(19)

(11) Publication number: IL 122106 B
(43) Publication date: 30.11.2010
(51) Int. Cl: **H04L 09//00;**

(12) PUBLISHED NATIONAL APPLICATION

(21) Application number: 122106
(22) Date of filing: 04.11.1997

(71) Applicant: ENCOTONE LTD. 6 HAHOSHLIM ST. HERZLIA PITUACH 46722 P.O BOX 12006 IL ISAAC J. LABATON 955 HAR ARGAMAN MACABIM 71908 IL

(72) Inventor: JMB, FACTOR & CO. LTD. HAR HOTZVIM HI-TECH PARK, P.O.B. 45087, JERUSALEM 91450 IL 02-5714777 02-5711259

(54) METHOD AND ALGORITHMS FOR IDENTIFICATION AND VALIDATION
(54) שיטות ואלגוריתמים עבור זיהוי ואישור

שיטות ואלגוריתמים עבור זיהוי ואישור

METHOD AND ALGORITHMS FOR IDENTIFICATION AND VALIDATION

Method and algorithms for identification and validation

Technical Field

The present invention-relates, generally., to methods and mathematical algorithms for identification, including remote identification and validation of messages. The methods and mathematical algorithms for identification of this invention make possible the validation of the identification without the necessity of conventional authentication procedures and without the need of databases in the reception side, without compromising security levels.

Background art and technical problems.

Service providers (Social Securities, Telecoms (PTTs), financial institutions like brokers, banks, merchants, etc, referred in the following as Identification Nodes or ID Nodes.) willing to provide remote services through the phone, internet, or any other media are in need of secure identification of an individual, an entity or an account which asks for the respective service.

The common way to identify remotely a person , an entity or an account (in the following : the Presumed Caller) with some security, against the ID Node is by adding an additional step to the identification, which is generally referred to as authentication.

Authentication consists of providing the ID Node , an additional piece of information, usually a secret code alongside with the identification information , which is used to corroborate that the identification is accurate and that the Remote Caller is not impersonating somebody else , in other words, that the Remote Caller is the Presumed Caller.

The additional piece of information is usually a secret code or a password, but also can be a Dynamic Code, computed by means of software, or by a token in the hands of the Remote Caller.

It is clear that any constant and non-variable piece of information does not add much security to the identification process because this said constant piece of information can become easily known to a third party (potential attackers/ potential impostors/ eavesdroppers) and , therefore, the Presumed Caller can be impersonated effortlessly.

By the contrary, authenticating by means of variable piece of information (in the following: Dynamic or One Time Code) is more secure.

All of the known ways of authentication by using a Dynamic or One Time Code need necessarily the (usually prior) step of identification of the Remote Caller against the ID Node , being the identification made by means of a name (like a login name), a serial number, an additional fix code, etc. This constant part of the message will be referred as the Identification Message. Thus, the Secure –Identification- By-Means-Of-Authentication procedure consists of the following steps:

- Identification step : identify who the Remote Caller is supposed to be, by means of receiving a constant (non variable or at least non constantly variable) piece of information referred to as the Identification Message. This step will be referred to as the Pre-Authentication Identification.
- second :Use of Database by the Identification Node ; look into a database, for the Presumed Caller's secret information or computing keys, in order to compute the dynamic piece of information , (referred to as the Valid Dynamic Code) which is expected from such Presumed Caller at such particular time, or at this event, if the scheme is sequential.

- third: Proper Authentication ; compares this Valid Dynamic Code (computed at the Identification node) with the Dynamic Code received (referred to as the Received Dynamic Code) in order to check if both codes match, and eventually, if so, the Identification Node corroborates the identification of the Remote Caller as the Presumed Caller.

A variation of this authentication method is referred to as the Challenge and Response method. Correspondingly, in this case also, the Remote Caller identifies himself by a constant Identification Message. This step correspond the one referred to, above , as the Pre-Authentication Identification.

- The Remote caller receives a Challenge generated and sent by the Identification node, computes a response, and this response plays the role of the said Dynamic Code. As before, the Identification node, after identifying the Remote Caller (the Pre-Authentication Identification), needs a data base in order to find out what is expected as a response of such challenge, for such particular Remote Caller at such moment.

The general characteristic of all these schemes is the need of databases at the Identification Node.

These database needs to be mentained and up-dated. That creates a problem with the management of the keys , and with synchronized update of databases, etc. These problems becomes acute when the Service provider have a multitude of identification nodes disseminated through several countries.

Another problem of the conventional schemes for Remote Identification is the repudiation problem, i.e.: if the Presumed Caller can deny the genuinely of the communication, after the Secure–Identification- By-Means-Of-Authentication procedure has been completed.

The repudiation potential problem is, indeed, the most frightful problem affronted for the Service Providers, servicing Remote Callers.

In order to illustrate the exposure and liability affronted by the Service Providers due to potential Repudiation effects, we will use a case of a Lottery Operator supporting Remote Callers bets.

Let assume that Remote Callers are placing bets and paying with Credit Cards after being authenticated by the said Secure–Identification- By-Means-Of-Authentication procedure. Now, there will be one fortunate Remote Caller who wins the prize; all the others , i.e. 1,000,000 gamblers lost.

Lets assume that one of them repudiates the Identification Process (referred also as transaction), stating that he did not call at all, and that the Lottery Operator itself fabricated the communication. Due to the fact that the Lottery Operator's ID Node authenticates the remote callers, it is true that the Lottery Operator ID Node has the full capabilities to compute as many Valid Dynamic Codes as the Lottery Operator may desire. Therefore, the Lottery Operator has the full capabilities to fabricate transactions, and there is no way to prove who is right, the Lottery Operator or the remote caller who repudiates the transaction.

Naturally, if just one case like this appears in the newspapers, all the others 999,999 losers may claim the same, with the disastrous consequent effect on the Lottery Operator.

Therefore, the authentication method (the said Secure–Identification- By-Means-Of-Authentication procedure) carries, intrinsically, the foundation for repudiation effects, due to the fact that the receiving- authenticating side should know how to compute the secret Dynamic Code as accurately as the Remote caller.

A further drawback of the said Secure–Identification- By-Means-Of-Authentication method is the fact that a Remote Caller is trackable. That means an eavesdropper can follow all the transaction made by one particular Remote Caller due to the fact that he/she is sending the same constant Identification Message every time.

This tracking possibility creates a lack of security and privacy for some of the Remote Callers (specially government officers i.e.: ministers, police officers etc.).

Summary of the invention

The present invention discloses methods and algorithms for secure identification, whereas such identification can be remote or on-the-spot, without the necessity of a complementary step of authentication. In other words, according to the method of this invention , a One Step-One Way Identification Procedure is presented, being such step a secure identification, without the need of an additional authentication step. This identification , avoids the possibility of impersonation, due to the fact that it is valid only one time and consequently cannot be intercepted and re-used, it is non-repudiable and non-trackable. In addition, according with the present invention , the Identification Node/System Administrator, i.e.: Service Provider (Social Securities, Corporation's Customer services, Telecoms, financial institutions: brokers , banks, etc.) willing to provide services through the phone lines , Internet or any other media, does not need a database at the identification node.

The invention consists of the use of reversible mathematical algorithms as shown below, according with the principles disclosed in US patent 5,524,072.

One of the preferred specifications of the method of this invention is the use of reversible mathematical algorithms in conjunction with the Challenge /Response method whereas the Identification Node generates a random challenge.

This challenge is transmitted to the Remote Caller. The Remote Caller generates a response, a totally dynamic one (no constant part) , and such response constitutes the secure Identification Message. It is important to emphasize that, according to the method of this invention, the ID Node does not need to know in advance who is the Remote Caller, in order to identify him.

That means that the Identification Node is able to check the identification , without the need of a database, or without the need to knowing in advance who the Remote Caller is supposed to be.

This invention also provides a method for a non repudiable identification, which means that the Remote Caller will not be able to claim that the identification message was fabricated by the Service Provider, because , according to the method of this invention , there are 3 entities, namely, Remote Caller , Service Provider/ID Node and Arbitrator, and none of the 3 knows everything .

Also according to the method of this invention, the response message computation is composed of more than one step, being the first step, a computation of the result (in the following referred to as *RI*) inferred from the Arbitrator Seed number. The said result *RI* , will be one of the arguments of the Reversible algorithm .

The anti- repudiation feature of this invention consists of the following steps:
The Service provider receives a signature, applies on it the reverse of the said reversible algorithm, and recuperates the original arguments , including *RI*. Due to the fact that *RI* is computed by the Remote Caller using the Arbitrator seed number , which is not known to the Service Provider, that means that a correct *RI*, validated and corroborated by the Arbitrator, can come only from the Remote Caller, and can not be fabricated by the service provider.

