(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
03 September 2020 (03.09.2020)

WIPO | PCT

(10) International Publication Number
WO 2020/173662 A1

(51) International Patent Classification:
H04L 9/00 (2006.01)

(21) International Application Number:
PCT/EP2020/052495

(22) International Filing Date:
31 January 2020 (31.01.2020)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
19305233.9 27 February 2019 (27.02.2019) EP

(71) Applicant: **THALES DIS FRANCE SA** [FR/FR]; 6, rue de la Verrerie, 92190 Meudon (FR).

(72) Inventor: **JELJELI, Hamza**; THALES DIS FRANCE SA - C/O Intellectual Propertyy Department - 6, rue de la Verrerie, 92190 Meudon (FR).

(74) Agent: **BRICKS, Amélie**; 6, rue de la Verrerie, 92190 Meudon (FR).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

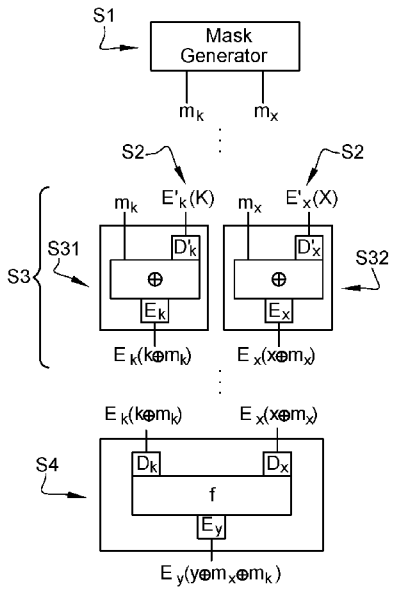
Declarations under Rule 4.17:

— *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))*

— *of inventorship (Rule 4.17(iv))*

(54) Title: METHOD SECURED AGAINST SIDE-CHANNEL ATTACKS WITH A NEW MASKING SCHEME PROTECTING LINEAR OPERATIONS OF A CRYPTOGRAPHIC ALGORITHM

Fig. 5



(57) **Abstract:** The present invention relates to a method secured against side channel attacks performing an operation (f) of a cryptographic algorithm on an input message in a whitebox implementation using a cryptographic key, said operation being linear with a Boolean exclusive OR (XOR) operation or an arithmetic operation (+), wherein : said method is performed by a client device comprising at least one hardware processor and a random generator and comprises : - generating (S1) a first mask and a second mask (m_k, m_x) by said random generator, - obtaining (S2) an encoded chunk of the cryptographic key ($E'_k(k)$) and an encoded chunk of the input message ($E'_x(x)$), - determining (S31) an encoded value ($E_k(k \oplus m_k)$) of the chunk of the cryptographic key masked with the first mask (m_k), by using at least one look up table mapping the obtained encoded chunk of the cryptographic key ($E'_k(k)$) and the first mask (m_k) to said encoded value of the masked chunk of the cryptographic key ($E_k(k \oplus m_k)$), - determining (S32) an encoded value ($E_x(x \oplus m_x)$) of the chunk of the input message (x) masked with the second mask (m_x), by using at least one look up table mapping the obtained encoded chunk of the input message ($E'_x(x)$) and the second mask (m_x) to said encoded value of the masked chunk of the input message ($E_x(x \oplus m_x)$), - determining (S4) an encoded value ($E_y(y \oplus m_x \oplus m_k)$) of the result (y) of performing said operation (f) on the chunk of the cryptographic key (k) and on the chunk of the input message (x), masked with the first and second masks, by using at least one look up table mapping the determined encoded value of the chunk of the cryptographic key masked with the first mask ($E_k(k \oplus m_k)$) and the determined encoded value of the chunk of the input message masked with the second mask ($E_x(x \oplus m_x)$) to said encoded value of the masked result of said operation ($E_y(y \oplus m_x \oplus m_k)$).

[Continued on next page]

WO 2020/173662 A1

Published:
— *with international search report (Art. 21(3))*

METHOD SECURED AGAINST SIDE-CHANNEL ATTACKS WITH A NEW MASKING SCHEME PROTECTING LINEAR OPERATIONS OF A CRYPTOGRAPHIC ALGORITHM

FIELD OF THE INVENTION

The present invention relates to the field of cryptographic methods and devices protected against side channel analysis, and more particularly to a new masking scheme protecting linear operations of a cryptographic algorithm against side-channel attacks.

BACKGROUND OF THE INVENTION

Cryptographic algorithms are commonly used for ensuring the privacy of communications by encryption, for authentication or for generating a verifiable signature. However, cryptographic processes are more and more deployed in applications executed on open devices. For example many user devices now have access to the Internet, such as PCs, tablets, smartphones, and can be used for playing copyrighted digital content such as audio or video files. Such files may be cryptographically protected, for example using digital rights management (DRM) files, in order to make it available to a user only as long as the user as subscribed to an online multimedia streaming service.

