TITLE: Keyless Encryption Schemes with Addressable Physically Unclonable Elements

INVENTORS: Bertrand F. Cambou

CROSS REFERENCE TO RELATED APPLICATIONS AND PUBLICATIONS

This is the original provisional patent application. All references mentioned in this application are herein incorporated by reference without disclaimer.

FIELD OF THE INVENTION

The invention relates to cybersecurity and authentication systems. More particularly, the invention relates to applications utilizing physically unclonable functions (PUFs) in cryptographic protocols.

BACKGROUND OF THE INVENTION

The design of keyless encryption schemes has been an important area of research in cryptography for several years. Traditional cryptography uses keys to encrypt messages and keys to decrypt the ciphers. The key generation, key distribution, and storage of these keys is extremely complex and is a focus for the adversary. Additional attacks such as the ones based on differential power analysis attempt to extract the cryptographic keys during encryption and decryption cycles. Other attacks, such as the ones enabled by quantum computers, focus on key extractions.

Previous work has proposed keyless encryption schemes using Boolean operations such as exclusive OR to handle challenge-response data streams from physically unclonable functions (PUFs). Such methods tend to provide stream ciphers, which are not as safe as block ciphers, in which blocks of bits are encrypted together to increase their protection. The error rates due to the natural drifts of physical devices can limit the applicability of such methods. Other methods offer keyless schemes using more complex mathematical methods, such as the ones based on Galois Fields, but they do not use complementary hardware protection, which is not a solution easy to implement. Additionally, others offer partial keyless solutions limited to authentication, digital signature, or are still partially using keys. Therefore, the cryptographic solutions with real keyless schemes have limited impacts due to their complexity or offer lower levels of cyber-protection.

Physical unclonable functions (PUFs) act as the fingerprint of electronic components, exploiting the natural variations created during manufacturing. The design of the cryptographic schemes presented in this disclosure are based on addressable PUFs and PUFs based on arrays of cells with variable parametric values.

Addressable PUFs act as virtual bags of addressable keys. An example of architecture is shown in FIG. 1. The handshake allows the communicating parties to independently find the same addresses and generate the same cryptographic keys that are used to establish an encrypted communication channel. The server has access to a look-up table storing an exact image of the

PUF. The handshake can be a random number, generating message digests through hash functions and multi-factor authentication. The handshake can be replaced by remote tokens independently generating the same random numbers for both servers and client devices.

SUMMARY OF THE INVENTION

[ADD SUMMARY OF THE CLAIMS HERE]

BRIEF DESCRIPTION OF THE DRAWINGS

The drawings described herein constitute part of this specification and includes exemplary embodiments of the present invention which may be embodied in various forms. It is to be understood that in some instances, various aspects of the invention may be shown exaggerated or enlarged to facilitate an understanding of the invention. Therefore, drawings may not be to scale.

FIG. 1 depicts encryption schemes with key generation from addressable PUFs.

FIG. 2 depicts keyless encryption schemes with addressable PUFs, according to one embodiment.

FIG. 3 depicts an enrollment of addressable PUFs.

FIG. 4 depicts a generation of the addresses and orders from the message digests.

FIG. 5 depicts a keyless encryption scheme after handshake.

FIG. 6 depicts a keyless encryption scheme.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

The present invention described herein relates to novel encryption schemes which do not require cryptographic keys for authentication application.

General Architecture

The generic architecture allowing keyless encryptions with addressable PUFs is shown in FIG. 2. During the enrollment cycles that occur upfront in a secure environment (described in more detail below), an image of the PUF of the client device is downloaded in a look-up table of the encrypting device of the server.

The server generates a random number, T_w, which allows the independent generation of message digests by both the server and the client device with hash function, and multi-factor authentication. As described further below, a set of N+1 addresses ($a_{w,0}$, ..., $a_{w,N}$) pointing at the addressable PUF are generated from the message digest, as well as a set of N+1 natural numbers ($b_{w,0}$, ..., $b_{w,N}$) describing the order in which the blocks of the cipher will be ordered.

Using the randomly selected set of N cells ($a_{w,0}$, ..., $a_{w,N}$) and the set of N orders ($b_{w,0}$, ..., $b_{w,N}$), the encrypting device of either communicating parties can encrypt a message into a block cipher, as described further on. The only device able to decrypt the block cipher is the second encrypting device.

A new random number, T'_w, pointing to a new set of addresses with different orders can configure on demand the encryption engines to encrypt/decrypt new messages. The length N+1 of the set of addresses and the set of orders can be adjusted to allow the encryption of shorter or longer messages.

Enrollment of the Client Devices

As done in PUF-based crypto-systems, images of the constellation of the addressable PUFs of the client devices are downloaded in look-up tables of the server during secure enrollment cycles (FIG. 3). The initial readings of the PUFs (the challenges) are generated during this operation. To guarantee successful schemes, the initial readings need to be comprehensive. For example, each cell is characterized as much as 1,000 successive times under different electrical conditions with varying currents and voltages. In the example described below, the resistance of the cells of the memristor PUF arrays are measured 51 times at five different current levels: 10nA, 50nA, 100nA, 200nA, and 400nA. For each cell, the average value of the resistance at each current is stored in the look-up table.