Brief description of the drawing figures

The subject of the invention will, hereinafter, be described in conjunction with the appended drawing figures, wherein like numerals do designate like elements, and:
figure #1 represents the flow diagram of the One Time set-up functionality, for the Arbitrator and for the System (Service Provider, referred to, also, as ID Node, but can represent a multitude of ID Nodes) for one particular preferred specification referred to hereinafter.

figure #2 represents the flow diagram of the identification process for one particular preferred challenge –response based specification referred hereinafter.

figure #3 represents the flow diagram of the anti-repudiation process for one particular preferred specification referred hereinafter

figure #4 represents the flow diagram of the identification process for one particular preferred time based specification referred hereinafter

figure #5 represents the flow diagram of the identification process for one particular preferred time-based with drift correction specification referred hereinafter

figure #6 represents the flow diagram of the identification process for another particular preferred time-based with drift correction specification referred hereinafter, which use database at the ID Node

Figure #7 represents a preferred specification for the function FTolerance referred hereinafter

Detailed Description of Preferred Exemplary Embodiments

The present invention provides a method and algorithms for secure identification of multitude of to-be-identified entities (each one referred to as Remote Callers or RC) against one central identification entity , (referred as the service Provider/ System Administrator/ ID node) without a need of a database at the ID Node. Further, the method and the algorithm's architecture provides the means for a one-way , non-repudiable, non-trackable identification.

According to the method of this invention :
a reversible algorithm (in the following referred to as a Reversible Function) F_SYS_{ssn} is selected out of a family of algorithms, by the Service Provider, preferably by means of a selection of a Seed Number (in the following referred to as System Seed Number or SSN) and such reversible algorithm is distributed to all of the Remote Callers; also a specific Remote Caller's Identification number (referred to as *C_ID*) is provided, by the Service provider , to each of the Remote Callers or agreed with each RC (i.e.: Driving License number).

According with one specification of the method of this invention , for each identification, the Service Provider selects a random number (in the following referred to as *Ch* or Challenge) and transmits it to the Remote Caller. The Remote Caller applies the function F_SYS_{ssn} to the arguments: *C_ID* and *Ch* generating response *R2*.

$$F\_SYS_{ssn}(C\_ID, Ch) = R2$$

This response is transmitted to the SP's Identification Node.
At the Identification node the reverse of the system function $F\_SYS_{ssn,}$ referred to as

$F^{-1}\_SYS_{ssn}$, is applied obtaining/recuperating *C_ID* and a number presumed to be *Ch,* in the following referred to as *Ch'*

$F^{-1}\_SYS_{ssn}$ (R2)=*C_ID,Ch'*

Now the ID Node/ System Administrator compares whether the *Ch* sent , and the *Ch'* received/computed match, and if so, it means that *C_ID* is a correct and valid identification.

One preferred specification of this invention consists of the usage of the so called Rabin Algorithm in the following manner:

The Service Provider selects 2 large prime numbers, P1 and P2, each one of them should be congruent with 7 (mod 8) ,or alternatively, selects a System Seed Number (SSN), and the 2 prime numbers are consequently inferred from it (See Fig#1 blocks #5 and 6). These 2 prime numbers will determine , as parameters, the Reverse Algorithm , as explained below, but such prime numbers will not be transmitted , nor communicated to the Remote Caller, and will remain as secret system keys under the supervision of the Service Administrator(See Fig#1 block #7).

Now the Service Provider makes the product P1*P2= SM (which is referred to as the System Module) and sends the said System Module (SM) to potential Remote Callers, not necessarily open, but embedded in the function F_SYS_{ssn} (See Fig#1 block #8). This function F_SYS_{ssn} can be masked on a chip.

One the preferred model for the function F_SYS_{ssn} can be specified by means of a function $FF_M(x)$ as follows:

$$FF_M(x) = x^2 \pmod M$$

The Remote caller makes the following computation:

$FF_{SM}(Ch) = L_1$ and then

$FF_{SM}(L_1) = L_2$ and then

$FF_{SM}(L_2) = L_3$ and then

$FF_{SM}(L_3) = L_4$ and then

.....=.....

and so on for q (where q is any natural)

$F_{SM}(L_{q-1}) = L_q$

And then, defining the function *LSB*(X) as the "Least Significant (n)Bit(s) of" X

(Where n is any natural number)

$FF_{SM}(L_q) = T1$ and then *LSB*(T1)=c1

$FF_{SM}(T1) = T2$ and then *LSB*(T2)=c2

$FF_{SM}(T2) = T3$ and then *LSB*(T3)=c3

$FF_{SM}(T3) = T4$ and then *LSB*(T4)=c4

.....=.....= ...

and so on till n (where now n is the amount of digits of the concatenation $Ch \circ C\_ID$)

$FF_{SM}(Tn-1) = Tn$ and then *LSB*(Tn)=cn

now the Remote Caller concatenates the c1,...,cn and refer to it as C

$C=c1 \circ c2 \circ c3 \circ c4 \circ \circ \circ \circ \circ \circ \circ \circ cn$

Where $\circ$ stands for concatenation

Then he/ she bitwise-xors C with the concatenation $Ch \circ C\_ID$

$$(c1 \circ c2 \circ c3 \circ c4 \circ \circ \circ \circ \circ \circ \circ \circ cn) \otimes (Ch \circ C\_ID) = V$$

Where $\otimes$ stands for bitwise-xor

And then the Remote Caller computes Tn+m (being m any natural) in the following manner:

$FF_{SM}(Tn)=Tn+1$

and so on for n+m

$FF_{SM}(Tn+m-1)=Tn+m$

Now the Remote Caller concatenates V and Tn+m

$R2=V \circ Tn+m$

We will refer to R2 as the Cipher

$R2=V \circ Tn+m$

The Remote Caller now transmits the Cipher R2 to the ID Node/System Administrator; due to the fact that ID Node/ System Administrator knows the prime numbers P_1 and P_2 the ID Node/ System Administrator can reverse the algorithm and recover *Ch* and *C_ID*, as follows:

The **ID Node/ System Administrator** uses the Euclidean Algorithm in conjunction with P_1 and P_2, to compute

$$L1 \text{ and } L2$$

so that

$$L_1 P_1 + L_2 P_2 = 1 \pmod{SM}$$

Define

$$K_1 = L_1 P_1$$

$$K_2 = L_2 P_2$$

compute

$$Y_1 = Tn+m \pmod{P_1}$$

$$Y_2 = Tn+m \pmod{P_2}$$

compute

$$Z_1 = Y_1^{((P_1+1)/4)} \pmod{P_1}$$

if $Z_1^{((P_1-1)/2)} = -1 \pmod{P_1}$ then $Z_1 = P_1 - Z_1$

compute

$$Z_2 = Y_2^{((P_2+1)/4)} \pmod{P_2}$$

if $Z_2^{((P_2-1)/2)} = -1 \pmod{P_2}$ then $Z_2 = P_2 - Z_2$

compute

$$Tn+m-1 = Z_1 K_2 + Z_2 K_1$$

applying the same procedure to T_{n+m-1} compute T_{n+m-2}

and so on, obtaining all the T_s until T_0

now the ID Node/ System Administrator proceeds as follow:
it takes Ti and computes

$$C_i = LSB(Ti)$$

for i=1,,n .
The results are concatenated

$$C = c1 \circ c2 \circ c3 \circ c4 \circ \circ \circ \circ \circ \circ \circ \circ cn$$

Then the ID Node/ System Administrator XORs C with the said V

[$V=(c1 \circ c2 \circ c3 \circ c4 \circ \circ \circ \circ \circ \circ \circ \circ c_n) \otimes (Ch \circ C\_ID)$ sent for the RC]

and obtain

$(c1 \circ c2 \circ c3 \circ c4 \circ \circ \circ \circ \circ \circ \circ \circ cn) \otimes V =$

$(c1 \circ c2 \circ c3 \circ c4 \circ \circ \circ \circ \circ \circ \circ \circ c_n) \otimes (c1 \circ c2 \circ c3 \circ c4 \circ \circ \circ \circ \circ \circ \circ \circ c_n) \otimes (Ch \circ C\_ID) =$

$(Ch \circ C\_ID)$

Having recuperated the *C_ID*, the identification is completed, while the recuperation of the *Ch* eventually authenticates it, to the extent that *Ch* should be exactly the *Ch* sent to the RC.

It is worth to note that the example shown represents an extremely secure methodology to identify a Remote Caller due to the fact that the sensitive secrets ,namely, the knowledge how to factorize the SSN (namely P_1 and P_2) is only available at the ID Node/ System Administrator and not at the Remote Caller's level (i.e.: in the tokens).

A further variation of this method is an identification method for a multitude of to-be-identified-entities (RCs) against a central identification entity (ID Node/ Service Provider), the said identification arbitrated by an independent arbitrator entity comprising the steps of:

the independent arbitrator entity characterizing specific algorithms for each of the RCs, and distributing the said algorithms to each of the RCs;

the Service Provider distributing one reversible algorithm for all and to all of the RC;
and for each identification operation, the RC:
applies the independent arbitrator entity's algorithm to a challenge **Ch** provided by the ID Node for such identification operation computing a first result **R1**, and then
applies the central identification entity's reversible algorithm to the said challenge **Ch** , to its identification data **C_ID**, and to the said first result **R1** computing a second result **R2**(the Cipher),
and the RC transmits the said **R2** to the ID Node.
The ID Node applies the reverse of the reversible algorithm to the received second result **R2**, computing a presumed challenge **Ch'** , a **C_ID**, and a presumed first result **R1**;
and whereas the ID Node compares the challenge previously send **Ch** to the said presumed challenge **Ch'**, and if they are identical,
the ID Node sends to the independent arbitrator entity
the presumed first result **R1**,
together with RC's identification data (which can be C_ID or other data, like a RC's Serial Number)), and
the said challenge previously sent to the RC, namely Ch,

and whereas the arbitrator entity retrieves the specific algorithm sent by him to the specific RC and applies the said specific algorithm to the challenge **Ch** computing the true first result **R1**, and compares such true first result **R1**

with the presumed first result **R1**, and , if eventually the both are identical, the arbitrator entity corroborates the identification to the ID Node; and whereas the ID Node , having received the corroboration from the arbitrator entity , accepts the presumed identification data **C_ID** as true identification data corresponding to the RC.