Cryptographic processes are then executed in an environment where not all users or devices can be trusted. Such a context is usually called a white-box attack context. In such a context, cryptosystems sensitive data such as private keys are vulnerable to attacks since the attacker has full access to the software implementation of the cryptographic processes. Binary code of such processes is completely accessible and editable by the attacker that can analyze the binary code of the cryptography application and, for example, memory pages or registers used for temporary storage during the execution. Such manipulations may, for example, be performed using debuggers and hardware emulation tools.

Software implementations of cryptosystems able to resist white-box attacks have been sought. In such implementations the private key may be provided on the subscriber's device, e.g., a mobile device such as a mobile telephone, in a manner that it can be used by the device to decrypt or generate a signature without revealing the private key. The key might be hidden in some way inside the code implementing the decryption/signature algorithm and the algorithm may be obfuscated so that it is very difficult to determine where the key is hidden. The key's value might be also protected so that it is very difficult to determine it. This is referred to as white-box cryptography.

Software implementations of cryptosystems able to resist white-box attacks may nevertheless be sensitive to side-channel attacks, performed by capturing reads and writes to memory or registers of the device performing the cryptographic operations.

A commonly used countermeasure against side-channel attacks is the masking of the intermediate values of the cryptographic operations with a random mask. In that case, an attacker performing a first-order side-channel attack at a given point of the encryption process would only get random values and would not obtain any information on the secret key used for the cryptographic process. Such a countermeasure may have an important impact on performances by causing memory and/or time overhead.

An example of such a masking scheme is given in the case of an operation f depicted on **Figure 1** having as inputs a chunk of a message x and a chunk of a cryptographic key k, and as output y such that y = f(x,k). As shown on **Figure 2**, all values are encoded by encodings E_x, E_k and E_y and masked by masks m_x, m_y and m_k. Before applying the function f, the masked inputs must be unmasked.

Look-Up Tables (LUT) are often used for implementing such calculations. When a LUT inputs and outputs are masked by random masks, the LUT must be adapted in order to include the unmasking of its inputs and the masking of its outputs, which is called randomizing the LUT. Unfortunately, such a randomizing operation is specific to the masks used for the LUT inputs and outputs, and must be performed again each time a different mask is used.

As a result, there is a need for a masking scheme having a lower performance impact on cryptographic operations than usual random masking schemes, while still being resistant at least to first-order side channel attacks.

SUMMARY OF THE INVENTION

For this purpose and according to a first aspect, this invention therefore relates to a method secured against side channel attacks performing an operation of a cryptographic algorithm on an input message in a whitebox implementation using a cryptographic key, said operation being linear with a Boolean exclusive OR (XOR) operation or an arithmetic operation (+),

wherein :

said method is performed by a client device comprising at least one hardware processor and a random generator and comprises :

- generating a first mask and a second mask by said random generator,
- obtaining an encoded chunk of the cryptographic key and an encoded chunk of the input message,
- determining an encoded value of the chunk of the cryptographic key masked with the first mask, by using at least one look up table mapping the obtained encoded chunk of the cryptographic key and the first mask to said encoded value of the masked chunk of the cryptographic key,
- determining an encoded value of the chunk of the input message masked with the second mask, by using at least one look up table mapping the obtained encoded chunk of the input message and the second mask to said encoded value of the masked chunk of the input message,
- determining an encoded value of the result of performing said operation on the chunk of the cryptographic key and on the chunk of the input message, masked with the first and second masks, by using at least one look up table mapping the determined encoded value of the chunk of the cryptographic key masked with the first mask and the determined encoded value of the chunk of the input message masked with the second mask to said encoded value of the masked result of said operation,

said masks being boolean masks if said operation is linear with a Boolean exclusive OR (XOR) operation and arithmetic masks if said operation is linear with an arithmetic operation (+).

Such a method enables to perform the operation f using masked inputs and outputs without requiring any randomization of a LUT corresponding to the operation f and without requiring the generation of a dedicated output mask for masking the output of the calculation.

In an embodiment, said step of masks generation may be correlated with the step of determining an encoded value of the result of performing said operation on the chunk of the cryptographic key and on the chunk of the input message, masked with the first and second masks.

This enables the method to prevent an attacker from setting the masks to constant values.

Said random generator may be configured to generate first mask and second mask under an encoded state, and said encoded first and second masks may be used instead of the unencoded first and second masks in the steps of said method.

It enables to hide the value of the masks during the whole computation and therefore enables the method to resist second order side channel analysis attacks.

The cryptographic algorithm may perform an operation among a linear part of a hash function, of a block cipher and of a stream cipher.

The hash function may be among MD5, SHA1, SHA2, SHA256, SHA512, Skein functions, the block cipher may be among AES, XTEA, FEAL, SPECK, Threefish functions and/or the stream cipher may be among Salsa20, ChaCha functions.

