SYSTEM FOR CONTROLLING ACCESS TO A FUNCTION HAVING CLOCK SYNCHRONIZATION

The present invention relates to an electronic system for authentication of individuals and/or messages, in particular for controlling access to a function, enabling a user conditionally to obtain a service or some other provision to be delivered by a specialized service unit associated with the system in question.

More particularly, the invention relates to a system for control of access to, or authentication of messages handled in, a computer or, more generally, a computerized network, use of which is reserved for persons having been duly legitimately entitled. Such networks may serve, for example, to provide all kinds of services entailing a transaction, usually with a monetary consideration, such as television shopping, pay television, home banking, interactive televised games, or also confidential faxes etc.

The known systems of this kind generally include at least one portable electronic device (referred to as a "card" for convenience in the remainder of the present description) and at least one central electronic verification device (referred to as a "server" in the remainder of the description) which is intended conditionally to deliver authorizations for access to the function or service.

Such authorization is delivered only in the event of agreement or matching between a password produced in the card and a password produced in the server. These passwords are obtained by encrypting a variable with the aid of an encryption algorithm using an encryption key, which operations take place simultaneously in the card and in the server.

Usually, the encryption process employs a time parameter, often involving the use of clocks both in the cards and in the servers. In the ideal case, these clocks should be synchronized with one another; synchronization is difficult to achieve in practice, especially if, for obvious reasons, it is sought to install in the cards clocks which do not have very high accuracy.

A system exhibiting the general characteristics stated above is described in U.S. Patent No. 4,800,590. In this system, each card includes a processor intended iteratively to calculate a password with the aid of an encryption algorithm. This latter periodically encrypts a parameter (referred to as the "seed") which is itself dependent on the parameter used for the previous calculation.

In such a system, the clocks of the card and of the server must in principle be synchronous so as to avoid unjustified denials of access (which are not from fraudulent or erroneous requests on the part of the user of the card).

However, whereas the clock of the server may be highly accurate, the same is not the case for the cards. Indeed, since a modest cost in respect of the cards is a paramount design constraint, it is not possible to incorporate in the cards expensive high-quality clocks which would be free of drift. A high level of synchronization between the clocks of the cards and of the server cannot therefore be envisioned in practice.

In U.S. Patent No. 4,885,778, passwords are also periodically calculated during given successive time spans. A time dependent validity range is established in the server with a length much greater than the time span. When an access request is formulated, the server computes as many passwords as there are time spans contained in the time dependent validity range. If a password of the card matches with one of the passwords thus calculated in the server, access is granted. This system also requires a significant amount of calculating work in the server.

U.S. Patent No. 5,361,062 discloses an authentication system which involves transferring to the server, at the moment at which an access request is input, a part of the time value available in the card on the basis of its clock. The example given in the patent is the minutes value of the clock at the moment of access. When the password is transferred, this value will be used in the server which examines whether the seconds value of its clock is less than or greater than 30 seconds. If this value is less, the server takes the chosen minutes value. In the contrary case, the minutes value in the server is increased by one unit.

This process can operate correctly only if the clock of the card is either synchronous with the clock of the server with a predefined tolerance or lags behind the latter. If, on the other hand, the clock of the card leads that of the server, the latter will be unable to find the password calculated in the card during the access request in progress, since the server will not perform iterations until reaching that of the current minute of its own clock. In this case, no access request, even formulated legitimately, will be able to succeed. However, in incorporating low-quality clocks into the cards (which, as described above, is necessary for cost reasons), it is not possible to determine during manufacture in which direction the clock of a card will drift with respect to the clock of the server, particularly since the drift may be caused by wholly unpredictable phenomena such as temperature variations to which the card may be subjected.

Another drawback of the system according to U.S. Patent No. 5,361,062 is that the elemental time unit clocking the password calculation iterations cannot be chosen to be very small, since the smaller the time unit, the greater the calculational burden both for the server and for the cards. Therefore, in practice, this time unit has been fixed at between 1 and 10 minutes. This implies that each password generated in the card remains valid throughout the duration of such a time unit. This impairs the security of the system. A hacker can in fact obtain a password by intercepting it on the link between the card and the server and could then obtain authorization of access, since he has the right password.

EP Patent Application No. 0 324 954 discloses an access control system which involves transmission to the server by the user of a synchronization data item which pertains to the number of access requests made previously. More precisely, if after calculating the password in the card and in the server there is non-agreement, the server displays a data item relating to the access requests which it has made in respect of this card. This data item must be entered into the card by the user in order to ensure that the same number of access requests is recorded in the card and in the server. This prior system has the drawback of involving the user in the synchronization procedure and of disclosing to any hacker the number of access requests made, although this value serves in the

calculation of the password. It would therefore be preferable not to keep it secret.

The object of the present invention is to provide an authorization system which is free of the drawbacks of the prior art discussed above.

In particular, an object of the invention is to provide an authentication system wherein each card can transmit synchronization information to the server so that the latter is able to synchronize on the clock provided in each card, provided relative drifts of the clocks in both the respective cards and the server remain within predetermined limits.

Accordingly, there is provided a system for authentication of individuals and/or messages, including at least one first portable unit personalized for an individual and at least one second verification unit capable of delivering an authentication function authenticating the individual and/or the messages.

The first unit comprises:

first generating means for generating at least a first variable;

first calculating means for producing a first password with the aid of at least a first encryption algorithm for encrypting at least the first variable; and

transmission means for transmitting the first password to the second unit;

The second unit comprises:

second generating means for, in response to an authentication request made by way of a specified one of the first units, generating at least a second variable assigned to this first unit;

second calculating means for producing a second password with the aid of at least one second encryption algorithm for encrypting at least the second variable, the first and second variables being produced in concert, but independently in the first and second units;

means for comparing the first and second passwords; and

delivering means for, in the event of a predetermined consistency of the passwords, delivering the authentication function.

The transmission means is configured so as to transmit with the first password n digits having the least significant weight of the first variable from the first unit to the second unit.

