages to be signed are strings of 1 to 256 bits, and the padded messages *MP* and *MP'* are strings of 1 to 32 bytes;

— number *t* is 32, the extended messages *ME* are strings of 32 bytes, and the messages with redundancy *MR* and *MR'* are strings of 64 bytes.

Moreover, the intermediate integers *IR* and *IR'* and the signatures Σ are strings of 512 bits (*k*–1 bits).

B.1.3 Example 1

This example illustrates padding, extension and truncation for signing a message of 100 bits.

C BBAA 9988 7766 5544 3322 1100

Signature process

After padding four zeroes to the left, the padded message *MP* is a string of 13 bytes. Therefore *z*=13 and *r*=5.

MP = **0**C BBAA9988 77665544 33221100

The extended message *ME* results by repeating the 13 successive bytes of *MP*, in order and concatenated to the left, until obtaining a string of 32 bytes.

ME = 55443322 1100**0C**BB AA998877 66554433
221100**0C** BBAA9988 77665544 33221100

The extended message with redundancy *MR* is a string of 64 bytes obtained by interleaving the 32 bytes of *ME* and 32 bytes of redundancy. An alteration of the 26-th byte (E2) codes the message border.

MR = 44559944 88335522 3311EE00 E70C66BB
BBAADD99 0088FF77 22664455 99448833
55223311 EE00**E20C** 66BBBBAA DD990088
FF772266 44559944 88335522 3311EE00

The intermediate integer *IR* results from *MR* by truncating to 511 bits, by padding to the left one bit valued to 1 and by replacing the least significant byte: $\mu_2 \parallel \mu_1 = 00$ is replaced by $\mu_1 \parallel 6 = 06$.

Because *v* is odd, the representative element *RR* is *IR*.

RR = *IR* = **C**4559944 88335522 3311EE00 E70C66BB
BBAADD99 0088FF77 22664455 99448833
55223311 EE00E20C 66BBBBAA DD990088
FF772266 44559944 88335522 3311EE**06**

RR is raised to the power *s* modulo *n*. The signature Σ is here the complement to *n* of the result.

Σ = 309F873D 8DED8379 490F6097 EAAFDABC
137D3EBF D8F25AB5 F138D56A 719CDC52
6BDD022E A65DABAB 920A8101 3A85D092
E04D3E42 1CAAB717 C90D89EA 45A8D23A

Verification process

The signature Σ is less than $n/2$. The resulting integer IS is obtained by raising Σ to the power 3 modulo n.

```
IS =      3BAA66BB  77CCAADD  CCEE11FF  18F39944
          FFF7F3C4  BAA73D12  FF5FA767  21A0A33D
          CFE6460E  EF7BFD29  27E55E52  896205B7
          13756A80  4E9B0774  5FFEC5E1  E7BB52B1
```

The intermediate integers are strings of 512 bits where the most significant bit is valued to 1 and the least significant nibble is valued to 6. Because n is here congruent to 7 mod 16 and IS to 1 mod 16, the recovered intermediate integer IR' is $n–IS$.

```
IR' = n–IS =  C4559944  88335522  3311EE00  E70C66BB
              BBAADD99  0088FF77  22664455  99448833
              55223311  EE00E20C  66BBBBAA  DD990088
              FF772266  44559944  88335522  3311EE06
```

The recovered message with redundancy MR' is here the string of 64 bytes where a padded zero is followed by the 511 least significant bits of IR', except for the least significant byte; according to the permutation Π stating $\Pi(0)$ = E, EE06 denoted by $\mu_4 \parallel \mu_3 \parallel \mu_2 \parallel 6$ is replaced by $\mu_4 \parallel \mu_3 \parallel \Pi^{-1}(\mu_4) \parallel \mu_2$ valued to EE00.

```
MR' =     44559944  88335522  3311EE00  E70C66BB
          BBAADD99  0088FF77  22664455  99448833
          55223311  EE00E20C  66BBBBAA  DD990088
          FF772266  44559944  88335522  3311EE00
```

The first non-null sum is the 13-th sum valued to 5. Thus z=13 and r=5. The recovered padded message MP' is the string of the 13 bytes of MR' in the least significant odd positions.

```
MP' =           0C  BBAA9988  77665544  33221100
```

The four most significant bits (r–1=4) of MP' are null. The message itself is recovered as the string of the least significant 100 bits ($8z$+1–r=100) of MP'.

```
C  BBAA  9988  7766  5544  3322  1100
```

The signature is accepted because the 511 least significant bits of the recovered message with redundancy MR' are recovered in the extended message with redundancy computed from MP', exactly as MR from MP.