According to a second aspect, this invention therefore relates also to a computer program product directly loadable into the memory of at least one computer,

comprising software code instructions for performing the steps of the methods according to the first aspect when said product is run on the computer.

According to a third aspect, this invention therefore relates also to a non-transitory computer readable medium storing executable computer code that when executed by a client device comprising at least one hardware processor performs the method according to the first aspect.

According to a fourth aspect, this invention therefore relates also to a client device comprising :

- at least one hardware processor and a random generator configured to perform the method according to the first aspect,
- at least one memory for storing the encoded values and the results of the calculations performed during the different computing steps.

Said random generator may be configured to generate the first mask and second mask under an encoded state.

BRIEF DESCRIPTION OF THE DRAWINGS

The following description and the annexed drawings set forth in detail certain illustrative aspects and are indicative of but a few of the various ways in which the principles of the embodiments may be employed. Other advantages and novel features will become apparent from the following detailed description when considered in conjunction with the drawings and the disclosed embodiments are intended to include all such aspects and their equivalents.

- Figure 1 illustrates schematically performing an operation f of a cryptographic algorithm;
- Figure 2 illustrates schematically a method performing an operation f of a cryptographic algorithm with a usual masking scheme;
- Figure 3 is a schematic illustration of a system according to an embodiment of the present invention;

- Figure 4 is a schematic illustration of a client device according to an embodiment of the present invention;
- Figure 5 illustrates schematically a method according the present invention;
- Figure 6 illustrates schematically a method according to an embodiment of the present invention.

DETAILED DESCRIPTION OF EMBODIMENTS OF THE INVENTION

In the description detailed below, reference is made to the accompanying drawings that show, by way of illustration, specific embodiments in which the invention may be practiced. These embodiments are described in sufficient detail to enable those skilled in the art to practice the invention. It is to be understood that the various embodiments of the invention, although different, are not necessarily mutually exclusive. For example, a particular feature, structure, or characteristic described herein in connection with one embodiment may be implemented within other embodiments without departing from the spirit and scope of the invention. In addition, it is to be understood that the location or arrangement of individual elements within each disclosed embodiment may be modified without departing from the spirit and scope of the invention. The description detailed below is, therefore, not to be taken in a limiting sense, and the scope of the present invention is defined only by the appended claims, appropriately interpreted, along with the full range of equivalents to which the claims are entitled.

The invention provides a method performing an operation f of a cryptographic algorithm which, as described on **Figure 1**, applies to a chunk of an input message x and uses a chunk of a cryptographic key k. The operation f is implemented using a lookup-table (LUT) compatible with encoded inputs and generating an encoded output. The LUT performs the decoding of the encoded inputs $E_k(k)$ and $E_x(x)$, computes y the result of the function f and gives an encoded output $E_y(y)$; E_k, E_x and E_y being random bijective mappings. This LUT-based implementation of the operation f is not resistant to side-channel attacks of any-order.

The aim of the invention is to make this method secured against side channel attacks while consuming less resources than a usual masking scheme such as the one described on **Figure 2**.

As depicted on **Figure 3**, this method, and possibly the other steps of the cryptographic algorithm, is performed by a client device 101. Such a client device 101 may for example be a personal computer, or a smartphone. The client device 101 may be connected to at least one distant server 102 through a network 103.

As depicted on **Figure 4**, the client device 101 may include at least one hardware processor 201, connected via a bus 202 to a computer readable memory circuit including a random access memory (RAM) 203, a read-only memory (ROM) 204, and/or a non-volatile memory (NVM) 205. The client device 101 may also include a random number generator (RNG) 206 (called also “mask generator”), which is implemented as a software RNG run by the hardware processor. The client device 101 may further include an interface 207. Such an interface may be either a wired interface such as a USB, Ethernet or Thunderbolt interface, or a wireless interface, such as a Wifi or Bluetooth interface. The interface 207 may be used to connect the client device 101 to the network 103 which may be a wired network such as an Ethernet network or to a wireless network, e.g., wide-area networks, WiFi networks, or mobile telephony networks, through which communication may be performed with the server 102.

The following paragraphs describe, as depicted on **Figure 5**, the steps of a **method** secured against side channel attacks and performing an operation f of the cryptographic algorithm, performed by the client device 101 according to a first aspect of the invention.

The method according to the invention is designed to be used when the operation to be performed is linear with a Boolean exclusive OR (XOR) operation or an arithmetic operation, such as an addition or a subtraction modulo 2^i with i an integer. It may be used for performing an operation among a linear part of a hash function (MD5, SHA1, SHA2, SHA256, SHA512, Skein functions…) of a block cipher (XTEA, FEAL, SPECK, Threefish functions…) and of a stream cipher (Salsa20, ChaCha functions…). More

precisely, this operation may for example be part of an algorithm comprising only linear operations such as the hash functions SHA1/SHA2 or SHA1/SHA2-based HMAC algorithms. It may also be included in a linear part of an algorithm also having non-linear parts such as AES or DES cryptographic schemes.