After enrollment, at a given address, the server extracts the value stored in the look-up table, while the PUF is subject to a fresh measurement to extract a value called the response. In strong PUFs, which are used in this scheme, the responses match the values stored in the look-up table very well and their corresponding addresses. Therefore, the data stored in the look up table represents an accurate image of the data generated at each cycle by the PUFs.

Generation of Addresses and Block Orders

The hash functions convert the random number T_w into a message digest. In the experimental validation, the standard hash algorithm SHA-3 was used. Other algorithms work as well, such as MDA, SHA-1, SHA-2, and others. The message digest of SHA-3 are 512-bit long, as shown in FIG. 4.

The message digests are segmented in blocks of 16 bits:

$A^{1}_{0,w}\|A^{16}_{0,w}$	$A^{17}_{1,w}\|A^{32}_{1,w}$	...	...	$A^{496}_{31,w}\|A^{512}_{31,w}$

Each block is rotated to the left once to generate a second data stream:

$A^{2}_{32,w}\|A^{1}_{32,w}$	$A^{16}_{33,w}\|A^{17}_{33,w}$	...	...	$A^{497}_{34,w}\|A^{496}_{34,w}$

This is repeated 15 more times to generate 15 additional data streams:

$A^{16}_{480,w}\|A^{15}_{480,w}$	$A^{32}_{481,w}\|A^{31}_{481,w}$	...	...	$A^{512}_{511,w}\|A^{511}_{511,w}$

By putting together these 16 data streams, a stream of 512 blocks of 16 bits is formed, representing an 8,192-bit long stream:

$A^{1}_{0,w}\|A^{16}_{0,w}$	$A^{17}_{1,w}\|A^{32}_{1,w}$	...	...	$A^{512}_{511,w}\|A^{511}_{511,w}$

The 8,192-bit long stream is then segmented to generate a set of N+1=256 addresses ($a_{w,0}$, ..., $a_{w,N}$), and a set of N+1=256 orders ($b_{w,0}$, ..., $b_{w,N}$) and N+1=256. In this example, it is assumed that each address $a_{w,i}$ needs 20 bits to point at a particular cell of an array containing 1024x1024 cells and 8 bits to point at a particular order $b_{w,i}$ for this cell. Twenty-eight bits are therefore needed for each cell, and a total of 28x256=7,168 bits of the data stream are involved in the example.

In this method, it is possible that the address of the same cell is called more than once, which is not a problem for this scheme. It is also likely that the same order is called more than once, while some positions will be missing. The orders ($b_{w,0}$, ..., $b_{w,N}$) needs to be re-ordered into a new set (o_0, ..., o_N) without collisions in such a way that all positions 0 to N shall be represented wherein:

1) The orders ($b_{w,0}$, ..., $b_{w,N}$) , all natural numbers, are ranked from the lowest order to the highest order;
2) The lowest order $b_{w,v}$ corresponds to a re-order o_v, equal 0;
3) The next order $b_{w,x}$ corresponds to a re-order o_x, equal 1;
4) If several orders correspond to the same natural number, the first one on the data stream extracted from the message digest takes the lowest possible re-ordered number, the second one takes the next possible number.

For example, see below, if N=8, and the set of orders $b_{w,i}$ is re-ordered in the set of N+1 orders O_i with $i \in \{0, 8\}$:

i	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
$\mathbf{b_{w,i}}$	5	3	8	4	5	6	8	5	7
$\mathbf{o_i}$	2	0	7	1	3	5	8	4	6

At this point of the scheme, N+1 addresses ($a_{w,0}$, ..., $a_{w,N}$), and N+1 orders ($b_{w,0}$, ..., $b_{w,N}$) that are re-ordered as (o_0, ..., o_N). The sets are independently generated by the server, and the client device, are transmitted to their respective encrypting module. (NOTE: The examples presented in this section have the objective to clarify the method, not to limit the size of the streams used in the embodiment of the invention).

Encryption Schemes

The message to encrypt is a stream of 4 X N bits that are fragmented into blocks of 4 bits:
$M=(m_{1,0}\, m_{1,1}\, m_{1,2}\, m_{1,3})...(m_{i,0}\, m_{i,1}\, m_{i,2}\, m_{i,3})...(m_{N,0}\, m_{N,1}\, m_{N,2}\, m_{N,3})$ Equation 1

Each block i can be read as a natural number $Q_i \in \{0, 15\}$ to generate a stream of numbers:

$\{Q_1,...Q_i,...Q_N\}$ with $i \in \{1, N\}$ and $Q_i \in \{0, 15\}$ Equation 2

At each address $a_{w,i}$ the resistance $R_{w,N}$ is either read from the look-up table of the server, or measured from the PUF. From $R_{w,N}$, the value $C'_{w,N}$ is computed in the following way:

$C'_{w,0} = R_{w,0}(1 + 7.5K)$ $R_{w,0}$ is read at address $a_{w,0}$ Equation 3

$C'_{w,i} = R_{w,i}(1 + K\,Q_i)$ $R_{w,i}$ is read at address $a_{w,i}$ Equation 4

The resulting data stream is generated:

$C' = \{C'_{w,0}, C'_{w,1}, ..., C'_{w,i}, ..., C'_{w,N}\}$ Equation 5

The set of re-ordering numbers are needed to finalize the encryption and permute the N+1 C's.