The second unit further comprises:

(a) substituting means for, based on the current value of the second variable, generate a substituted variable wherein n digits having the least significant weight are replaced by the n digits having the least significant weight of the first variable, the substituted variable thereby comprising a replaced first group of n least significant digits and a second group of m highest significant digits; and

(b) third calculating means for

(1) retaining as second variable for the calculation of the second password the substituted variable, if the substituted variable and the current value of the second variable are consistent with at least a first predetermined condition,

(2) if the first predetermined condition is not satisfied, adjusting in the substituted value the second group of m highest significant digits, the adjusting being dependent from a second predetermined condition, the adjusting generating a substituted and adjusted variable, and

(3) retaining as second variable for the calculation of the second password the substituted and adjusted variable.

From these characteristics, it follows that the variables subjected to encryption in the card and in the server can be matched to one another in order to produce passwords which are identical to each other or have a predetermined relationship with one another and to do so during each access request despite the drifting which these variables may exhibit. Similarly to an embodiment disclosed in copending application filed on the same day for "System for Controlling Access to a Function, Using a Plurality of Dynamic Encryption Variables", the calculating means of the second unit can have a different algorithm from that of the calculating means of the first unit, e.g. the algorithm in the second unit can be a decryption algorithm applied to the first password to derive the first variable, generated in the first unit, which is compared in the server with the second variable generated in the second unit.

If the variable is time-based, this matching may require no further iteration in the server. If the variable is the number of authentication requests

formulated by the card, the number of iterations, which could possibly be necessary, would still be limited to a very small value.

BRIEF DESCRIPTION OF THE DRAWINGS

Other characteristics and advantages of the invention will emerge in the course of the description which follows, given merely by way of example and referring to the appended drawings in which:

Figure 1 is a general diagram of an authentication system according to the invention;

Figure 2 is a simplified flowchart illustrating the principle of the running of the operations in a portable unit or card belonging to the system according to the invention, when an access request is processed;

Figures 3, 3A and 3B show a simplified flowchart illustrating the principle of the running of the operations in a central unit or server belonging to the system according to the invention when an access request is processed;

Figure 4 is a partial flowchart illustrating a detail of the operation of the server relating to the variation as a function of the duration of use of a card, of a validity range imposed on a synchronization value of the clocks;

Figure 5 is the flowchart of a first variant of a detail of the operation of the server via which a matching of the system according to the invention to the drifting of the clocks of the cards in the course of their use may be implemented;

Figure 6 shows a second variant of a matching to the drifting of the clocks in the cards;

Figure 7 is a flowchart of a detail of the operation of the server pertaining to a routine for monitoring the magnitude of the drift of the clocks over a predetermined time period;

Figure 8 shows a flowchart of the mode of calculating a key derived from encryption and used for calculating the password.

Represented in Figure 1 is a very simplified diagram of an authentication or access control system according to the invention.

The system is assumed to grant conditional access to a function which in Figure 1 is symbolized with the rectangle 1. The term "function" should be taken in a very broad sense. It designates any authentication function, in

particular an authentication to which access is conditioned by an authorization involving authentification entailing verification of the card with the aid of which the request is formulated, and preferably also identification of the person requesting access to the function so as to ascertain whether his request is legitimate.

The function may be of any nature, for example a function for access to premises, to a computerized network or to a computer, to a transaction of a pecuniary nature (television shopping, home banking, interactive televised game, pay television) etc. The function may also involve the authentication of messages according to the invention.

This so, it may be seen in Figure 1 that the system according to the invention can, according to a particular embodiment, include at least one first unit 2 referred to here as the "card" and at least one second unit 3. It will be noted that the access control system according to the invention can include a large number of first units and one or more second units, but in any event generally a considerably smaller number of second units than first units. The numbers of units 2 and 3 therefore in no way limit the invention.

The first unit or card 2 is preferably portable and personalized so as to be assigned personally to a given user. It takes for example the form of a pocket calculator or credit card and it bears a public identification number 5 indicated diagrammatically in Figure 1. This number can be registered therein unenciphered and be assigned thereto at the time of its initialization. It can also be formed by the name of the user or any other information specific to him. The personal identification number or PIN is secret and is usually memorized by the authorized user of the card. In the preferred embodiment, to use the card, the PIN is entered into the card by the user; the card compares the entered PIN with a stored value. If the entered PIN and the stored value match, the user is granted access to the card; if not, the card will not function to generate a password. The public identification number identifies the particular card with respect to unit 3 from among the plurality of such cards to be used in the overall access control system.

The card 2 includes a keypad 6 intended to allow the entry of information, such as for example the identification numbers already mentioned, as well as various function buttons 7. It also includes a display screen 8 and is furnished with an integrated circuit 9 which includes in particular a duly programmed microcontroller as well as the customary ROM and RAM memories.

The card 2 also includes a communication device 10 making it possible to communicate with the second unit 3 either directly or via a more or less long-distance transmission link. This device can take numerous forms, for example a DTMF telephone link, a device for transmitting data by infrared rays, a so-called "connected mode" device in which the card is inserted into an appropriate reader or any other transmission device well known in the art.

The second unit 3 encompasses firstly interface means which allow for the provision of communication with the card 2. In the embodiment represented in Figures 1 and 3B, these interface means are symbolized with a rectangle 12; they may take numerous forms. They may for example be a specially dedicated reader, but they may also take the form of a computer terminal inserted for example into a network, or a personal computer equipped with an infrared type interface, etc. Their particular feature is that they can communicate in an appropriate form with the card or cards associated therewith.

The interface means 12 can comprise a keypad 13 and a display screen 14 to enable the user to enter information to be communicated to a part 15 of the second unit 3, such as for example passwords or data to be authenticated and relating to the function 1. However, this information and/or these data can be entered in other ways, in particular automatically without manual intervention by the user, for example simply by entering the card into the interface 12 or else via a function button 7 causing an infrared modulated transmission.

The interface 12 communicates with the other part 15 (hereafter referred to as the server) of the second unit 3. This communication symbolized by the connection 16 can be done at short-range or long-range by any appropriate means. The information flowing over this connection is in particular the password to be subject to control in the server 15 and possibly data or messages to be

authenticated and operated on.

The server 15 comprises in particular a processor 17 and a memory or database 18. The processor is capable of conditionally freeing the functions 1, addressed by the access requests formulated by the various cards, it being possible for these functions to be carried out inside the server (as symbolized in Figure 1) or outside. It should be noted that the server 15 generally cooperates with a large number of cards by way of interfaces such as the interface 12.