In order to secure the operation f against side channel attacks while consuming less resources than a usual masking scheme, the main idea of the invention is to change the way the input values of the operation f are masked.

The values handheld by a cryptographic algorithm are encoded using bijective mappings in order to prevent an attacker from reading such values in the client device memory. Usually, when masking is used as a countermeasure to side channel attacks, as described on Figure 2, such values are masked by combining an already encoded value with a mask. As a result, such a masking must be removed before performing a subsequent operation, as described here above.

On the contrary, the method according to the invention combines the decoded chunk of the input message x and the chunk of the cryptographic key k with a mask before encoding the result of the combination. By doing so, the encoded inputs to the linear operation f may be decoded without unmasking, and then the operation f itself may be performed without unmasking such inputs either, thanks to the linearity of the operation f with the XOR or arithmetic operation used for the masking.

More precisely, during a first step S1, the random generator of the client device generates a first mask m_k and a second mask m_x. These masks are boolean masks if the operation f is linear with a Boolean exclusive OR (XOR) operation and are arithmetic masks if the operation f is linear with an arithmetic operation. In the following paragraphs, the symbol $\oplus$ represents both XOR in the case of Boolean masks and any such arithmetic operation in the case of arithmetic masks.

During a second step S2, the client device obtains an encoded chunk of the cryptographic key $E'_k(k)$ and an encoded chunk of the input message $E'_x(x)$. Such a

chunk may be any part of the whole key or input message, including the whole key or input message itself.

The server 102 may have generated randomly the bijective mappings which are to be used for encodings, and the cryptographic key, and it may have provided it to the client device. The client device may derive the encoded chunks of the key and the message by itself from the whole key and the whole input message, or it may read such encoded chunks in one of its memories or receive it through its interface 207.

During a third step S3, the client device applies the generated mask to the obtained chunks by using lookup tables.

More precisely it determines during a first masking step S31 an encoded value $E_k(k \oplus m_k)$ of the chunk of the cryptographic key k masked with the first mask m_k, by using at least one key masking look up table. The key masking look up tables map the obtained encoded chunk of the cryptographic key $E'_k(k)$ and the first mask m_k to said encoded value of the masked chunk of the cryptographic key $E_k(k \oplus m_k)$. In order to do so, the key masking look-up tables are equivalent to, as illustrated on figure 5, decoding the encoded key $E'_k(k)$, then masking using the first mask m_k and finally encoding E_k.

The second step S2 and the first masking step S31 may be merged in one step and be implemented with a single instruction reading access to the look up tables with as read indexes the generated mask and the obtained chunk to obtain the encoded value of the masked chunk of the cryptographic key.

During a second masking step S32, the client device determines an encoded value $E_x(x \oplus m_x)$ of the chunk of the input message x masked with the second mask m_x, by using at least one input message masking look up table. The input message masking look up tables map the obtained encoded chunk of the input message $E'_x(x)$ and the second mask m_x to said encoded value of the masked chunk of the input message $E_x(x \oplus m_x)$.

In order to do so, the input message masking look-up tables are equivalent to, as illustrated on figure 5, decoding the encoded message $E'_x(x)$, then masking using the second mask m_x and finally encoding E_x.

In these two masking steps two or more operations among decoding, masking and encoding may be performed altogether by a single LUT. Preferentially decoding, masking and encoding are performed altogether by a single LUT to ensure the confidentiality of the encodings.

Finally, during a fourth step S4, the client device performs securely the computation y = f(x,k), protected against the side channel attacks by masking, using a Look up table.

More precisely, it determines an encoded value $E_y(y \oplus m_x \oplus m_k)$ of the result y of performing the operation f on the chunk of the cryptographic key k and on the chunk of the input message x, masked with the first and second masks, by using at least one operation look up table. The operation look up tables map the determined encoded value of the chunk of the cryptographic key masked with the first mask $E_k(k \oplus m_k)$ and the determined encoded value of the chunk of the input message masked with the second mask $E_x(x \oplus m_x)$ to said encoded value of the masked result of the operation $E_y(y \oplus m_x \oplus m_k)$.

In order to do so, as illustrated on figure 5, the operation look up tables are equivalent to decoding both the encoded value $E_k(k \oplus m_k)$ of the chunk of the cryptographic key masked with the first mask m_k and the encoded value $E_x(x \oplus m_x)$ of the chunk of the input message x masked with the second mask m_x, and then performing the operation f, and finally encoding the result E_y.

In this fourth step two or more operations among decoding, performing the operation f and encoding may be performed altogether by a single LUT.

Preferentially decoding, performing the operation f and encoding are performed altogether by a single LUT to ensure the confidentiality of the encodings.