$O = \{o_0, o_1, ..., o_i, ..., o_N\}$ Equation 6

$C = \{C_{w,0}, C_{w,1}, ..., C_{w,k}, ..., C_{w,N}\}$ Equation 7

with $C_{w,k} = C_{w,oi} = C'_{w,i}$ and $k \in \{0, N\}$

The block cipher has thereby been permuted by the orders generated by the message digests. The overall encryption scheme, which is shown in FIG. 5, can be protected by additional nonce and random numbers that enhance entropy, which make frequency analysis difficult. It must be noted that in this example of embodiment, the first cell and its resistance, $R_{w,0}$, is used to calibrate the PUF, not to encrypt part of the message. The reason that the value is multiplied by (1+7.5K), as shown in Equation 3, is to make the result as neutral as possible when compared with the other values. On average all other values are also multiplied by the same factor, 7.5 is the average value of Q_i, with values varying from 0 to 15. The parameter K can take arbitrary values typically from 0.2 to 2, as long as the communicating parties use the same number.

Example: a 32-bit long stream need to be encrypted, N=8:
M=(0110)(0101)(0100)(1011)(0001)(0110)(1001)(0111)

The values of the resistance, in M-ohm at the 9 cells selected by the set of addresses $a_{w,i}$ is:
1.7 2.5 2.9 2.1 1.2 2.2 1.9 2.6 2.0

The set of orders $b_{w,i}$ and their re-ordering oi is the one presented above. The encryption scheme of this example (with k=0.2) is shown below. The cipher C is converted into a digital stream and transmitted to the receiving party.

i	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
M_i		0110	0101	0100	1011	0001	0110	1001	0110
Q_i		6	5	4	11	1	6	9	7
$C'_0 = R_0(1+7.5K)$; $C'_i = R_i(1+KQ_i)$ K=0.2									
R_i	1.7	2.5	2.9	2.1	1.2	2.2	1.9	2.6	2.0
C'_i	4.25	5.5	5.8	3.78	3.84	2.64	4.18	7.28	4.8

$b_{w,i}$	5	3	8	4	5	6	8	5	7
o_i	2	0	7	1	3	5	8	4	6
C_i	5.5	3.78	4.25	3.84	7.28	2.64	4.8	5.8	4.18

Decryption Schemes

To decrypt the cipher, the receiving party has access to the set of addresses and the set of orders. The overall decryption scheme is shown in FIG. 6.

From the addresses, the set of resistances $R_{w,i}$ are generated from the PUF:

$$R= \{R_{w,0} , R_{w,1} ,..., R_{w,i} ,..., R_{w,N}\} \quad \text{Equation 8}$$

The set of orders $O = \{o_0 , o_1 ,..., o_i ,..., o_N\}$ allows the reordering the cipher in the right order and generate the following stream:

$$C' = \{C'_{w,0} , C'_{w,1} ,..., C'_{w,i} ,..., C'_{w,N}\} \quad \text{Equation 9}$$

$$R'_{w,0} = C'_{w,0} / (1 +7.5K) \text{ is compared with } R_{w,0} \text{ and read at the address } a_{w,0} \quad \text{Equation 10}$$

Ratio between $R'_{w,0}$ and $R_{w,0}$, which is read from the PUF, allows the calibration of all resistances and reduces the error rate between challenges and responses. The values of the natural number Q_i are generated from C':

$$Q_i = (C'_{w,i} - R_{w,i})/(KR_{w,i}) \text{ where } R_{w,i} \text{ is read at the address } a_{w,i} \text{ and } i \in \{1, N\} \quad \text{Equation 11}$$

The message is then retrieved by converting each number Q_i into a block of four bits.

Generalization

Small variations around the schemes presented in this disclosure were not presented to avoid confusion. Some important variations are:

1) Use of other handshake protocols. The essence of this invention is to use the independently generated message digest to drive the encrypting/decrypting modules. The handshake can be replaced by other methods to share random numbers. For example, tokens can have multiple parties having access to constantly changing random numbers;
2) Combine the resulting encryption schemes with methods using symmetrical or asymmetrical encrypting methods;
3) Use the PUFs to provide multi-factor authentication for access control;

The described features, advantages, and characteristics may be combined in any suitable manner in one or more embodiments. One skilled in the relevant art will recognize that the circuit may be practiced without one or more of the specific features or advantages of a particular embodiment. In other instances, additional features and advantages may be recognized in certain embodiments that may not be present in all embodiments.