Figure 2 represents a simplified flowchart of the various operations which unfurl when a request for access to a function is formulated by the user of such a card.

To initiate the procedure, the public identification number 5 must firstly be communicated to the server 15. This operation can be carried out in various ways. For example, it may be communicated directly to the server as soon as the card is inserted into the interface 12. It can be punched directly into the keypad 13 thereof by the user himself, or else into the keypad 6 of the card 2 and be transferred by the link 10. Communication can also be effected via a remote link, such as a telephone line or by Hertzian link.

The user must also input his entitlement by punching, at 20, his personal identification code or PIN code into the keypad 6 of the card. The code punched in is compared at 21 with the PIN code stored in memory in the card. In the event of non-agreement, the card refuses to be used by the user at 22, the user possibly being allotted several consecutive attempts before being faced with final refusal, if they are all abortive.

If on the other hand the PIN code entered and the PIN code in memory agree, the program triggers the operation for calculating the password.

This calculation consists in an encryption with the aid of an encryption algorithm which may be secret or public. In the latter case, it may be an algorithm referred to as DES (Data Encryption Standard) by experts in this art, such DES being a public algorithm using a secret key.

According to the aforesaid copending Patent Application, the algorithm uses input parameters dependent on dynamic variables which, in one embodiment, are three in number. This concept is also used in the embodiment

disclosed in the present application. Among these dynamic variables, a first one is a variable N stored in a register or counter 24 (figure 2) and is a function of the number of access requests made by the card, and a time variable T, stored in a register or counter 25. These variables are each input as 32 bits. They may represent the actual value of the number of access requests or of the time or possibly a value resulting from the application of any mathematical function of these variables. The third dynamic variable is a secret encryption key Kn used by the algorithm to encrypt the two variables just mentioned. Other ways of describing the algorithm in question include that the algorithm generates an output as a function of the current values of Kn, N and T or that Kn is encrypted in accordance with a key comprising a value which is a function of N and T. The encryption key Kn is stored in memory in a register 26 and is updated with each access request as explained below in regard to Figure 8.

The variable to be encrypted T emanates from a time counter 27 which is driven by a base clock 28 of the card whose output is divided in such a way that the counter 27 is incremented with a predetermined temporal resolution, for example every 0.5 seconds. The number contained in the counter 27 is subjected to an extraction operation at 29 through which a predetermined number (8 in the example) of lowest order digits or bits (or more generically the digits in any numeration system including the binary numeration system, the hexadecimal system and so on) are separated therefrom. These bits or digits are discarded and are no longer used for calculating the password. The place value of the digit or bit of immediately higher order as compared with the discarded digits or bits determines a temporal resolution whose corresponding time unit will be referred to as X in what follows, and with which the password can be calculated.

According to one of the aspects of the present invention, there is a provision for making the number of discarded digits adjustable in order to make it possible to choose this resolution which can thus be determined by programming (symbolized by the block 30 in Figure 2), for example on initializing the card as a function of the degree of security which it is wished to assign thereto or possibly in the course of the use of the card. Indeed, the higher the resolution the shorter will be the duration of validity of a password calculated at a given instant. Since a

hacker will thus have less time available to grab and possibly use the password in place of the card's legitimate user, security is strengthened as the resolution is greater.

The number of remaining digits (24 bits in this instance) symbolized at 31 is subjected to a 32-digit complementation operation at 32, through which the corresponding number receives a given number of digits (8 zero value digits in the example) in these higher order place values. The actual number of digits received will of course correspond to the number of digits extracted in extract unit 29.

The resulting number T is the second variable which will be subjected to encryption. It is stored in the register 25. It will be noted that the least significant digit constitutes the time unit X which determines the resolution involved. If the system works with binary notation and if the step of the counter 27 is 0.5" as indicated above by way of example, the time unit may have values of 0.5", 1", 2", 4", 8" etc.

The content of the register 24 representing the number of authentication requests (sometimes referred to as the number of events by experts) made is placed in a register 24'.

The two variables T and N thus produced are subjected to encryption according to a password calculation process shown diagrammatically in the chain-dotted rectangle 33.

This calculation entails a concatenation or any other mathematical and/or logical combination operation at 34 of the two variables N and T from the registers 24' and 25, and then encryption by the algorithm at 35, the latter possibly being a public algorithm (DES) or a secret algorithm. The calculation produces a result A at 36 from which at 37 can be deleted a group of lowest order digits or highest order digit. The number of digits of this group is determined through a predetermined value "A-length" or Y (which is chosen at 56 in the example). This operation produces the value Ac.

At 38, this value Ac is completed with the groups of digits of lowest order LSB(N) and LSB(T) of the values contained in the registers 24' and 25 respectively (in the example 4 digits per variable).

The encryption performed by the operations designated overall by the rectangle 33 produces at 39 a new value of the variable N, e.g. the current value N being increased by one unit. Alternatively, the new value of N could be obtained by increasing the current value by two (or any other number) each time. Also, the number of units of incrementation could vary with each access request. Of course, the incrementation itself and the way of incrementing must be the same in the card and in the server.

The new value of N is used for a calculation operation at 40 intended to compute the new value Kn+1 of the third dynamic variable or secret encryption key. The result of this operation at 40 is substituted for the previous value of Kn in the register 26 so as to be used when a new authentication request is formulated.

The value obtained at 38 is displayed on the screen 8 of the card 2 in order to be communicated to the server 15 for example by punching the word into the keypad 13 of the interface 12. This communication can also be carried out automatically via links 10 and 16 or through any other appropriate means of communication (block 41).

Referring to Figure 3, the running of the operations performed by the second unit 3 in order to process a request for authentication, so as to gain access to function 1, formulated as described above by a card 2, will now be described.

The server 15 holds in its memory or data base 18 a table containing several values specific to each card 2 (see also Figures 4 to 7). Thus, for each of the cards 1, 2.....x of the system, the memory 18 contains in particular the current value Kna of the encryption key and the value of the number of requests made with this card, these variables being produced in concert with (i.e., to agree with or to have a predetermined relationship with, e.g., identity as in the disclosed embodiment) the creation of these same variables in the card. It will be noted in this regard that these variables are never transferred from the card to the server, nor vice versa and therefore never flow over the transmission lines 10 and 16.