Accordingly, the above mentioned three entities- non repudiable, non-trackable identification methodology can be more precisely specified as follows: defining a third independent party, referred to as Arbitrator or certificator, beside the 2 entities mentioned above, namely the Remote Caller and the ID Node/ System Administrator; and , additional to the steps above mentioned the Arbitrator selects first an algorithm: (referred as $\mathbf{F\_ARB}^{1}_{ASN}$) which is preferably selected out of a family of algorithms $\mathbf{F\_ARB}^{1}$ by means of a selection of a seed number (in the following referred as Arbitrator Seed Number =*ASN*) (See Fig#1 Block 1)

The fundamental purpose of this step (or purpose of this algorithm ($\mathbf{F\_ARB}^{1}_{ASN}$) according with this particular aspect of this invention is to generate a different pseudo-random number for each RC (See Fig#1 block #2). This pseudo-random number, referred to as *RCAN* (Remote Caller's Arbitrator Number) , will determine the algorithm (see Fig #1 Block #3) which will correlate the transaction related number *Ch* (and optional *C_ID*) with an arbitrator inferred number referred to as *R1*. The *RCAN* should be the result of the product of 2 large prime numbers, p'1 and p'2 , both of them congruent with 3 (mod4). In the following this algorithm is referred to as $\mathbf{F\_ARB}^{2}_{RCAN}$ (See Fig#1 block #4). Hence the computed *R1* will be different and specific for each transaction.

Accordingly this algorithm($F\_ARB^1_{ASN}$) should be applied to an RC's openly known number, like the Remote Caller's Serial Number or any other publicly known number associated with the Remote Caller, in order to generate the *RCAN* (Remote Caller's Arbitrator Number)

$F\_ARB^1_{ASN}$ (Serial Number)=*RCAN*

The second algorithm (or function) selected by the Arbitrator is referred to as $F\_ARB^2_{RCAN}$ which is also preferably determined out of a family of algorithms by means of a the previous obtained Remote Caller Arbitrator Number *RCAN*. (See Fig#1 block #3).

It is worth to note that the Arbitrator needs to select just one number , namely the *ASN* which is the same for all the cases belonging to such Arbitrator, all the other numbers , like the *RCAN* , which are different and dedicated for each RC, and the *R1*, which are different for each transaction , are inferred from the *ASN*

The Remote Caller applies first the $F\_ARB^2_{RCAN}$ to *Ch* and (optional, to *C_ID*) getting a response *R1* (See Fig#2 block #11).

$F\_ARB^2_{RCAN}$ (*C_ID*, *Ch*)=*R1*

Until here the arbitrator related aspect of this preferred specification;
now the Remote Caller accomplishes the Service Provider System related part of this preferred specification of the identification algorithm:
the Remote Caller applies the said F_SYS_{ssn} to *Ch* , *C_ID* and *R1* getting *R2*
(See Fig#2 block #12).
F_SYS_{ssn} (*Ch* , *C_ID*, *R1*) = *R2* (the Cipher)
And sends such Cipher to the ID Node/System Administrator (See Fig#2 block #13).

The ID Node receives the Cipher. (See Fig#2 block #14) In order to identify non-repudiably the remote caller , without using any data-base the ID Node/ System Administrator , who knows P_1 and P_2, applies the reverse of F_SYS_{ssn} to *R2* , that is

$F^{-1}\_SYS_{ssn}$ is applied to *R2* getting the original arguments ,namely, *C_ID*, *Ch'* and *R1*

$F^{-1}\_SYS_{ssn}(R2)=C\_ID,Ch',R1$

Actually, checking if *Ch'* eventually matches with the *Ch* transmitted, meaning that the *C_ID* is validated as an authentic Caller Identification Number (See Fig#2 block #15 and 16).

Now the *R1* is kept for repudiation cases, that is, cases where the Remote Caller denies that he /she had sent the said Cipher , *R2*, or , in other words, when the Remote Caller claims that the *R2* was fabricated by the SP. (See Fig#2 block #17).

According with the three entities - non repudiable identification methodology presented in this invention (See Fig#3). , just in case of repudiation problems or , alternatively for each desired case (i.e.: all cases) , the Service Provider will provide the Arbitrator with (optional the *C_ID* and), the RC's Serial Number, the *Ch* and the Presumed *R1*. (See Fig#3 blocks #19,20 and 21).

Now the Arbitrator can apply the $F\_ARB^1_{ASN}$ to the Serial number obtaining the *RCAN*

$F\_ARB^1_{ASN}$(Serial Number)=*RCAN*

(See Fig#3 block #22).

Now the Arbitrator applies the $F\_ARB^2_{RCAN}$ (optional ,to *C_ID* and) to *Ch* and gets the true *R1*

(See Fig#3 block #23).

$F\_ARB^2_{RCAN}(C\_ID,Ch)=R1$

Which is the true *R1*

and, therefore the arbitrator can check/ compare if the true *R1* matches with the Presumed *R1*. (See Fig#3 blocks #24 and 25). If , eventually , these two numbers are not identical, the *R2*, is suspected to be a fabrication(See Fig#3 block #27).. But if the two numbers match, there is no doubt that the *R2* is coming from the Remote Caller, because there is no way by which the ID Node/ System Administrator can compute (fabricate) a true *R1*.

(See Fig#3 block #26).

One preferred specification of the method of the three entities - non repudiable identification presented in this invention is as follows:

The RC by means of the algorithm $F\_ARB^2_{RCAN}$ which was characterized by the arbitrator throughout *RCAN* computes the transaction specific *R1*, using the transaction specific Challenge sent by the ID Node/ System Administrator , as follows:

The Remote Caller computes

$$M_1 = Ch^2 (\text{mod } RCAN)$$

$$M_2 = M^2_1 (\text{mod } RCAN)$$

$$\ldots\ldots = \ldots\ldots\ldots\ldots$$

and so on , until any selected p (natural)

$$S_1 = M_p = M^2_{p-1} (\text{mod } RCAN)$$

Then the Remote Caller computes $C'_1 = LSB(S_1)$

Consequently

$$S_2 = S^2_1 \pmod{RCAN} \rightarrow C'_2 = LSB(S_2)$$

and so on

$$S_n = S^2_{n-1} \pmod{RCAN} \rightarrow C'_n = LSB(S_n)$$

The result is:

$$RI = C'_1 \circ \circ \circ \circ \circ \circ \circ \circ C'_n$$

Consequently the step referred to above as $F\_ARB^2_{RCAN}(Ch) = RI$ was accomplished.

Referring now to the System related part of the methodology, a preferred specification for the function F_SYS_{ssn} is presented.

According with this specification of the function F_SYS_{ssn}, the System Administrator selected Seed (SSN) is used in order to infer the SM (as explained above).

Now the RC computes

$$T_1 = RI^2 \pmod{SM} \Rightarrow C_1 = LSB(T_1)$$

$$T_2 = T^2_1 \pmod{SM} \Rightarrow C2 = LSB(T_2)$$

..... = ... ⇒.....=.....

$$T_n = T^2_{n-1} \pmod{SM} \Rightarrow C_n = LSB(T_n)$$

$$T_{n+1} = T^2_n \pmod{SM}$$

$$\ldots\ldots = \ldots\ldots\ldots\ldots$$

$$T_{n+m} = T^2_{n+m} \pmod{SM}$$

Now concatenating the Cs

$$C = C_1 \circ \circ \circ \circ \circ \circ \circ \circ C_n$$

Then, computes the Cipher message :

$$R2 = [(C\_ID \circ CH)] \oplus C] \circ T_{n+m}$$

Therefore, the step referred to above as F_SYS_{ssn} (Ch)=$R2$ was accomplished. Obviously, while the C_ID is always the same for a particular Remote Caller, the cipher (the message which is actually sent) is different from any prior Cipher to the extent that Ch is different of any prior Ch which has been sent to such RC.

The Remote Caller now transmits the Cipher $R2$ to the ID Node/System Administrator Due to the fact that ID Node/ System Administrator knows the prime numbers P_1 and P_2 the ID Node/ System Administrator can reverse the algorithm and recover Ch and C_ID, as follows:

The **ID Node/ System Administrator** uses the Euclidean Algorithm in conjunction with P_1 and P_2, to compute what is referred to

as the reverse algorithm or Reverse Algorithm, $F^{-1}\_SYS_{ssn}$, as one preferred specification:

compute L1 and L2

so that

$$L_1 P_1 + L_2 P_2 = 1 (\text{mod } SM)$$

Define

$$K_1 = L_1 P_1$$

$$K_2 = L_2 P_2$$

compute

$$Y_1 = Tn+m \ (\text{mod } P_1)$$

$$Y_2 = Tn+m \ (\text{mod } P_2)$$

compute

$$Z_1 = Y_1^{((P_1+1)/4)} \ (\text{mod } P_1)$$

if $Z_1^{((P_1-1)/2)} = -1 \ (\text{mod } P_1)$ then $Z_1 = P_1 - Z_1$

$$Z_2 = Y_2^{((P_2+1)/4)} \ (\text{mod } P_2)$$

if $Z_2^{((P_2-1)/2)} = -1 \ (\text{mod } P_2)$ then $Z_2 = P_2 - Z_2$

compute

$$Tn+m-1 = Z_1 K_2 + Z_2 K_1$$

applying the same procedure to T_{n+m-1} compute T_{n+m-2}

and so on obtaining all the Ts until T_0 is recuperated

Now the ID Node/ System Administrator proceeds as follow:
It takes Ti and computes

$$C_i = LSB(Ti)$$

and so on for i=n,n-1,...,1, (namely , also T_1 is recuperated)
Clearly, the Reverse algorithm applied to T_1 generates the *R1* which is stored together with the correspondent *Ch* for potential repudiation cases.