B.1.4 Example 2

This example illustrates a simpler case: a 256-bit message is neither padded nor extended with a 513-bit modulus.

```
FEDC  BA98  7654  3210  FEDC  BA98  7654  3210
FEDC  BA98  7654  3210  FEDC  BA98  7654  3210
```

Signature process

The message is a string of 256 bits, coded over exactly 32 bytes. Therefore z is 32 and r is 1. The message equals the padded message MP and the extended message ME.

```
ME = MP =  FEDCBA98  76543210  FEDCBA98  76543210
           FEDCBA98  76543210  FEDCBA98  76543210
```

The extended message with redundancy MR is a string of 64 bytes.

```
MR =      1DFEA7DC  6BBAD098  F2764954  85323E10
          1CFEA7DC  6BBAD098  F2764954  85323E10
          1CFEA7DC  6BBAD098  F2764954  85323E10
          1CFEA7DC  6BBAD098  F2764954  85323E10
```

The intermediate integer IR results from MR by truncating to 511 bits, by padding to the left one bit valued to 1 and by replacing the least significant byte.

Because v is odd, the representative element RR is IR.

```
RR = IR =  9DFEA7DC  6BBAD098  F2764954  85323E10
           1CFEA7DC  6BBAD098  F2764954  85323E10
           1CFEA7DC  6BBAD098  F2764954  85323E10
           1CFEA7DC  6BBAD098  F2764954  85323E06
```

RR is raised to the power s modulo n. The signature Σ is here the result.

```
Σ =       319BB9BE  CB49F3ED  1BCA26D0  FCF09B0B
          0A508E4D  0BD43B35  0F959B72  CD25B3AF
          47D608FD  CD248EAD  A74FBE19  990DBEB9
          BF0DA4B4  E1200243  A14E5CAB  3F7E610C
```

Verification process

The signature Σ is less than $n/2$. The resulting integer IS is obtained by raising Σ to the power 3 modulo n.

Because IS is here congruent to 1 mod 16, the recovered intermediate integer IR' is here IS.

```
IR' = IS =  9DFEA7DC  6BBAD098  F2764954  85323E10
            1CFEA7DC  6BBAD098  F2764954  85323E10
            1CFEA7DC  6BBAD098  F2764954  85323E10
            1CFEA7DC  6BBAD098  F2764954  85323E06
```

The recovered message with redundancy MR' is here the string of 64 bytes where a padded zero is followed by the 511 least significant bits of IR', except for the least significant byte; according to the permutation Π stating $\Pi(1)$ = 3, 3E06 denoted by $\mu_4 \parallel \mu_3 \parallel \mu_2 \parallel 6$ is replaced by $\mu_4 \parallel \mu_3 \parallel \Pi^{-1}(\mu_4) \parallel \mu_2$ valued to 3E10.

```
MR' =     1DFEA7DC  6BBAD098  F2764954  85323E10
          1CFEA7DC  6BBAD098  F2764954  85323E10
          1CFEA7DC  6BBAD098  F2764954  85323E10
          1CFEA7DC  6BBAD098  F2764954  85323E10
```

The first non-null sum is the 32-nd sum valued to 1. Thus z=32 and r=1. The recovered padded message MP' is the string of the 32 bytes of MR' in odd positions.

```
MP' =     FEDCBA98  76543210  FEDCBA98  76543210
          FEDCBA98  76543210  FEDCBA98  76543210
```

The recovered message is a string of 256 bits.

```
FEDC  BA98  7654  3210  FEDC  BA98  7654  3210
FEDC  BA98  7654  3210  FEDC  BA98  7654  3210
```

The signature is accepted because the 511 least significant bits of the recovered message with redundancy MR' are recovered in the extended message with redundancy computed from MP', exactly as MR from MP.