When communicating the password composed of the elements Ac, LSB(T) and LSB(N), the user's identification number (user ID) (block 5 of Figure 1) is also transmitted to the server 15 to enable it to extract from its memory the values Kna and Na relating to the card in question (block 50 of Figure 3). The components of the password are then retrieved as symbolized at 51, 52 and 53 respectively.

As regards the number of access requests, it should be noted that there may be a drift between the number counted in the card 2 and that counted in the server 15. Indeed, it may happen that the user of the card 2 formulates an authentication request without completing it, stated otherwise, without for example transferring the password to the second unit 3. This could also be done with ill intent by a hacker keen to disturb the system. It is clear that a disparity such as this between the number accounted for in the card and that counted in the server can only go one way, the excess always appearing in the card.

As moreover the number of requests constitutes one of the variables subjected to encryption in the card and in the server and since moreover in the present embodiment the encryption key depends on this number, a complete access request following one or more uncompleted access requests must give rise in the server to an iterative calculation of the password to compensate for the difference between the numbers accounted on both sides.

The value Na extracted at 54 from the memory 18 is therefore subjected at 55 to a substitution operation during which the value LSB(Na) originating from the memory 18 is replaced by the value LSB(N) originating from the card 2, the rest HSB(N) of the value Na being by assumption the same as that of variable N in the card 2. This operation produces a value symbolized in the block 56 and called Nc. This variable thus comprises a replaced or substituted first group of least significant digits LSB(N) and a group of highest significant digits HSB(N) or value HSB(N).

This value Nc is subjected to a test at 57 in order to determine whether it is strictly less than the value Na extracted from the memory 18. If it is greater than or equal to this value, the test is affirmative and the process passes to a subtraction operation at 58 in order to establish the difference D between the

values Nc and Na. The result determines the number of operations required to "make up" the number of password calculations which the server has to perform in order to tally with the corresponding number in the card. If the difference D is equal to zero, there was no difference between the numbers of requests accounted for respectively in the card and in the server, so that no iteration is then required. In the contrary case, the necessary iterations are performed at 59.

If test 57 culminates in a negative response, the value HSB(Nc) is incremented at 60 by one unit in order to recover the right value of Nc (block 61). This value is then ready at 62 to serve as dynamic variable to be encrypted. It is to be noted that the modification of HSB(N) is done in accordance with the particular mode being used in the card (incrementation of one or two units, etc...) for the derivation process for the new value of variable N performed in module 39 of Figure 2 described above.

The following concrete examples will facilitate understanding this mechanism, the reasoning being followed for convenience under decimal notation. (It will be similar for any other system of notation). It is recalled that N will always be greater than or equal to Na.

If N=Na=22, for example, LSB(N)=LSB(Na) and the test at 57 will be affirmative. D = 0 and no iteration is required and the value Nc is used directly (block 62).

If N=24 and Na=22, LSB(N) transmitted with the password will be 4 and Nc=24. D at 58 will be equal to 2, two iterations will be required in order for Na to become equal to N and Nc likewise equal to 24.

If N=31 and Na=22, LSB(N) transmitted with the password will be 1 and Nc at 56 will be equal to 21. The test at 57 will be negative, since Nc<Na. Consequently, the value HSB(N) will be increased by one unit and Nc becomes equal to 31. D=9 and nine iterations will be required in order for Na to become equal to 31. All the iterations having been performed, D=0 and Nc will have the value suitable for the encryption to be able to be performed (block 62).

It may be noted at this juncture that in all circumstances the number of iterations required will always be relatively low compared with the number of iterations required in the system of the aforesaid U.S. Patent

No. 5,361,062 in which the encryption key is itself a function of time. In the example given hereinabove wherein the value HSB(N) may be incremented by one unit, the maximum allowable difference D is 9 and access will be denied if $D \geq 10$ (in decimal unit). It may be possible to impose a limit on the drifting of the number of access requests accounted for in the server 15 with respect to that enumerated in the card 2. The value D can therefore be subject to a test at 63 to check whether it is greater than a value Dx constituting the maximum allowable difference. If it is observed that this value is exceeded, access will be denied at 64.

If the maximum allowable difference Dx is greater than 9 ($D \geq 10$), it might be necessary to increment the HSB(N) by more than one unit. In such an embodiment, access is not denied at block 86 when the test 85 is negative, but the HSB of Nc is incremented by a further unit as shown by the broken lines 61A in Figure 3A. The difference between the new value of Nc and Na is calculated (block 58) and access is denied if $D > Dx$ (block 63). If $D \leq Dx$, the process continues as disclosed hereinabove until access is granted, if $Aa = Ac$ (block 85), or is denied (block 86) if $Aa \neq Ac$.

It is to be noted that the value HSB(N) may also be modified in other ways than adding one unit thereto in order to recover the value Nc in accordance with the various alternatives for incrementing N discussed above. I.e., this modification may be made in accordance with any mathematical function.

Before describing the process for establishing the time variable in the server, it should be pointed out that the system according to the invention knows a reference time REF-TIME with respect to which all the cards and the server or servers are referenced. This time is for example the instant at which the system is entered into service on the first occasion (for example 1 January 1996 at 00h00'00").

The server 15 includes a base clock 70 or system clock (Figure 3b) which is in principle considerably more accurate than the clocks of the cards 2. A register 71 contains the difference between the position of this clock and the

reference time REF-TIME, this difference being called ΔREF-TIME.

The server establishes at 72 the difference between the value of its base clock and the value ΔREF-TIME, this giving rise at 73 to a server clock value HORL. SERV. established on 32 bits in the present embodiment. This clock value HORL. SERV. exhibits the same temporal resolution as the value HORL. CARTE contained in the counter 27 (0.5" in the example described). On initializing each card, the value HORL. CARTE which is the difference between current time and the value ΔREF-TIME is loaded into the register 27 (Figure 2) of the latter. In other words, if all the clocks were ideal and did not drift, there would at each instant be equality between the values HORL. CARTE and HORL. SERV. of the registers 27 and 73.

In reality, all the clocks have a drift which is larger for those of the cards than for that of the server or servers of the system.

Examined firstly will be the manner in which the system according to the invention is synchronized with each card at the time of an authentication request so as to culminate in a password which is compatible with that calculated in the card during this authentication request, despite the relatively large drift which the clock of the card may exhibit.