Since the operation f is linear with the XOR or arithmetic operation used for the masking, it is possible to perform the operation f on the chunk of the cryptographic key and the chunk of the input message without unmasking it first. Consequently, the LUT, or combination of LUTs, used at the fourth step for the operation f can be the standard LUT representing the operation f. There is no need to randomize it for making it compatible with a specific masking of its inputs or for outputting a masked output, which makes the implementation faster compared to the commonly used masking scheme presented in figure 2. Such a standard LUT may be provided by the server 102 to the client device 101. It also requires generating less masks as the output mask of the operation f is the XOR of the 2 inputs masks rather than a specific third mask as shown on Figure 2.

The random generator may be configured to generate the first mask and second mask under an encoded state $E_m(m_k)$, $E_m(m_x)$. In such a case, the client device may use these encoded first and second masks instead of the unencoded first and second masks in the steps of the method described here above. This embodiment is illustrated on figure 6.

In this embodiment, during the first and second masking steps S31 and S32, the masking look up tables also decode the encoded first and second mask, before masking the chunk of the cryptographic key k and the chunk of the input message x.

Such an embodiment prevents, during the whole computation, the masks from being handheld as unencoded values, which masks makes the implementation resistant to second-order side channel attacks.

In another embodiment, said step of masks generation S1 is correlated with the fourth step S4 determining an encoded value $E_y(y \oplus m_x \oplus m_k)$ of the result y of performing the operation f on the chunk of the cryptographic key k and on the chunk of the input message x, masked with the first and second masks. Thus an attempt to modify the behavior of the random generation is not possible without compromising the cryptographic computation, which prevents any attempt by an attacker of forcing masking using a constant value. Such a correlation is for example detailed in the patent application EP17306665.5.

According to a second aspect, this invention therefore relates also to a computer program product directly loadable into the memory of at least one computer, comprising software code instructions for performing the steps of the methods according to the first aspect when said product is run on the computer.

According to a third aspect, this invention therefore relates also to a non-transitory computer readable medium storing executable computer code that when executed by a client device comprising at least one hardware processor performs the methods according to the first aspect.

According to a fourth aspect, this invention therefore relates also to a client device 101 comprising :

- at least one hardware processor 201 and a random generator 206 configured to perform the steps of the methods according to the first aspect,
- at least one memory for storing the encoded values and the results of the calculations performed during the different computing steps.

CLAIMS

1. A method secured against side channel attacks performing an operation (f) of a cryptographic algorithm on an input message (x) in a whitebox implementation using a cryptographic key (k),
said operation being linear with a Boolean exclusive OR (XOR) operation or an arithmetic operation (+),
wherein :
said method is performed by a client device (101) comprising at least one hardware processor (201) and a random generator (206) and comprises :
- generating (S1) a first mask and a second mask (m_k, m_x) by said random generator,
- obtaining (S2) an encoded chunk of the cryptographic key ($E'_k(k)$) and an encoded chunk of the input message ($E'_x(x)$),
- determining (S31) an encoded value ($E_k(k \oplus m_k)$) of the chunk of the cryptographic key masked with the first mask (m_k), by using at least one look up table mapping the obtained encoded chunk of the cryptographic key ($E'_k(k)$) and the first mask (m_k) to said encoded value of the masked chunk of the cryptographic key ($E_k(k \oplus m_k)$),
- determining (S32) an encoded value ($E_x(x \oplus m_x)$) of the chunk of the input message (x) masked with the second mask (m_x), by using at least one look up table mapping the obtained encoded chunk of the input message ($E'_x(x)$) and the second mask (m_x) to said encoded value of the masked chunk of the input message ($E_x(x \oplus m_x)$),
- determining (S4) an encoded value ($E_y(y \oplus m_x \oplus m_k)$) of the result (y) of performing said operation (f) on the chunk of the cryptographic key (k) and on the chunk of the input message (x), masked with the first and second masks, by using at least one look up table mapping the determined encoded value of the chunk of the cryptographic key masked with the first mask ($E_k(k \oplus m_k)$) and the determined encoded value of the chunk of the input message masked with the second mask ($E_x(x \oplus m_x)$) to said encoded value of the masked result of said operation ($E_y(y \oplus m_x \oplus m_k)$),

said masks being boolean masks if said operation is linear with a Boolean exclusive OR (XOR) operation and arithmetic masks if said operation is linear with an arithmetic operation (+).

2. The method of claim 1, wherein said step of masks generation (S1) is correlated with the step of determining (S4) an encoded value ($E_y(y \oplus m_x \oplus m_k)$) of the result (y) of performing said operation (f) on the chunk of the cryptographic key (k) and on the chunk of the input message (x), masked with the first and second masks.

3. The method of claim 1 or 2 wherein said random generator (206) is configured to generate first mask and second mask under an encoded state ($E_m(m_k)$, $E_m(m_x)$), and wherein said encoded first and second masks are used instead of the unencoded first and second masks in the steps of said method.