The cj results are concatenated

$$C = c1 \circ c2 \circ c3 \circ c4 \circ \circ \circ \circ \circ \circ \circ \circ cn$$

Then the ID Node/ System Administrator XORs C with the said V
and obtains

$$(c1 \circ c2 \circ c3 \circ c4 \circ \circ \circ \circ \circ \circ \circ \circ cn) \otimes V =$$

$$(c1 \circ c2 \circ c3 \circ c4 \circ \circ \circ \circ \circ \circ \circ \circ c_n) \otimes (c1 \circ c2 \circ c3 \circ c4 \circ \circ \circ \circ \circ \circ \circ \circ c_n) \otimes (Ch \circ C\_ID) =$$

$$(Ch \circ C\_ID)$$

Having recuperated the *C_ID*, the identification is completed, while the recuperation of the *Ch* eventually authenticates it, to the extent that *Ch* should be exactly the *Ch* sent to the RC.

In case that the Remote Caller repudiates the transaction, the ID Node/ System Administrator will provide the Arbitrator with the RC's Serial Number, the transaction *Ch* and the presumed *R1* (optional, the *C_ID*)

The Arbitrator will apply the . $\mathbf{F\_ARB}^1_{ASN}$ to the Serial number obtaining the *RCAN*

$\mathbf{F\_ARB}^1_{ASN}$ (Serial Number)=*RCAN*

Now the Arbitrator applies the $F\_ARB^2_{RCAN}$ (optional ,to *C_ID* and) to *Ch* and gets the true *R1* , as follows

The Arbitrator computes

$$M_1 = Ch^2 (\text{mod } RCAN)$$

$$M_2 = M^2_1 (\text{mod } RCAN)$$

and so on , until

$$S_1 = M_{40} = M^2_{39} (\text{mod } RCAN)$$

Then the Arbitrator computes $C'_1 = LSB(S_1)$

Consequently

$$S_2 = S^2_1 \,(\text{mod } RCAN) \qquad \rightarrow C'_2 = LSB\,(S_2)$$

and so on

$$S_n = S^2_{n-1} \,(\text{mod } RCAN) \qquad \rightarrow C'_n = LSB\,(S_n)$$

The result is the true *R1* for this specific Remote Caller and for this specific transaction (*Ch*)

$$R1 = C'_1 \circ \circ \circ \circ \circ \circ \circ \circ C'_n$$

and, therefore the Arbitrator can check if the true *R1* matches exactly with the Presumed *R1* as stated by the ID Node/system administrator. If , eventually , these two numbers are not identical, the *R2* was, indeed, a fabrication. But if the two numbers match, there is no doubt that the *R2* is coming from the Remote Caller, because there is no way by which the ID Node/ System Administrator can compute (fabricate) a true *R1*.

It is worth to note that the example shown represents an remarkably secure methodology to identify a Remote Caller due to the fact that the sensitive secrets , the knowledge how to factorize the *SSN* ,is only available at the ID Node/ System Administrator and not at the RC level (or in the tokens).

Another preferred specification of the ID algorithms of this invention are the methods and algorithms disclosed above, but , instead of the ID Node/ System Administrator sending a challenge (*Ch*) to the RC, the Remote Caller will use the Cipher's Generation Time , the time of the moment of computing the Cipher , or the Generation Time and Date, or the GMT and Date, with a resolution of seconds (or other). Each one of the

these possibilities will be referred to as Generation Time or GTime. (See Fig#4 block #10b).

The Remote Caller proceeds as in fig #4, hence the method, beside being non-repudiable and non-trackable, *is one-way*. The RC sends a message, which identifies himself/herself against the ID Node (or one of the ID Nodes of the Service Provider System), and this ends the protocol.

Therefore, all the time that the Identification is made on-line, both sides, Remote Caller and ID Node/ System Administrator know the correspondent GTime/ Reception Time (RTime) (See Fig#4block #14b)., and consequently, the ID Node/ System Administrator can verify the identification (See Fig#4 blocks #15b and 16b).

If the Identification is not on-line, the Generation Time used in the computation, can be communicated openly to the ID Node.

Coming back to on-line identifications and in order to overcome possible drift between the time measured by the Remote Caller and the time measured by the ID Node, according to the method of this invention, two successive identifications procedures will be accomplished during an identification session ,see fig#5 . Further, the time elapsed between the 2 identification procedures will be determined by the ID Node (see fig #5 Block #19d), which will signal/ trigger to the Remote Caller to send a new Cipher. The fact that the time elapsed is set by the ID node avoids the possibility of an impostor using 2 pre-recorded Ciphers in order to fraud the system.

Hence, measuring the Absolute Value of the drift difference fig #5 Block #27d, the ID Node can overcome the drift problem without compromising security.

Another preferred specification of the method of this invention similar of the specification described in fig 5, but instead of using the Time , the RC will use an RC's specific function of the Time , referred as F_T_RC(Time) preferably a linear function. According with this case (specification) the ID Node/ System Administrator will have the knowledge how to compute such F_T_RC(Time) , and therefore, for this case , a database will be necessary.

One particular model for this specification, is the one which uses functions of the Time referred to as

F_T_RC(Time)= RC_Time =Time + RC_cte,

where RC_cte is a constant value , different for each RC (see fig. #6).

Due to the fact that the method uses a database at the ID Node, anyway, it is convenient to keep the **RC_Time** and **RTime** of the previous transaction (referred as **RC_Time_pre** and **RTime_pre**) in the ID Node database in order to update the drift throughout the updating of the **RC_cte** value, which absorbs the drift as is shown in fig#6.

Basically, the ID Node, after receiving the **RC_Time**, embedded in the Cipher, and registering the **RTime** (Reception Time), will compare the difference :

RC_Time-RTime

with the previous difference (last recorded transaction)RC_Time-RTime, which will be referred as

RC_Time_pre - RTime_pre.

The said difference should be less than the tolerance value. Now , according with the method of this invention, such tolerance value will be a function (**FTolerance**) of the Absolute Value of the difference between RTime and RTime_pre that is

FTolerance (RTime - RTime_pre)

A preferred specification of FTolerance is shown in fig# 7

Albeit the invention has been described herein using specific examples of the function specifications , variation or alteration of such functions presented here do not represent a departure from the spirit of the invention as set forth here. The same is stated here for alterations of the sequential order of the steps or sub-steps used to explain the methodology presented, or the addition of encryption steps, like DES to the cipher or permutations of it or/ and the addition of transaction data to the cipher, whether such transaction data is encrypted or not and, also, the addition of Error Correction algorithms.

Even if the invention has been described herein using a general method for identification, an implementation of this method in tokens of other portable devices at the level of the RC , or /and devices for the ID Node and/or the arbitrator, whereas , as

example, the above named functions are been masked ,or otherwise embedded into chips , do not represent a departure from the spirit of the invention as set forth here.

Although the invention has been described herein in conjunction with the appended drawing figures and specific functions, those skilled in the art will appreciate that the scope of the invention is not so limited. Various modifications in the selection and arrangement of the various components , method and steps discussed herein may be made without departing from the spirit of the invention as set forth above.
1 .

1

שרד המשפטים

סמך זה הינו העתק שנסרק בשלמותו ביום ובשעה המצוינים .

ריקה ממוחשבת מהימנה מהמסמך המצוי בחיק.

תאם לנוהל הבדיקות במשרד המשפטים.

החתום

משרד המשפטים (חתימה מוסדית).

X Claims

1. An identification method for a multitude of to-be-identified entities against a central identification entity, the said identification arbitrated by an independent arbitrator entity, characterized by the steps of:

the independent arbitrator entity characterizing specific algorithms for each, and distributing the said algorithms to each, of the to-be-identified entities;

the central identification entity distributing one reversible algorithm (F_SYS_{SSN}) for all and to all of the to-be-identified entities;

and for each identification operation, the to-be-identified entity:

applies (11) the independent arbitrator entity's algorithm to a challenge (Ch) selected by the central identification entity for such identification operation computing a first result, and then

applies (12) the central identification entity's reversible algorithm to the said challenge, to its identification data and to the said first result computing a second result referred to as a cipher,

and the to-be-identified entity transmits (13) the said second result to the central identification entity;

whereas

the central identification entity applies (14) the reverse $F^{-1}\_SYS_{SSN}$ of the reversible algorithm to the received second result, computing a presumed challenge (Ch'), a presumed to-be-identified entity's identification data (C_ID), and a presumed first result (R1');

and whereas the central identification entity compares (15) the challenge (Ch) previously sent to the said presumed challenge (Ch'), and if they are identical,

the central identification entity sends (21) to the independent arbitrator entity the presumed first result (R1'), together with the to-be-identified presumed entity identification data or other identification data of such entity, and the said challenge (Ch),

and whereas the arbitrator entity retrieves the specific algorithm sent by it to the said to-be-identified entity and applies (23) the said specific algorithm to the challenge computing the true first result, and compares (24) such true first result with the presumed first result, and, if eventually are identical, the arbitrator entity corroborates (26) the veracity of the identification to the central identification entity;

and whereas the central identification entity, having received the corroboration from the arbitrator entity, accepts the presumed identification data as true identification data corresponding to the to-be-identified entity.