It has already been seen that the password calculated in the card is accompanied by a group of digits LSB(T), when it is transmitted to the server. This group of digits (four bits in the example described) constitutes a synchronization information item which is used as follows in the server 15.

The current value HORL. SERV. of the register 73 is subjected to several operations at 74 which are identical to those performed in the card 2 and are shown diagrammatically by the blocks 29 to 32 of Figure 2. They are therefore not detailed in Figure 3B.

The resulting value is symbolized in the block 75. A group of least significant digits LSB(Ta) (four bits in the example; value Ts) is deleted from this resulting value and replaced by the value LSB(T) of block 53, this operation being symbolized by the block 76. This substitution produces a substituted value Tc1 written into the block 77. It therefore includes the synchronization value LSB(T)

sent by the card to the server during transfer of the password Ac, this value constituting a group of least significant digits. The remainder of the digits of the substituted value Tc1 are a group of highest significant digits HSB(Ta).

In its memory 18, the server contains an information item or range Tv (which is Ta-T) for the synchronization value which therefore is subject to a threshold. According to two different calculating protocols, this range Tv is respectively as follows :

$$-5 \leq Ts \leq +4 \text{ (block 78)}$$

or

$$Ta-5 \leq Tc1 \text{ (or } Tc2 \text{ or } Tc3) \leq Ta+4$$

Ts is the difference between the value Ta and the value Tc1 determined in particular by the value (LSB)T originating from the card 2. It should be noted that the range Tv is preferably adjustable, as described below more in details in regard to Figure 4.

The substituted value Tc1 contained in the block 77 is subjected to a synchronization process by which it will be altered, if appropriate, as a function of the range Tv in order to compensate for the drift which the counter or register 27 exhibits with respect to the counter or register 73 of the server 15.

This process begins with a test at 79 to check whether the value Tc1 is compatible with the range Tv. As concerns the first mentioned calculating protocol, this operation includes, firstly, calculating the difference between the values Ta and Tc1, this difference being subjected to a comparison with the first validity range Tv mentioned above.

If the response to the test 79 is affirmative, this means that the difference falls within the range and that there is therefore compatibility with the latter. Indeed, under these conditions, Ta=Tc1, so that Tc1 can be used at 80 for the encryption in the guise of variable Tc (Figure 3A).

In the contrary case, an operation at 81 removes in the value Tc1 one unit from the least significant digit of the group of digits HSB(T) and the result Tc2 (which is a first substituted and adjusted value) is again subjected to a test at 82 to check whether it is compatible with the range Tv. If the answer is yes, it is this value Tc2 which will be adopted at 80 for the encryption in the form of the

variable Tc.

In the contrary case, an operation which is symbolized at 83 increments in the value Tc1 by one unit the least significant bit of the group HSB(T). The result Tc3 (which is a second substituted and adjusted value) is again subjected to a test at 84 to check whether it is compatible with the validity range. If the answer is yes the result is adopted at 80 in the guise of variable Tc in order to perform the encryption. The tests 82 and 84 are similar to the test 79 and they therefore include an operation of calculating the difference between the substituted value Ta and the substituted and adjusted value Tc2 or Tc3 respectively, and then a comparison of the difference with the validity range Tv.

A concrete example will make it possible to understand this synchronization process according to the first protocol which produces the current value Tc which can be used for calculating the password in the server 15. This example is summarized in Table 1 below and is based on decimal notation (Aa being the password calculated in the server (blocks 33a and 36a)):

TABLE 1

Register 25 (Tc) card 2	LSB(T)	Register 75 (Ta) server 15	Calculation tests 79, 82 and 84
...22	2	...25	Tc1 = ...22 Ta-Tc1 = +3⇒ calculate Aa
...19	9	...22	Tc1 = ...29 Ta-Tc1 = -7 Tc2 = ...19 Ta-Tc2 = 3⇒ calculate Aa
...22	2	...18	Tc1 = ...12 Ta-Tc1 = 6 Tc2 = ...02 Ta-Tc2 = 16 Tc3 = ...22 Ta-Tc3 = -4⇒ calculate Aa

It is therefore seen that in this example, the system accepts a

validity range of 10 time units X which is centered on the value Ta which is in fact the content of the register 75. If the time unit X is 128" or 2'8" for example, the validity range extends over 21'20". Supposing that the maximum drift per day of a card is ± 0.5 sec. and the life time of a card is two years, the maximum drift of the card will be ± 6 minutes. Supposing also that the error when initializing the clock counter 27 of the card is ± 1 minute and the error in the server clock 73 with respect to the reference time at any movement is ± 1 minute, the maximum difference at any moment between the card clock and the server clock may be ± 8 minutes. This maximum difference (± 8 minutes) is within the validity range (21'20") mentioned hereinabove. If the allowed validity range is set to be more than 10 times unit X, it would be necessary to perform a procedure similar to that followed for the access reminder N when the maximum allowable difference Dx is > 9. This means that the HSB(T) could be incremented or decremented by more than one unit. Naturally, in that case, if the test performed at 85 on the basis of the incremented value is negative, this test should be repeated on the basis of the decremented value, provided that both the incremented and the decremented value fall within the allowed validity range. It is to be noted further that, like the procedure followed for the access number N, the time related calculation may involve, instead of adding or subtracting one unit to the value HSB, any other modification of that value involving any mathematical function.

It is also to be noted that if the loss of synchronization with respect to either the access number N or the time T exceeds the above-discussed correctable ranges, the card and the server will not generate matching passwords. The card should then be re-initialized before further use.

The validity range (21'20" in the above example) means that the authentication process will not be impaired by the clock drifts indicated by way of example. However, it must be pointed out that the security of the system will be preserved since, in this range, the system may potentially calculate ten different passwords based, upon an access request, on the new time value, which is incremented by the time unit X with respect to the preceding time unit.

Thus, the same password is not kept during the validity range.

Calculation of the password in the server 15 with the aid of the variables Nc and Tc as well as of the encryption key Ka runs in the same manner as in the card by following the operations shown diagrammatically in the rectangle 33 of Figure 2 (block 33a of Figure 3A). This procedure gives rise to the creation of the password Aa at 36a. Aa is subjected to a comparison test at 85 to check whether it is equal to password Ac sent from the card 2 (block 51).