4. The method of claim 1, wherein the cryptographic algorithm performs an operation among a linear part of a hash function, of a block cipher and of a stream cipher.

5. The method of claim 4, wherein the hash function is among MD5, SHA1, SHA2, SHA256, SHA512, Skein functions, the block cipher is among AES, XTEA, FEAL, SPECK, Threefish functions and/or the stream cipher is among Salsa20, ChaCha functions.

6. A computer program product directly loadable into the memory of at least one computer, comprising software code instructions for performing the steps of any one of claims 1 to 5 when said product is run on the computer.

7. A non-transitory computer readable medium storing executable computer code that when executed by a client device (101) comprising at least one hardware processor (201) performs the steps of any one of claims 1 to 5.

8. A client device (101) comprising :

- at least one hardware processor (201) and a random generator (206) configured to perform the steps of any one of claims 1 to 5,

- at least one memory (203, 204, 205) for storing the encoded values and the results of the calculations performed during the different computing steps.

9. The client device of claim 8, wherein said random generator (206) is configured to generate the first mask and second mask under an encoded state.

Fig. 1

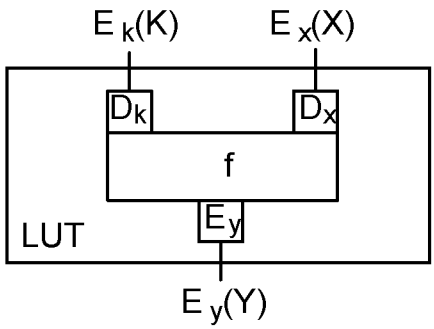
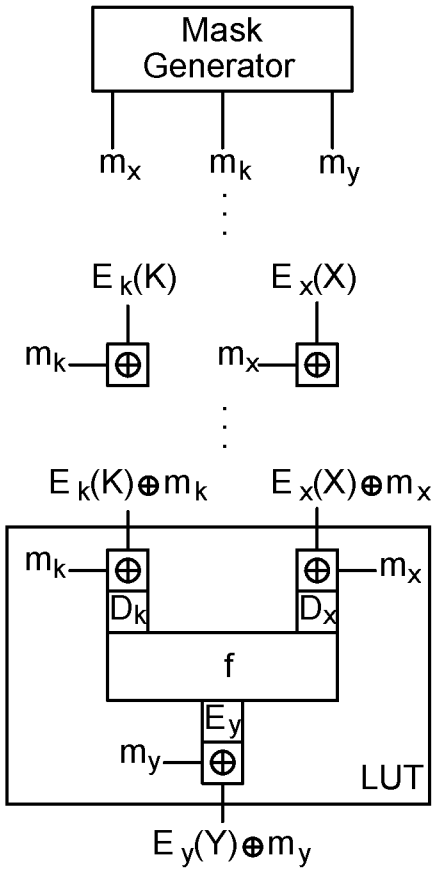

E k(K)
E x(X)
Dk
Dx
f
Ey
LUT
E y(Y)

Fig. 2


Mask Generator
mx
mk
my
E k(K)
E x(X)
mk
mx
E k(K)⊕mk
E x(X)⊕mx
mk
mx
Dk
Dx
f
Ey
my
LUT
E y(Y)⊕my