Reference throughout this specification to “one embodiment,” “an embodiment,” or similar language means that a particular feature, structure, or characteristic described in connection with the embodiment is included in at least one embodiment. Thus appearances of the phrase “in one embodiment,” “in an embodiment,” and similar language throughout this specification may, but do not necessarily, all refer to the same embodiment.

CLAIMS

The invention claimed is:

1. A keyless encryption/decryption scheme designed with addressable PUFs, which leverage the value of the physical parameters of each cell:

1.1. The initial readings of the physical parameters of the addressable PUF of a client device, are stored as references by a server to form an image of the PUF;

1.2. The server and the client device independently generate through handshakes a set of addresses pointing at a set cells in the PUF, or the image of the PUFs;

1.3. The message to encrypt is segmented in blocks of streams of bits, each block generating a natural number representing the decimal representation of its stream of bits; the natural numbers are used to modify the physical parameters describing each cell, using a method know by both the client device, and the server. The resulting information generate a data stream of bits;

1.4. The decryption of the cipher follows the opposite track: the original values of the physical parameters of the PUFs are retrieved, which allows to retrieve the original message;

2. A keyless encryption/decryption scheme designed with addressable PUFs, which i) leverage the value of the physical parameters of each cell, and ii) re-order the data stream:

2.1. The initial readings of the physical parameters of the addressable PUF of a client device, are stored as references by a server to form an image of the PUF;

2.2. The server and the client device independently generate through handshakes: i) a set of addresses pointing at a set cells in the PUF, or the image of the PUFs; ii) a set of data streams to re-order the positions of the cells selected by the set of addresses in a different order; these data streams are called the set of orders;

2.3. The message to encrypt is segmented in blocks of streams of bits, each block generating a natural number representing the decimal representation of its stream of bits; i) the natural numbers are used to modify the value of the physical parameters describing each cell, using a method know by both the client device, and the server, ii) the resulting data streams of bits that are thereby generated are re-ordered in a cipher by using the set of orders generated by the handshakes as described in claim 3.2 ii). The cipher is communicated to the other party for decryption;

2.4. The decryption of the cipher follows the opposite track: the ciphers are re-ordered by using the same set of order, which is known by both parties; the original values of the physical parameters of the PUFs are retrieved, which allows to retrieve the original message;

3. A keyless encryption/decryption scheme designed with addressable PUFs, which i) leverage the value of the physical parameters of each cell, and ii) adjust the reading of each cell with an additional parameter:

3.1. The initial readings of the physical parameters of the addressable PUF of a client device, are stored as references by a server to form an image of the PUF;

3.2. The server and the client device independently generate through handshakes: i) a set of addresses pointing at a set cells in the PUF, or the image of the PUFs; ii) a set of data streams used to drive each cell differently and adjust independently the physical parameters;

3.3. The message to encrypt is segmented in blocks of streams of bits, each block generating a natural number representing the decimal representation of its stream of bits; i) the natural numbers are used to modify the value of the physical parameters describing each cell, using a method know by both the client device, and the server, ii) the values of the physical parameter are adjusted independently based on the set of data generated by the handshakes as described in claim 3.1 ii). The cipher is communicated to the other party for decryption;

3.4. The decryption of the cipher follows the opposite track: the original values of the physical parameters of the PUFs are retrieved, and adjusted, which allows to retrieve the original message;

4. A keyless encryption/decryption scheme designed with addressable PUFs, which i) leverage the value of the physical parameters of each cell, ii) re-order the data stream; iii) adjust the reading of each cell with an additional parameter: this scheme combines the methods presented claims 1 to 3;

5. Wherein the handshakes of claims 1 to 4 use hash functions, and multi-factor authentication to protect the schemes;

6. Wherein the hashing functions of claim 1 to 5 are based on SH-1, SHA-2, SHA-3, SHA-128, SHA-256, SHA-384, SHA-512, or other known algorithms;

7. Wherein the PUFs of claims 1 to 4 are designed with memristors, resistive RAMs, phase change memories, conductive bridge RAMs, magnetic RAM, STT RAM, or ferro-electric RAMs;

8. Wherein the parameter describing the PUFs of claim 1 to 4 are the resistances of the cells, the electric current circulating in the cells, the voltage applied to the cell, the capacitance of the cells, or the inductance of the cell;

9. Wherein error correcting schemes are used to enhance the quality of the PUFs of claims 1 to 4;

10. Wherein the PUFs of claims 1 to 4 use a ternary representation to identify the unstable cells, and reduce the error rates;

11. Wherein an additional set of values are used in the scheme of claims 1 to 4, such as not to be limited with the value of the current to inject in each cell to change the value of the parameter used to encrypt a message;

12. Wherein one, or several cells of the PUF of claims 1 to 4 are used to calibrate the cipher, such as, not to be limited with resistance of know values;

13. Wherein the method to modify the value of each cell of claim 3, use parameters that are adjustable, and part of the cryptographic scheme, such as, not to be limited with, a parameter than amplify the values measured by the PUFs;