If the test substantiates an inequality, authentication is denied at 86. Otherwise, access to the function 1 is granted. A new value of the number of access requests is calculated at 87 (Na = Nc + 1) and written into the memory 18. The new value of the key Ka is likewise calculated at 88 and written into the memory 18, the two new values replacing the previous ones therein at the addresses corresponding to the card which has just formulated the authentication request. The calculation of the new value of Na and Ka are performed in like manner to the calculation of the new values of N and K in the card, as described above.

The extent of the validity range Tv of relevance above is determined by the temporal resolution, that is to say, by the chosen time unit X (operations 29 to 31 in the card and overall the operation 74 in the server). Thus, for example, if a time unit of 2'8'' is used, the range extends around the value Ta as follows:

$$-10'4'' \leq Ta \leq +8'32''.$$

If the time unit is doubled, the range becomes:

$$-21'20'' \leq Ta \leq +17'4''.$$

Figure 4 represents an alternative embodiment of the invention in which the validity range Tv is made variable as a function of the time of use of each card. To this end, for each of the cards, the memory 18 of the server holds a value representing its date of entry into service or of initialization. This value is extracted therefrom at 93 upon each access request following operation 50 (Figure 3A). The position of the clock 70 of the server 15 is algebraically

combined in 94 with this value and the result is submitted to a test 90 in order to check whether a predetermined period of time (1 year for example) has elapsed since the date of entry into service. As long as the response to the test remains negative, the range Tv established at 78 (Figure 3B) is fixed, at 91, inside relatively small predetermined limits, for example $-3 \leq Ts \leq +2$. If the test turns out to be affirmative, one year has elapsed since the entry of the card into service so that a larger range can be allowed, the clock of the card then having undergone a drift which remains within allowable limits. The validity Tv range is defined with new limits at 92, fixed for example at $-5 \leq Ts \leq +4$. Of course, the limits of the range Tv may be amended in other ways with the aid of values other than those just indicated.

The synchronization process executed in the server 15 upon each access request corresponds to a certain synchronization value which represents the difference Toffset between the clocks of the card 2 and of the server 15 at the time at which the access request is formulated. This value Toffset can be used in the server for calculating the password.

Two other alternative embodiments, represented in Figures 5 and 6, of the system according to the invention make it possible to implement this reset process and the following Table 2 illustrates this process through a numerical example in terms of time units X, with Toffset 1 and Toffset 2 being respectively the drifts for a given authentication request and for an immediately following authentication request, it being also noted that in this example the second protocol and the second range as mentioned above are used (only the lowest order digits of the respective values are indicated):

TABLE 2

Reg 25 Tc card	LSB(T)	Reg 75 Ta server	Toffset 1	Range Tv allowed for Tc	Toffset 2
22	2	23	0	18-27	-1
18	8	23	-1	17-26	-5
15	5	23	-5	13-22	-8
19	9	23	-8	10-19	-4
16	6	23	-4	14-23	-7
11	1	23	-7	11-20	-12
				06-15	

The memory 18 of the server 15 (Figure 5) includes a section in which particular values ΔREF-TIME are recorded for all the cards of the system. These values are updated for each card at the moment at which a request is completed. To do this, the general value ΔREF-TIME (block 71) of the system is combined additively at 100 with the value Toffset of the access request in progress (block 101) and the result of this operation is stored in the memory 18 at an address specific to the relevant card. During a subsequent access request, this value is extracted from the memory 18 and subtracted at 102 from the current value of the clock 70 of the system and the result is placed in the register 73 of the server so as to be used for this access request. This process is run individually for each card upon each access request made therewith.

In the embodiment represented in Figure 6, the value Toffset itself (block 110) observed for a given card during an access request is recorded in the memory 18 of the server 15 at the moment at which this request is completed. It is extracted therefrom for the following access request so as to be combined additively at 111 with the value contained in the register 73 to give rise to the corrected value of the server clock value (block 112). The latter is used during this following access request for the calculation performed in the server 15.

Figure 7 represents another embodiment of the invention making it possible to monitor the movements of the drift and if appropriate, to draw conclusions therefrom as regards the opportuneness of authorizing or denying a subsequent use of the card.

For each card, the memory 18 of the server 15 stores the drift value and the position of the clock 70 of the server which were valid during the previous request. The drift value is used as represented in Figure 6 to give the altered value (block 112). The previous value of the clock 70 is subtracted from the current value of the latter at 113 to give a value ΔT which represents the time which has elapsed since the latest request (block 114). The server 15 calculates the value Tc as described previously in regard to Figure 3B, the corresponding set of calculation operations being summarized in Figure 7 by block 115. Algebraic combination (block 116) of the result of the calculation and of the value of the block 73 then gives the new value of the drift (block 117) observed during the request in progress. This value is combined algebraically with the previous drift value extracted from the memory 18 (block 118) and the result Δoffset (block 119) is compared with the value ΔT obtained at 114 in the course of a test at 120 which checks whether the deviation observed corresponds to predetermined consistency criteria. For example, it is possible to assert that the drift may not exceed a given value during a predetermined time of use or else an anomaly observed by the server 15 leads to rejection of the card at 121, preventing any further access request. The new clock and drift values are recorded in the memory 18 at the addresses of the relevant card if the test at 120 turns out to be affirmative (blocks 122 and 123, respectively) so as to be used for the subsequent request.

Table 3 below shows a concrete example of this drift monitoring, the values being in time units of 64" and the calculation taking into account the time inaccuracies inherent in the system (the values indicate only the lower order digits and the calculation is made according to the second protocol here above):

TABLE 3

Reg 25 Tc	Reg 75 Ta	Drift card 2	Range Tv allowed	Drift n+1	Δ offset	ΔT (sem.)
22	23	0	18-27	-1	-1	3
18	23	-1	17-26	-5	-4	8
15	23	-5	13-26	-8	-3	7
19	23	-8	10-19	-4	+4	7
Detection of anomaly in the direction of drift						
16	23	-4	14-23	-7	-3	8
11	23	-7	11-20	-12	-5	1
Access denied, anomaly in the drift too large						

Figure 7 illustrates yet another embodiment of the invention according to which the value ΔOffset is used to recenter the validity range on the value of the block 73. For example, if the drift is in the negative direction, the range will be $-7 \leq Ts \leq +2$ and in the contrary case it will be $-3 \leq Ts \leq +6$, these values being given here merely by way of example, of course. Thus, the value obtained at 119 in Figure 7 will serve to define the new validity range at 124, it being possible to carry out the matching upon each new request.