2. An identification system for a multitude of to-be-identified entities against a central identification entity, the said identification arbitrated by an independent arbitrator entity, characterized by:

the independent arbitrator entity characterizing specific algorithms for each, having means for distributing the said algorithms to each, of the to-be-identified entities;

the central identification entity having means for distributing one reversible algorithm (F_SYS_{SSN}) for all and to all of the to-be-identified entities;

and for each identification operation, the to-be-identified entity providing:

means for applying (11) the independent arbitrator entity's algorithm to a challenge (Ch) selected by the central identification entity for such identification operation computing a first result, and then

means for applying (12) the central identification entity's reversible algorithm to the said challenge, to its identification data and to the said first result computing a second result referred to as a cipher,

and means for transmitting (13) the said second result to the central identification entity;

whereas

the central identification entity provides means for applying (14) the reverse $F^{-1}\_SYS_{SSN}$ of the reversible algorithm to the received second result,

computing a presumed challenge (Ch), a presumed to-be-identified entity's identification data (C_ID), and a presumed first result (R1');

and whereas the central identification entity provides means for comparing (15) the challenge (Ch) previously sent to the said presumed challenge (Ch'), and if they are identical,

the central identification entity provides means for sending (21) to the independent arbitrator entity the presumed first result (R 1), together with the to-be-identified presumed entity identification data or other identification data of such entity, and the said challenge (Ch),

and whereas the arbitrator entity provides means for retrieving the specific algorithm sent by it to the said to-be-identified entity and applying (23) the said specific algorithm to the challenge computing the true first result, and comparing (24) such true first result with the presumed first result, and, if eventually are identical, the arbitrator entity provides means for corroborating (26) the veracity of the identification to the central identification entity;

and whereas the central identification entity, having received the corroboration from the arbitrator entity, provides means for accepting the presumed identification data as true identification data corresponding to the to-be-identified entity.

3. The identification system of claim 2, wherein:

the arbitrator is configured to store and provide a different algorithm REAN for each of a plurality of to-be-identified remote entities (RE) and comprises a first processor having access to a first memory;

the said means for distributing one reversible a lgorithm F_SYS_{SSN}

comprises an identification device having a second processor having access to a second memory; and

wherein each of said to-be-identified remote entities comprises a remote entity memory and a remote entity processor, and is configured to store one of said arbitrator algorithms, said reversible algorithm and remote entity identity information; and

wherein said first processor cannot access said second memory and said second processor cannot access said first memory.

For the Applicant,

JMB, Factor & Co.
ENCO 720/6.1

שרד המשפטים

סמך זה הינו העתק שנסרק בשלמותו ביום ובשעה המצוינים .

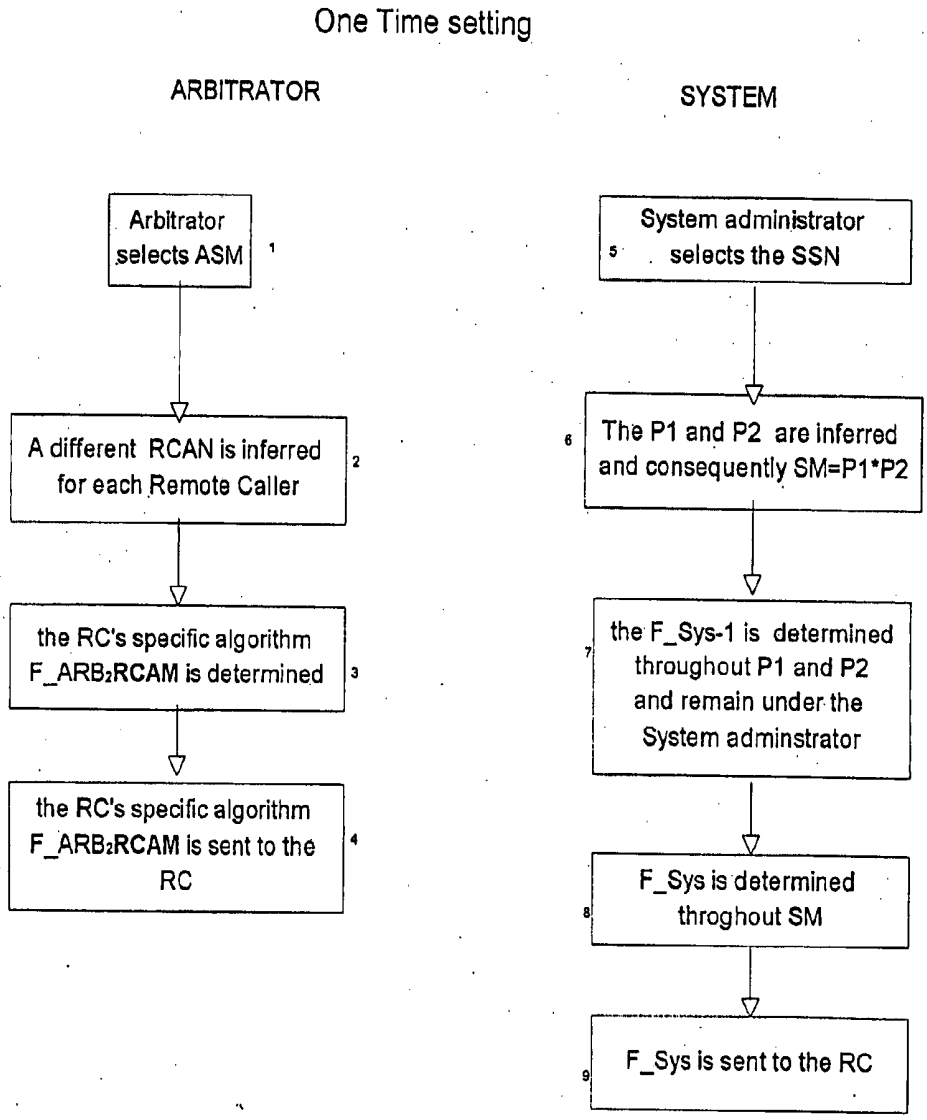
ריקה ממוחשבת מהימנה מהמסמך המצוי בחיק.

תאם לנוהל הבדיקות במשרד המשפטים.

החתום

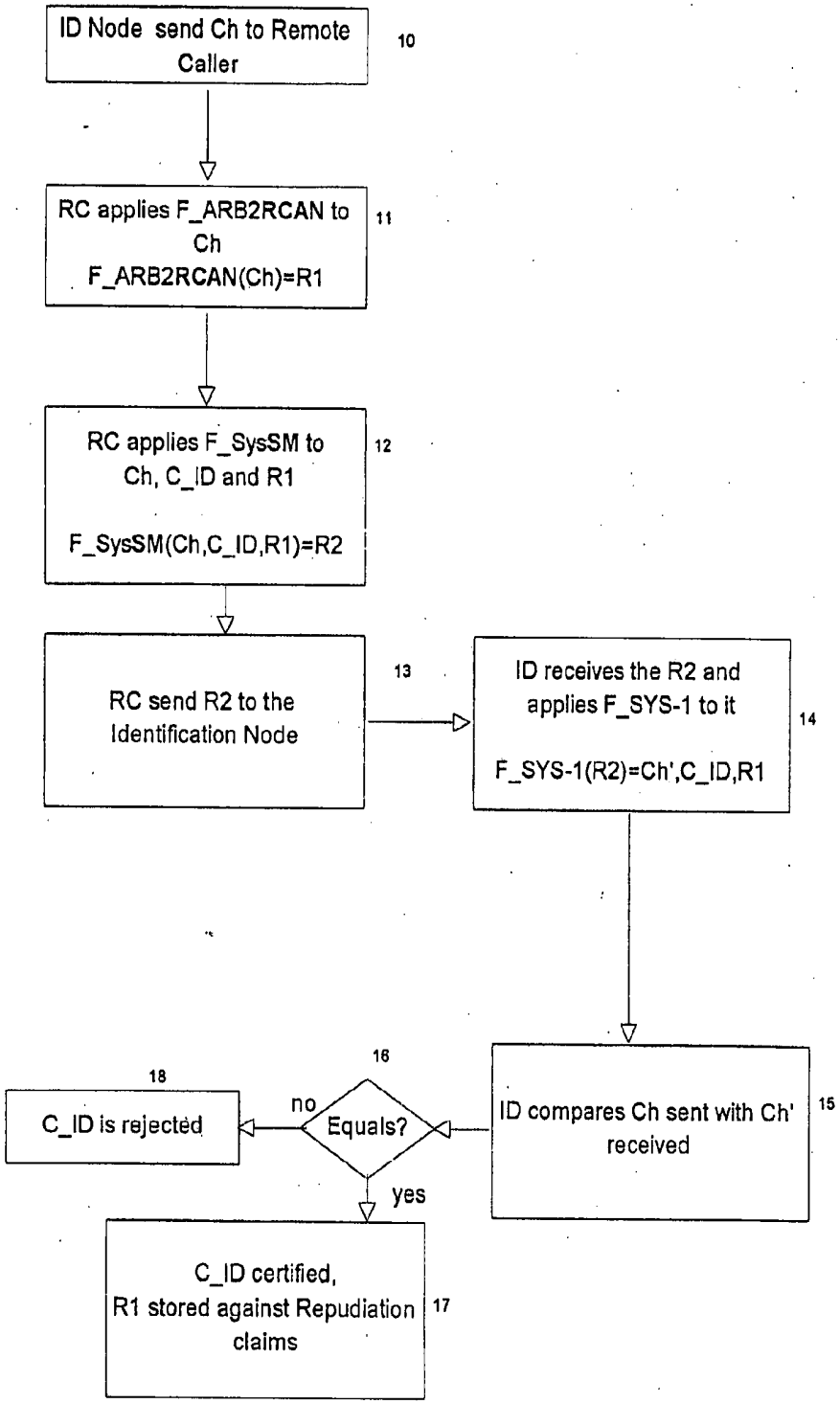
משרד המשפטים (חתימה מוסדית).