14. Wherein the handshakes of claims 1 to 4 use tokens allowing the communicating parties to generate the same random numbers;

15. Wherein additional encryption scheme that use cryptographic keys are combined with the schemes of claims 1 to 4;

16. Wherein the PUFs of claims 1 to 4 also provide ways to authenticate the client device;

ABSTRACT

The encryption schemes presented herein do not need cryptographic keys. The encryption directly uses physical unclonable functions (PUFs) or an image of the PUFs stored in separate devices to generate ciphers that can only be decrypted by the same PUFs or their images stored in separate devices. Through a handshake cycle, sets of addresses are generated to point at a set of cells in the PUFs, defined by their physical parameters, which are used for a particular encryption/decryption cycle. The messages to encrypt, which are streams of bits, are segmented into block of bits; the numerical values of these blocks are used to modify the values of the physical parameters of the selected cells. These modified values are reorganized randomly to form block ciphers that are transmitted to the communicating party. Having independent access to the same values of the physical parameters of the same set of cells, the receiving party measures the differences between the original and the modified values of these parameters. The differences allow to receiving party to calculate back the values of the block of bits that were used to modify the values of the physical parameters, and thereby retrieve the original messages. The size of the block ciphers in these schemes are typically 1,024-bits long. Multiple handshakes can be used to encrypt longer messages segmented in blocks of various sizes. The use of memristor based PUFs enhance the entropy of the schemes, because the value of the parameters of each cell is constantly adjustable.

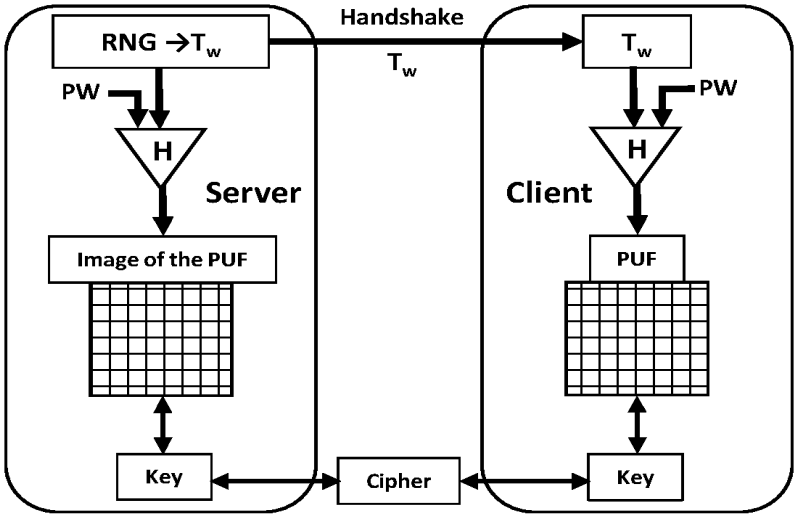
RNG →T_w
Handshake
T_w
T_w
PW
PW
H
H
Server
Client
Image of the PUF
PUF
Key
Cipher
Key

FIG. 1

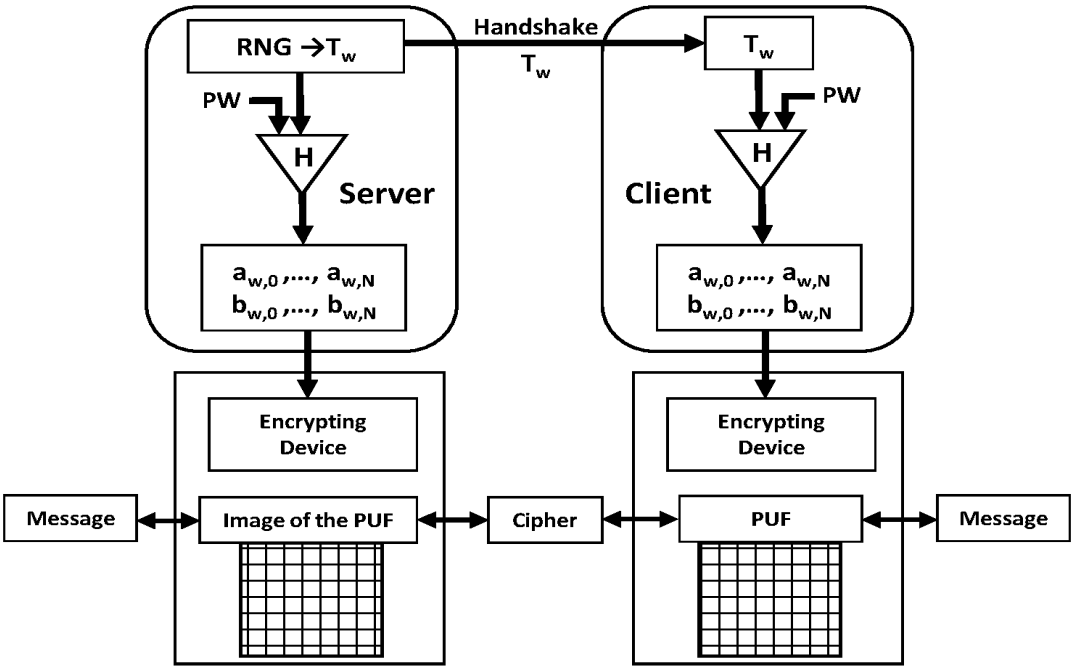
RNG →T_w
Handshake
T_w
T_w
PW
PW
H
H
Server
Client
$a_{w,0}$,…, $a_{w,N}$
$b_{w,0}$,…, $b_{w,N}$
$a_{w,0}$,…, $a_{w,N}$
$b_{w,0}$,…, $b_{w,N}$
Encrypting Device
Encrypting Device
Message
Image of the PUF
Cipher
PUF
Message