Figure 8 shows how the encryption key Kn (Kna) can drift from one request to another depending on the value Nn (Nna) of the number of requests made, with which it can be combined logically, and then encrypted to give rise to the encryption key Kn+1 (Kan+1) used during the next access request.

These operations are performed in concert both in the card 2 and in the server 15. Firstly, the values Nn+1 and Kn (blocks 130 and 131) are subjected at 132 to a logical combination operation, for example an EXCLUSIVE-OR combination, or alternatively to any other mathematical combination. The resulting intermediate variable Z is subjected to encryption at 133 with the aid of a known or public algorithm which may possibly be the same as that used at 33 and 33a (Figures 2 and 3A). The encryption can be performed with the aid of an

encryption key which is preferably the value of the current dynamic variable Kn, although any other secret key Q (block 134) may also be used.

The result of the encryption operation 133 is the new value Kn+1 of the encryption key which will be used during the next access request (block 135).

CLAIMS

I claim:

1. A system for authentication of individuals and/or messages, including at least one first unit (2) personalized for an individual and at least one second verification unit (3) capable of delivering an authentication function authenticating said individual and/or said messages, characterized in that :

(A) said first unit comprises:

first generating means (25 to 32 ; 24, 24') for generating at least a first variable (T, N) ;

first calculating means (33) for producing a first password (A, Ac) with the aid of at least a first encryption algorithm (35) using an encryption key (Kn) for encrypting at least said first variable (T, N) ;

transmission means (10, 12) for transmitting said first password (A, Ac) to said second unit (3) ;

(B) said second unit comprises :

second generating means (70 to 75 ; 18, 54) for, in response to an authentication request made by way of a specified one of said first units (2), generating at least a second variable (Ta, Na) assigned to this first unit;

second calculating means (33a) for producing a second password (Aa) with the aid of at least one second encryption algorithm using an encryption key (Kna) for encrypting at least said second variable (Ta, Na), said first and second variables being produced in concert, but independently in said first and second units (2,3) ;

means (85) for comparing said first and second passwords (Ac, Aa);

delivering means for, in the event of a predetermined consistency of said passwords, delivering said authentication function (1) ;

said transmission means (10, 12) being configured so as to transmit with said first password n digits having the least significant weight [LSB(T) ; LSB(N)] of said first variable (T, N) from said first unit to said second unit;

(C) said second unit further comprises :

a) substituting means (76, 77 ; 55, 56) for, based on the current value of said second variable (Ta, Na), generating a substituted value (Tc1 ; Nc1) wherein n digits having the least significant weight [LSB(Ta) ; LSB(Na)] are replaced by said n digits having the least significant weight [LSB(T) ; LSB(N)] of said first variable (T, N), said second variable thereby comprising a first group of least significant digits and a second group of m highest significant digits;

b) third calculating means (79, 81 to 84 ; 57, 58, 60, 61) for

c) retaining as second variable (Tc ; Nc) for the calculation of said second password (Aa) said substituted variable, if said substituted variable and said current value (Ta ; Na) of said second variable are consistent with at least a first predetermined condition,

d) if said first predetermined condition is not satisfied, adjusting in said substituted value by one unit (81, 83 ; 61) said second group of highest significant digits [HSB(T) ; HSB(N)], said adjusting being dependent from a second predetermined condition, said adjusting generating a substituted and adjusted variable (Tc1, Tc2, Tc3 ; Nc1), and

e) retaining as second variable (Tc, Nc) for the calculation of said second password (Aa) said substituted and adjusted variable.

2. System according to claim 1, wherein

- said first and second generating means (25 to 32 ; 24, 24') are configured so as to generate first and third variables ;

- said second generating means (70 to 75 ; 18, 54) are configured so as to generate second and fourth variables ; and

- said third and fourth variables are submitted to calculating operations similar to those applied respectively to said firs and second variables.

3. System according to claim 2, wherein said first and second units (2, 3) comprise respectively first and second combination means (31, 33a), for combining respectively said first and third variables (T, N) and said second and fourth variables (Ta, Na), using a predetermined mathematical function, before

encryption by said respective first and second encryption algorithms (36, 36a).

4. System according to claim 3, wherein said mathematical function is a concatenation of said first and third variables (T, N) and of said second and fourth variables (Ta, Na) respectively.

5. System according to any of claims 1 to 4, wherein first and second generating means provided in said first and second units (2, 3) are configured respectively so as to generate one of said variables as a function of the number of authentication requests (N, Na) made by said first unit.

6. System according to any of claims 1 to 4, wherein said first and second generating means provided in said first and second units (2, 3) are configured respectively so as to generate one of said variables as a function of time (T, Ta).

7. The system as claimed in any of claims 1 to 6, wherein said third calculating means further comprises:

a) means (57) for implementing said first condition configured so as to determine if said substituted variable (Na) is higher than or equal to said current value (Na) of the respective variable generated in said second unit (3) ; and

b) means (60, 61) for implementing said second condition configured so as to increment by one unit said second group of m highest significant digits [HSB(N)], if said substituted variable (Na) is strictly lower than said current value (Na), thereby generating a substituted and incremented value (Nc1), and for retaining said substituted and incremented value (Nc1) for the encryption with said second encryption algorithm (36a).

8. The system as claimed in any of claims 1 to 6, wherein said third calculation means further comprises:

- means (79) for implementing said first condition configured so as to:

a) calculate a first difference between said current value (Ta) and said substituted value (Tc1) ;

determine if said first difference is within a range (Tv) the extent of which is at the most equal to the number of units having the highest weight p_n

in said first group of n digits, as represented by the digit having the lowest weight p_{n+1} in said group of m highest significant digits [HSB(T)],

b) retaining as variable (Tc) to be encrypted in said second unit (3) said substituted variable (Tc1) if said difference is within said range, and

- means (82, 84) for implementing said second condition configured so as to:

a) adjust (81) by one unit in a first direction said lowest weight p_{n+1}, if said difference is outside said range (Tv) thereby generating a first substituted and adjusted value (Tc2),

b) calculate a second difference between said current value (Ta) and said first substituted and adjusted variable (Tc2),

c) determine if said second difference is within said range (Tv),

d) retain as a variable (Tc) to be encrypted in said second unit (3) said first substituted and adjusted variable (Tc2), if said second difference is within said range,

e) adjust by one unit (82) in a second direction opposed to said first direction said lowest digit p_{n+1} if said second difference is outside said range (Tv), thereby generating a second substituted and adjusted variable (Tc3), and

f) retaining as a variable to be encrypted in said second unit said second substituted and adjusted variable.