Fig#1(Set1)

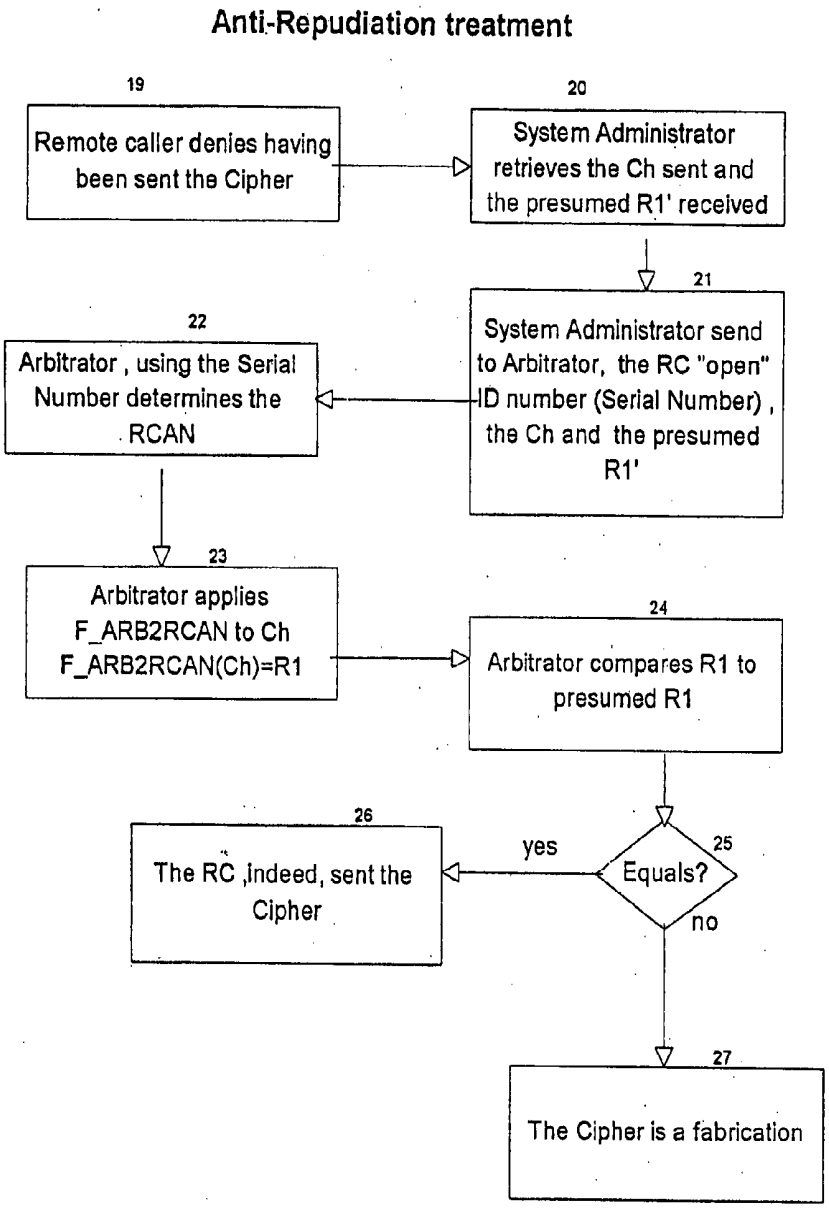

One Time setting
ARBITRATOR
SYSTEM
Arbitrator selects ASM
1
A different RCAN is inferred for each Remote Caller
2
the RC's specific algorithm F_ARB2RCAM is determined
3
the RC's specific algorithm F_ARB2RCAM is sent to the RC
4
System administrator selects the SSN
5
The P1 and P2 are inferred and consequently SM=P1*P2
6
the F_Sys-1 is determined throughout P1 and P2 and remain under the System adminstrator
7
F_Sys is determined throghout SM
8
F_Sys is sent to the RC
9

FIG#2(oper1)

Flow Diagram of the Identification process



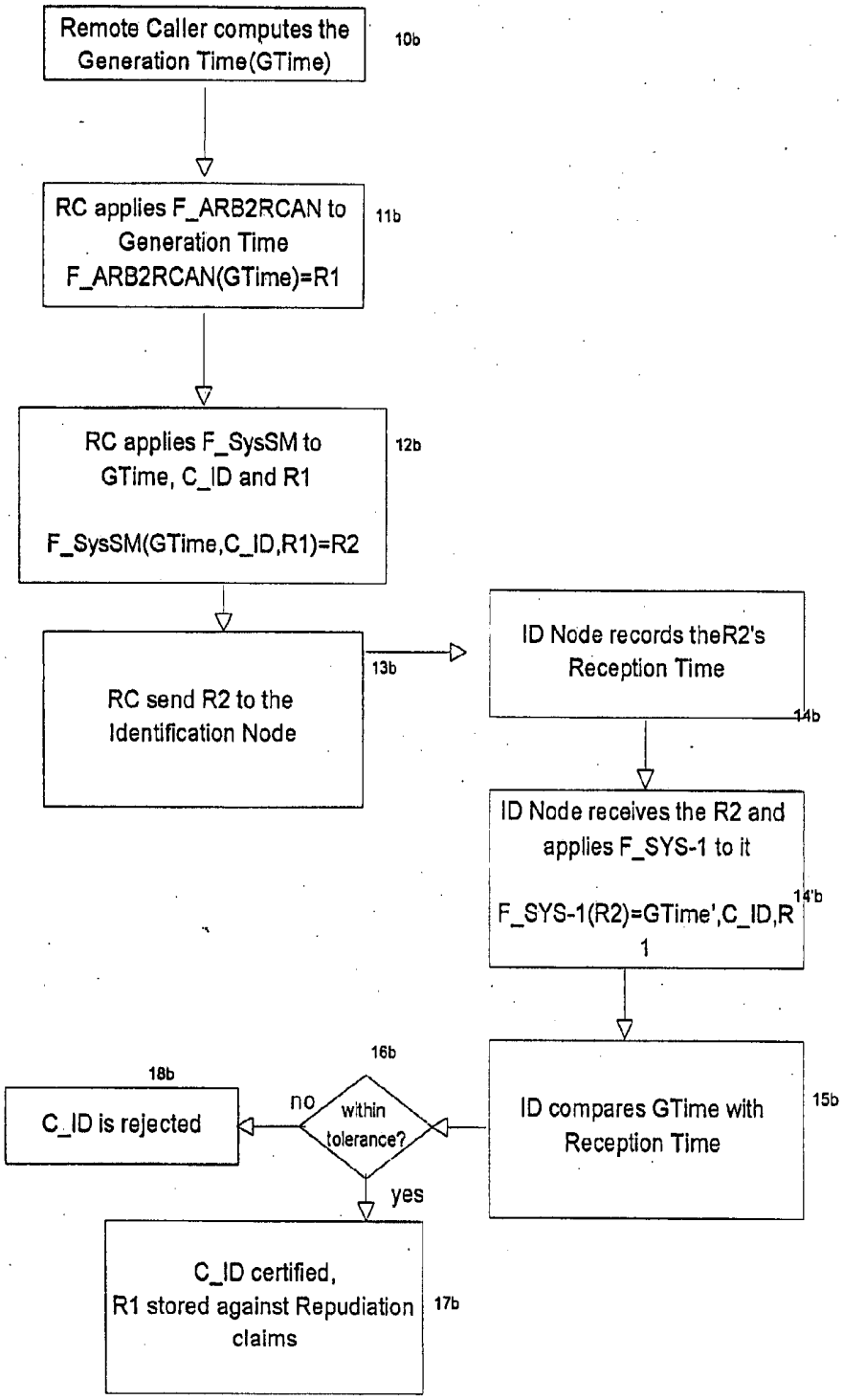
Fig#3(repu 1)


Anti-Repudiation treatment
19
Remote caller denies having been sent the Cipher
20
System Administrator retrieves the Ch sent and the presumed R1' received
21
System Administrator send to Arbitrator, the RC "open" ID number (Serial Number), the Ch and the presumed R1'
22
Arbitrator , using the Serial Number determines the RCAN
23
Arbitrator applies F_ARB2RCAN to Ch F_ARB2RCAN(Ch)=R1
24
Arbitrator compares R1 to presumed R1
25
Equals?
yes
no
26
The RC ,indeed, sent the Cipher
27
The Cipher is a fabrication

FIG#4(oper2)