FIG. 2

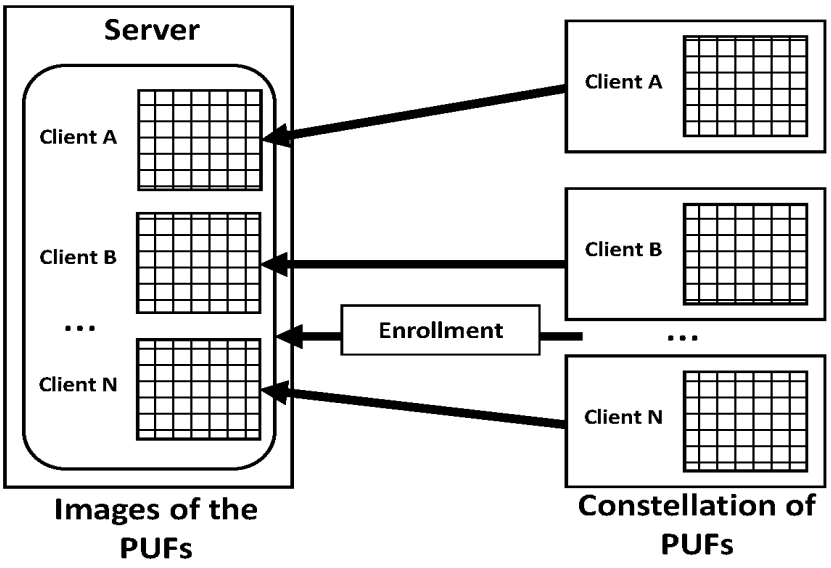
Server
Client A
Client B
...
Client N
Images of the PUFs
Enrollment
Client A
Client B
...
Client N
Constellation of PUFs

FIG. 3

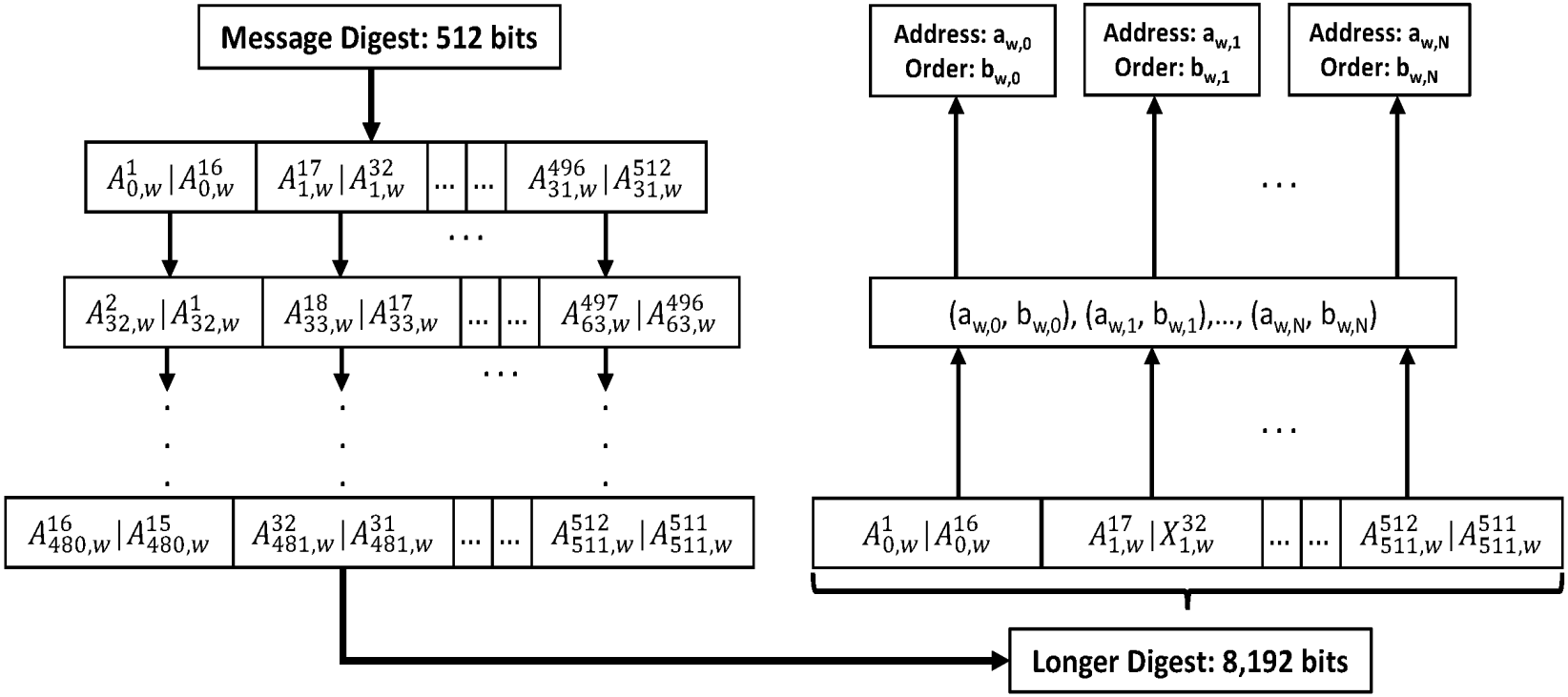
Message Digest: 512 bits
$A^{1}_{0,w} | A^{16}_{0,w}$
$A^{17}_{1,w} | A^{32}_{1,w}$
$A^{496}_{31,w} | A^{512}_{31,w}$
$A^{2}_{32,w} | A^{1}_{32,w}$
$A^{18}_{33,w} | A^{17}_{33,w}$
$A^{497}_{63,w} | A^{496}_{63,w}$
$A^{16}_{480,w} | A^{15}_{480,w}$
$A^{32}_{481,w} | A^{31}_{481,w}$
$A^{512}_{511,w} | A^{511}_{511,w}$
Address: $a_{w,0}$ Order: $b_{w,0}$
Address: $a_{w,1}$ Order: $b_{w,1}$
Address: $a_{w,N}$ Order: $b_{w,N}$
$(a_{w,0}, b_{w,0}), (a_{w,1}, b_{w,1}), \ldots, (a_{w,N}, b_{w,N})$
$A^{1}_{0,w} | A^{16}_{0,w}$
$A^{17}_{1,w} | X^{32}_{1,w}$
$A^{512}_{511,w} | A^{511}_{511,w}$
Longer Digest: 8,192 bits