9. The system as claimed in any of claims 1 to 6, wherein said third calculating means further comprises:

- means for defining a range of values the extent of which is at the most equal to a number of units having the highest weight p_n in said first group of n digits, as represented by the digit having the lowest weight p_{n+1} in said group of m highest digits [HSB(T)] ;

- means for generating first and second checking values, said checking values being respectively equal to the algebraic sum of said current value and the lowest limit value of said range and said current value and the highest limit of said range;

- means for implementing said first predetermined condition configured so as to:

a) determine if said substituted variable (Ts1) is between said first and second checking values;

b) retain as a variable (Tc) to be encrypted in said second unit said substituted variable (Tc1), if said substituted variable is between said first and second checking values;

- means for implementing said second condition configured so as to:

a) adjust by one unit in a first direction said lowest weight p_{n+1}, if said substituted value is outside said first and second checking values thereby generating a first substituted and adjusted variable (Tc2),

b) determine if said first substituted and adjusted variable (Tc2) is between said checking values;

c) retain as a variable (Tc) to be encrypted in said second unit said first substituted and adjusted variable (Tc2) if said first substituted and adjusted variable is between said checking values;

d) adjust by one unit in a second direction opposed to said first direction said lowest digit p_{n+1} if said first substituted and adjusted variable (Tc2) is outside said checking values thereby generating a second substituted and adjusted variable (Tc3), and

e) retain as a variable (Tc) to be encrypted in said second unit said second substituted and adjusted variable (Tc3).

10. The system as claimed in any of claims 1 to 9, wherein said third calculating means further comprises means for subtracting one of said values (Nc) subjected to encryption by said algorithm in said second unit (3) from said substituted variable (Nc) or, as the case may be, from said substituted and adjusted variable (Nc1), thereby generating a difference value (D), and

said second calculating means further comprising means (59) for iteratively calculating said second password (Aa) as a function of said difference value (D).

11. The system as claimed in any of claims 6 to 10, wherein said first and second generating means further comprises:

a base clock (28 ; 70) for supplying a basic clock signal composed of successive pulses having a predetermined frequency,

counter means (27, 73) for counting said pulses,

digit canceling means (29 ; 74) for canceling a predetermined number of the least significant digits from the contents of said counting means, thereby generating respectively in said at least one unit a first intermediate time value and in said second unit a second intermediate time value, whereby said first and second time dependent variables are generated respectively as functions of said first and second intermediate time values.

12. The system as claimed in claim 11, wherein each of said first and second generating means further comprises adjusting means (30) for regulating said predetermined number of least significant digits to be canceled from the contents of said counting means (27, 73).

13. The system as claimed in any of claims 11 and 12, wherein said first and second generating means each further comprises:

means (32, 74) for adding a predetermined number of digits to said intermediate time values at the side of the highest weight thereof.

14. The system as claimed in any of claims 8 and 9, wherein said second unit further comprises:

storing means (18) for storing the date of initialization of each of said first units (2),

means (90) for, at the time an authentication request is formulated with said at least one first unit (2), comparing the time period elapsed between the date relevant for said at least one first unit (2) and stored in said storing means (18) and at least a predetermined time interval, and

means for increasing said range (Tv) when said elapsed time interval exceeds said predetermined time interval.

15. The system as claimed in any of claims 1 to 11, wherein said third calculating means further comprises:

- means for, upon completion of an authentication request formulated with said at least one first unit, algebraically summing said current value of one of said variables (Ta) generated in said second unit and said substituted variable (Tc1) or, as the case may be, said substituted and adjusted variable (Tc2 ; Tc3) so as to generate an offset value (Toffset),

- means (18) for storing said offset value (Toffset) as a current offset value, said second calculating means being so arranged as to, during at least a subsequent authentication request formulated with said at least one first unit (2) calculate said substituted variable (Tc1), or, as the case may be, said substituted and adjusted value (Tc2, Tc3), as a function of said current offset value.

16. The system as claimed in claim 15, wherein said third calculating means are so arranged as to adapt said first and second conditions as a function of said current offset value (Toffset).

17. The system as claimed in claim 6, wherein said first and second generating means each further comprises:

- a base clock (28, 70) for supplying a basic clock signal composed of successive pulses having a predetermined frequency,

- counter means (27, 73) for counting said pulses, said first and second time dependent variables (T, Ta) being generated respectively as functions of the contents of said counter means,

- storing means (18) so arranged as to store, for each of said units, a compensating value (Toffset NOUVEAU) for compensating the contents of said counter means of said second unit against drift of the base clocks of said first and second units, respectively, and

- said third calculating means further comprising means (115, 116, 117) for generating said compensating value (Toffset NOUVEAU) as a function of said time offset value of said clocks.

18. The system as claimed in claim 17, wherein said third calculating means

further comprises:

first determining means (18, 70, 113) for determining the time interval ΔT between a current authentication request formulated by said at least one first unit and a subsequent authentication request formulated by said unit,

second determining means (118) for determining the difference between the time offset value calculated upon completion of said current authentication request and the time offset value calculated upon completion of said subsequent authentication request,

means (120, 121) for refusing said subsequent authentication request when said difference between said time offset values exceeds a predetermined level as compared with said time interval ΔT as determined by said first determining means.

19. The system as claimed in any of the preceding claims, wherein said first and second calculating means (130 to 135) are so arranged as to calculate said encryption keys used respectively in said at least one first unit and said second unit (2, 3) as dynamic encryption keys (Kn, Kna) varying as a function of corresponding variables (N, Na) respectively in said first and second units.

20. The system as claimed in any of the preceding claims, wherein said first unit (2) is a portable electronics device.