Flow Diagram of the Identification process

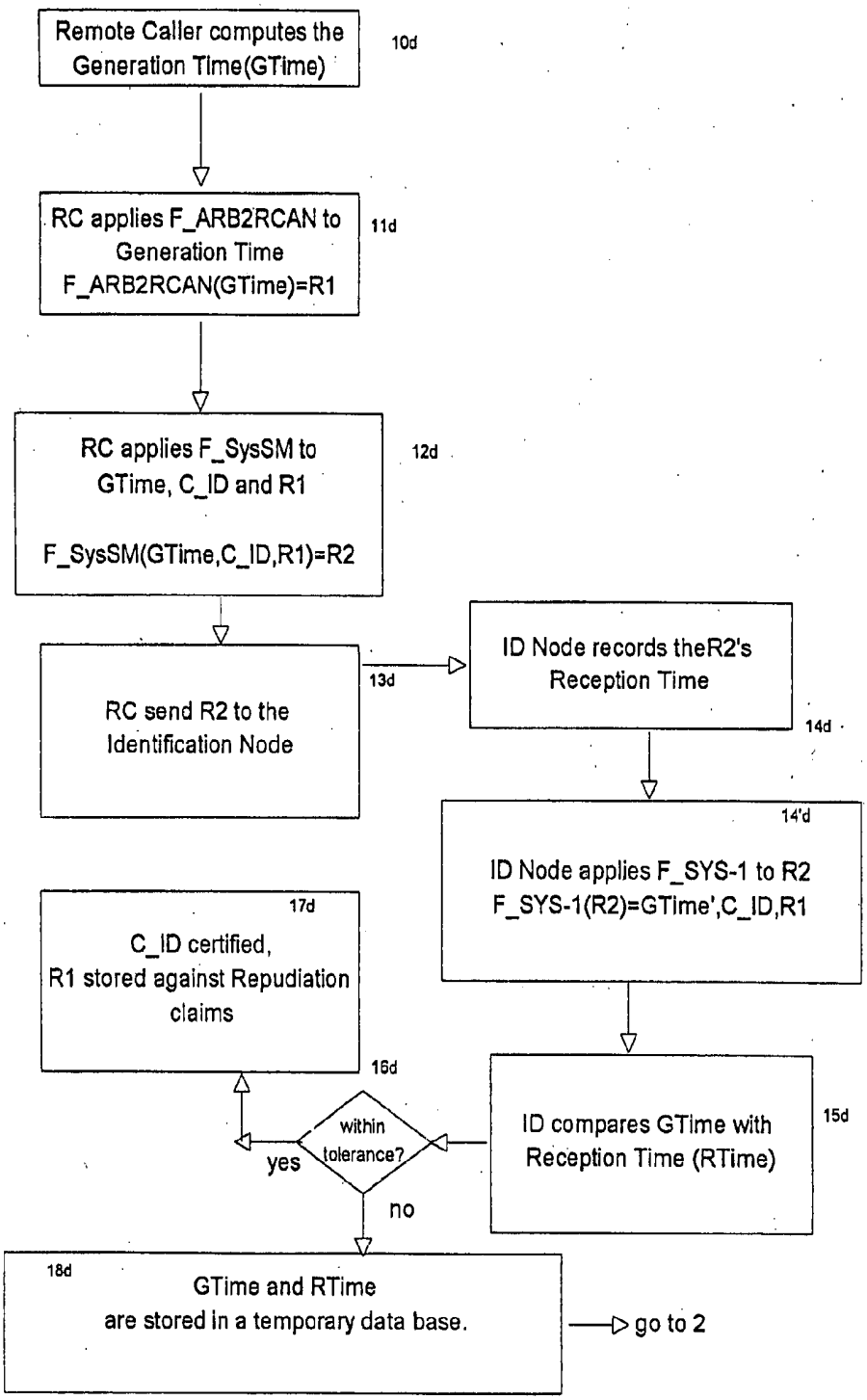
Time based

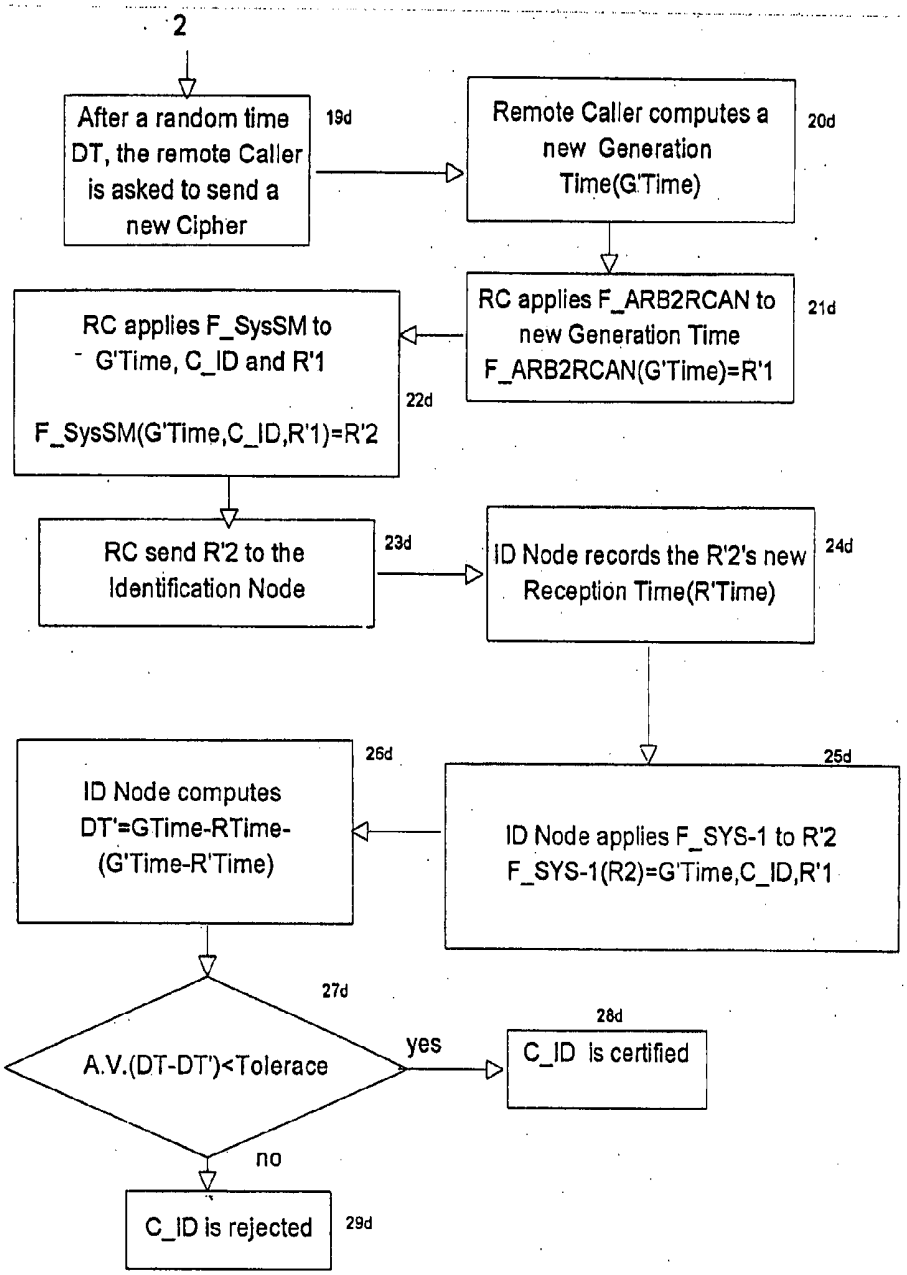


FIG#5(oper3)