FIG. 4

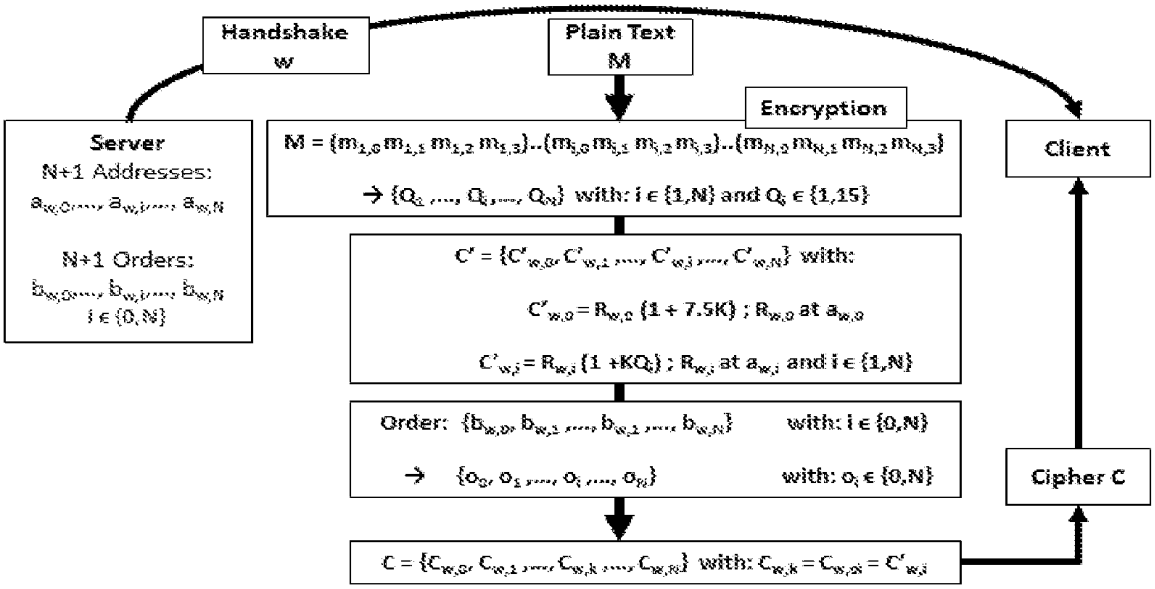
Handshake
w
Plain Text
M
Encryption
Server
N+1 Addresses:
$a_{w,0}$,..., $a_{w,i}$,..., $a_{w,N}$
N+1 Orders:
$b_{w,0}$,..., $b_{w,i}$,..., $b_{w,N}$
$i \in \{0,N\}$
$M = \{m_{1,0}\, m_{1,1}\, m_{1,2}\, m_{1,3}\}..\{m_{i,0}\, m_{i,1}\, m_{i,2}\, m_{i,3}\}..\{m_{N,0}\, m_{N,1}\, m_{N,2}\, m_{N,3}\}$
$\rightarrow \{Q_0, ..., Q_i, ..., Q_N\}$ with: $i \in \{1,N\}$ and $Q_i \in \{1,15\}$
$C' = \{C'_{w,0}, C'_{w,1}, ..., C'_{w,i}, ..., C'_{w,N}\}$ with:
$C'_{w,0} = R_{w,0}\{1 + 7.5K\}$; $R_{w,0}$ at $a_{w,0}$
$C'_{w,i} = R_{w,i}\{1 + KQ_i\}$; $R_{w,i}$ at $a_{w,i}$ and $i \in \{1,N\}$
Order: $\{b_{w,0}, b_{w,1}, ..., b_{w,i}, ..., b_{w,N}\}$ with: $i \in \{0,N\}$
$\rightarrow \{o_0, o_1, ..., o_i, ..., o_N\}$ with: $o_i \in \{0,N\}$
$C = \{C_{w,0}, C_{w,1}, ..., C_{w,k}, ..., C_{w,N}\}$ with: $C_{w,k} = C_{w,oi} = C'_{w,i}$
Client
Cipher C