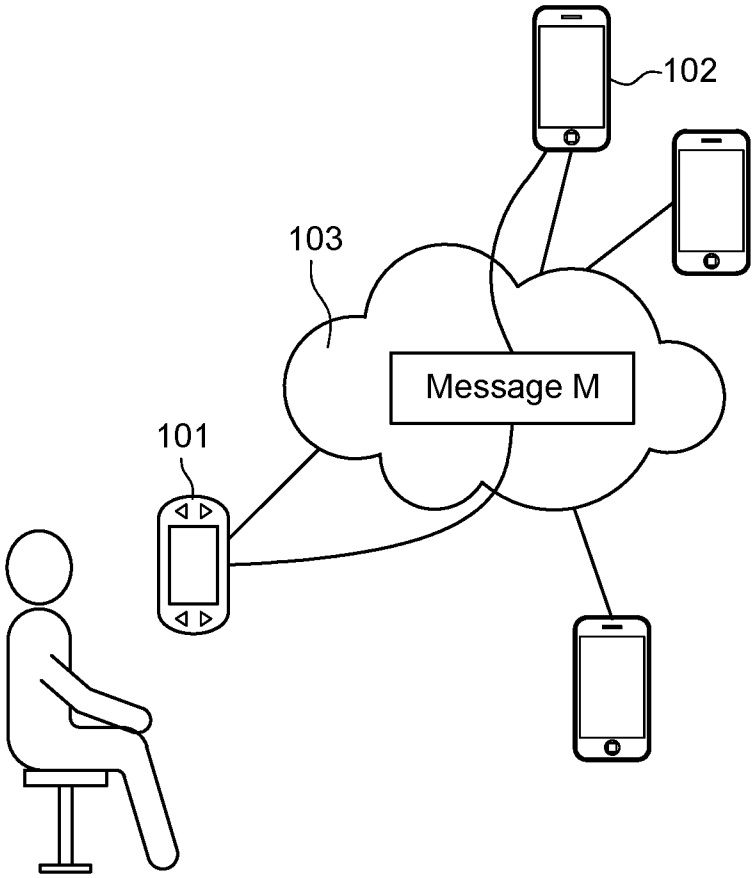

102
103
Message M
101

Fig. 3

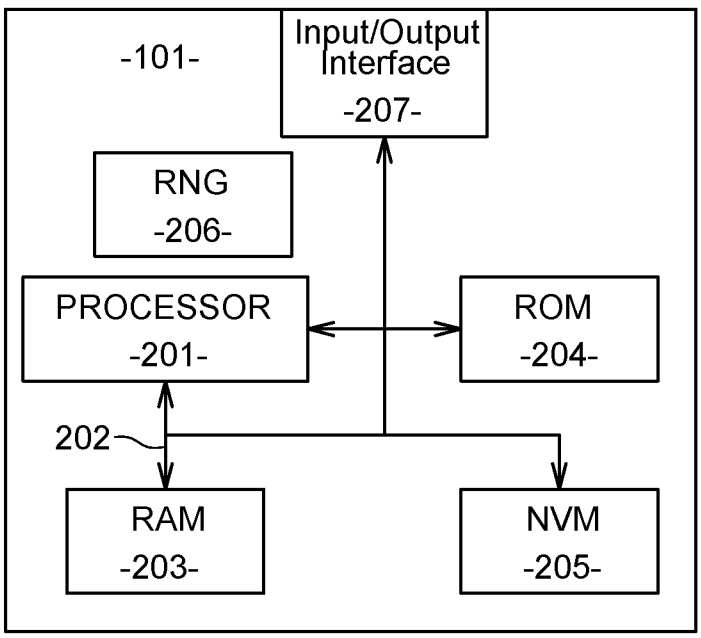

-101-
Input/Output Interface
-207-
RNG
-206-
PROCESSOR
-201-
ROM
-204-
202
RAM
-203-
NVM
-205-

Fig. 4

Fig. 5

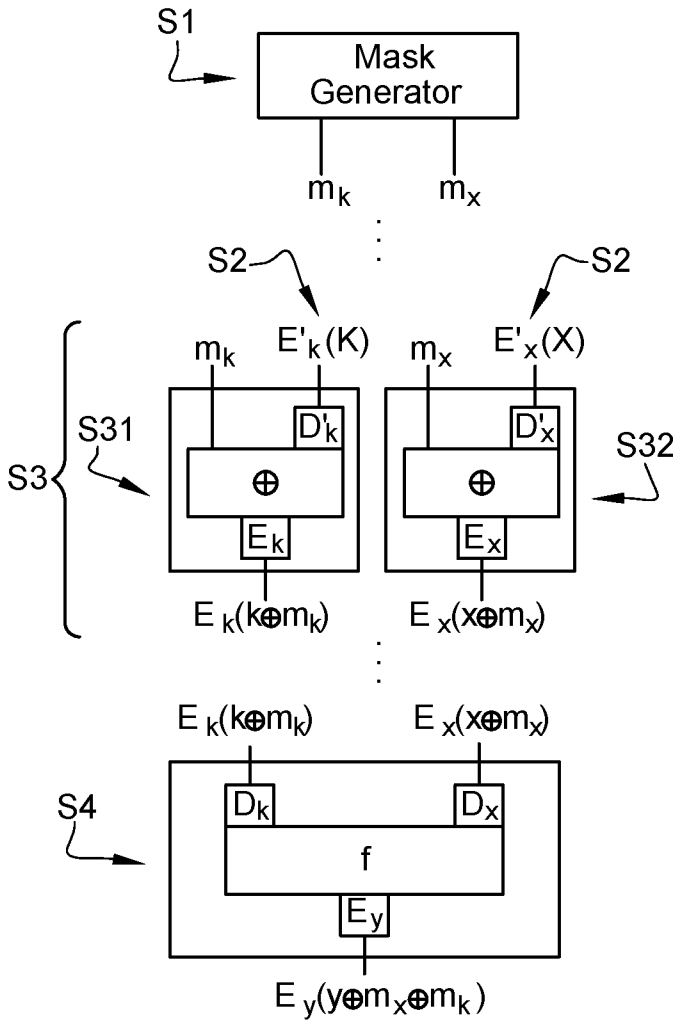
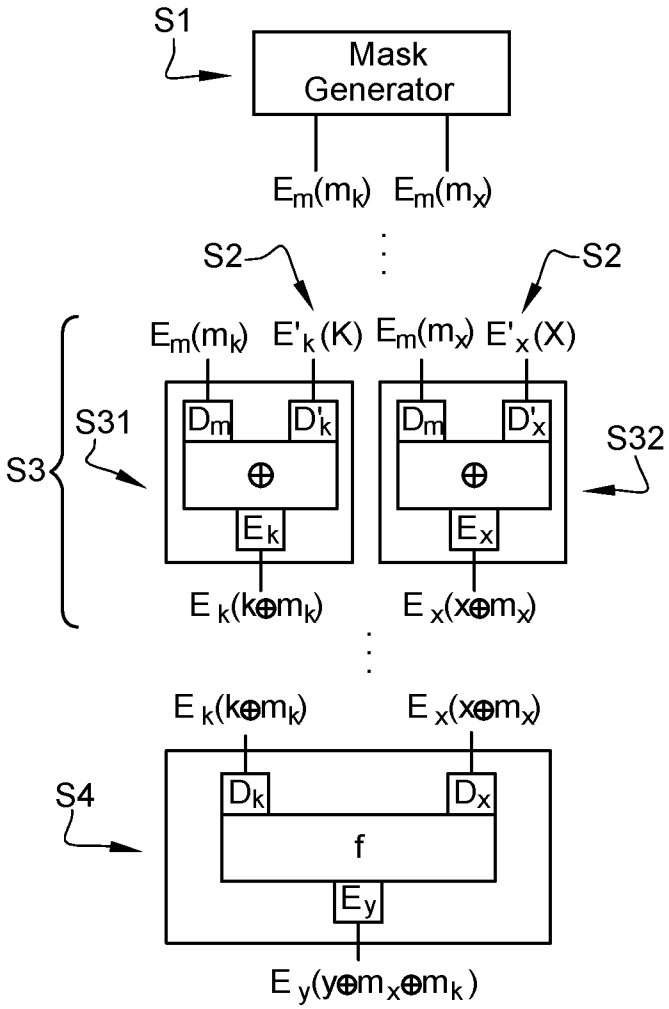