Flow Diagram of the Identification process

Time based with drift correction

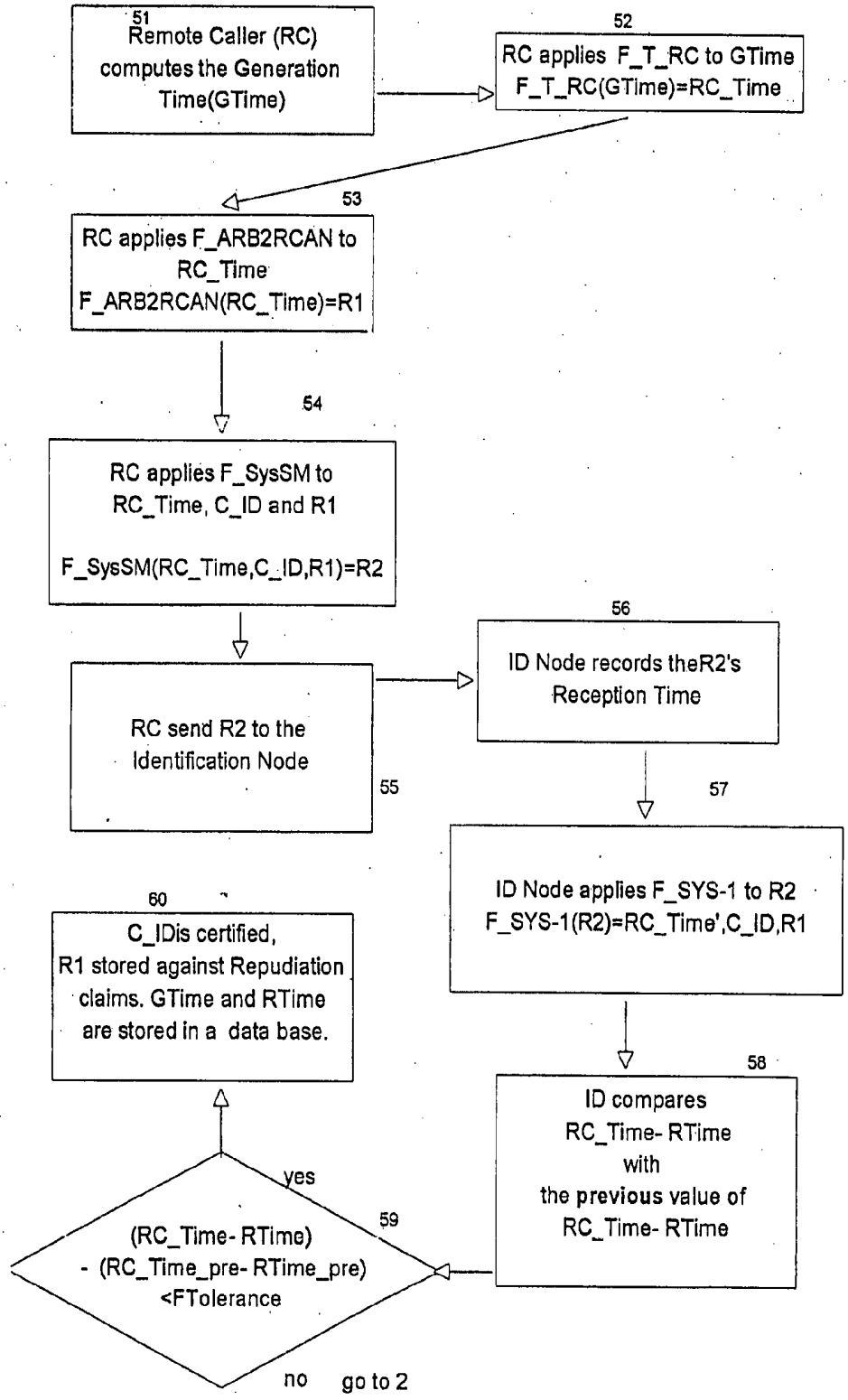



2
After a random time DT, the remote Caller is asked to send a new Cipher
19d
Remote Caller computes a new Generation Time(G'Time)
20d
RC applies F_ARB2RCAN to new Generation Time F_ARB2RCAN(G'Time)=R'1
21d
RC applies F_SysSM to G'Time, C_ID and R'1
F_SysSM(G'Time,C_ID,R'1)=R'2
22d
RC send R'2 to the Identification Node
23d
ID Node records the R'2's new Reception Time(R'Time)
24d
ID Node applies F_SYS-1 to R'2
F_SYS-1(R2)=G'Time,C_ID,R'1
25d
ID Node computes DT'=GTime-RTime-(G'Time-R'Time)
26d
A.V.(DT-DT')<Tolerace
27d
yes
C_ID is certified
28d
no
C_ID is rejected
29d

FIG#6(oper4)

Flow Diagram of the Identification process

Function of Time based, with drift correction



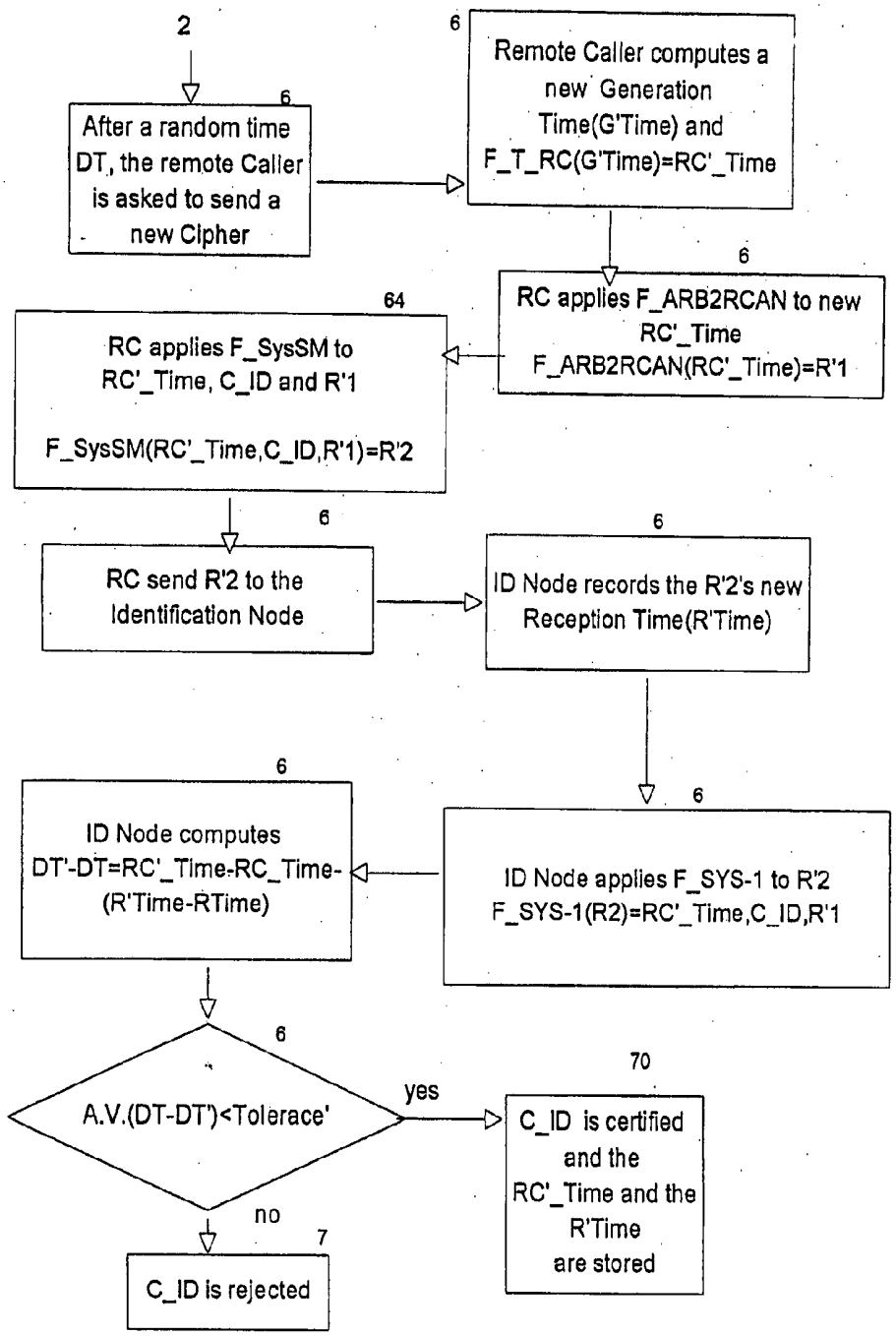
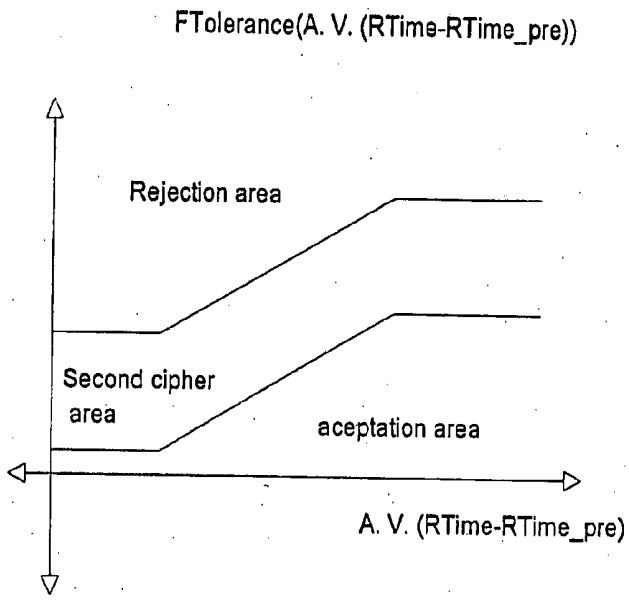

2
6
After a random time DT, the remote Caller is asked to send a new Cipher
6
Remote Caller computes a new Generation Time(G'Time) and F_T_RC(G'Time)=RC'_Time
6
RC applies F_ARB2RCAN to new RC'_Time F_ARB2RCAN(RC'_Time)=R'1
64
RC applies F_SysSM to RC'_Time, C_ID and R'1
F_SysSM(RC'_Time,C_ID,R'1)=R'2
6
RC send R'2 to the Identification Node
6
ID Node records the R'2's new Reception Time(R'Time)
6
ID Node applies F_SYS-1 to R'2 F_SYS-1(R2)=RC'_Time,C_ID,R'1
6
ID Node computes DT'-DT=RC'_Time-RC_Time-(R'Time-RTime)
6
A.V.(DT-DT')<Tolerace'
yes
70
C_ID is certified and the RC'_Time and the R'Time are stored
no
7
C_ID is rejected

Fig.# 7 (tolerance)



שרד המשפטים

סמך זה הינו העתק שנסרק בשלמותו ביום ובשעה המצוינים .

ריקה ממוחשבת מהימנה מהמסמך המצוי בחיק.

תאם לנוהל הבדיקות במשרד המשפטים.

החתום

משרד המשפטים (חתימה מוסדית).