FIG. 5

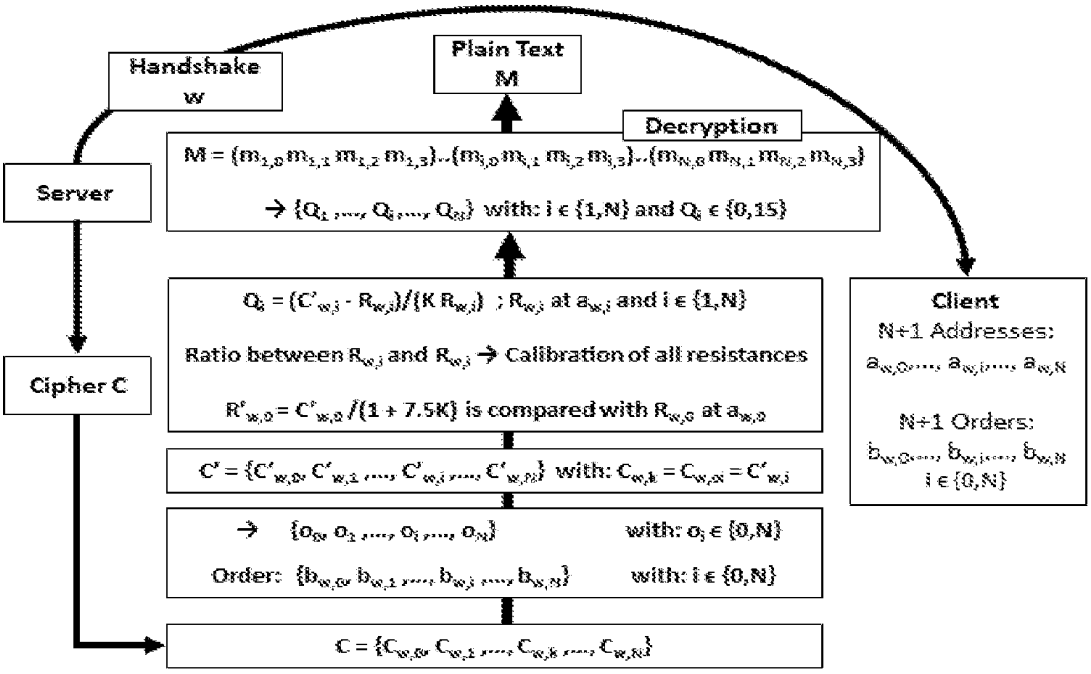
Handshake
w
Plain Text
M
Decryption
Server
M = {m1,0 m1,1 m1,2 m1,3}..{mi,0 mi,1 mi,2 mi,3}..{mN,0 mN,1 mN,2 mN,3}
→ {Q1 ,..., Qi ,..., QN} with: i ϵ {1,N} and Qi ϵ {0,15}
Qi = (C'w,i - Rw,i)/(K Rw,i) ; Rw,i at aw,i and i ϵ {1,N}
Ratio between Rw,i and Rw,i → Calibration of all resistances
R'w,0 = C'w,0 /(1 + 7.5K) is compared with Rw,0 at aw,0
Client
N+1 Addresses:
aw,0,..., aw,i,..., aw,N
N+1 Orders:
bw,0,..., bw,i,..., bw,N
i ϵ {0,N}
Cipher C
C' = {C'w,0, C'w,1 ,..., C'w,i ,..., C'w,N} with: Cw,k = Cw,oi = C'w,i
→ {o0, o1 ,..., oi ,..., oN} with: oi ϵ {0,N}
Order: {bw,0, bw,1 ,..., bw,i ,..., bw,N} with: i ϵ {0,N}
C = {Cw,0, Cw,1 ,..., Cw,k ,..., Cw,N}

FIG. 6