S1
Mask Generator
m_k
m_x
S2
S2
m_k
$E'_k(K)$
m_x
$E'_x(X)$
S3
S31
S32
D'_k
D'_x
E_k
E_x
$E_k(k \oplus m_k)$
$E_x(x \oplus m_x)$
$E_k(k \oplus m_k)$
$E_x(x \oplus m_x)$
S4
D_k
D_x
f
E_y
$E_y(y \oplus m_x \oplus m_k)$

Fig. 6


S1
Mask Generator
$E_m(m_k)$ $E_m(m_x)$
S2
S2
$E_m(m_k)$ $E'_k(K)$ $E_m(m_x)$ $E'_x(X)$
S3
S31
S32
D_m D'_k
D_m D'_x
⊕
⊕
E_k
E_x
$E_k(k \oplus m_k)$
$E_x(x \oplus m_x)$
$E_k(k \oplus m_k)$ $E_x(x \oplus m_x)$
S4
D_k D_x
f
E_y
$E_y(y \oplus m_x \oplus m_k)$

INTERNATIONAL SEARCH REPORT

International application No
PCT/EP2020/052495

A. CLASSIFICATION OF SUBJECT MATTER
INV. H04L9/00
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

EPO-Internal

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 2018/287779 A1 (LEE SEUNG KWANG [KR] ET AL) 4 October 2018 (2018-10-04) abstract paragraphs [0010] - [0024] paragraphs [0034] - [0081] ----- -/--	1-9

[X] Further documents are listed in the continuation of Box C.

[X] See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
13 February 2020	20/02/2020

Name and mailing address of the ISA/	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Di Felice, M

1

Form PCT/ISA/210 (second sheet) (April 2005)

INTERNATIONAL SEARCH REPORT

International application No
PCT/EP2020/052495

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	ITOH K ET AL: "DPA countermeasure based on the "masking method"", ELECTRONIC PUBLISHING, ARTISTIC IMAGING, AND DIGITAL TYPOGRAPHY; [LECTURE NOTES IN COMPUTER SCIENCE , ISSN 0302-9743], SPRINGER VERLAG, DE, vol. 2288, 1 December 2001 (2001-12-01), pages 440-456, XP002322028, ISBN: 978-3-540-24128-7 abstract Chapter I: "Introduction"; pages 440-441 Chapter 3: "Fixed-Value Masking Method"; pages 443-446	1-9
A	US 2017/195117 A1 (WU SHUANG [SG]) 6 July 2017 (2017-07-06) abstract paragraphs [0008] - [0034]	1-9

1

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/EP2020/052495

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2018287779 A1	04-10-2018	KR 20180110550 A	10-10-2018
		US 2018287779 A1	04-10-2018
US 2017195117 A1	06-07-2017	CN 106663387 A	10-05-2017
		EP 3186802 A1	05-07-2017
		SG 10201405852Q A	28-04-2016
		US 2017195117 A1	06-07-2017
		WO 2016043665 A1	24-03-2016