

TITLE: Method for verifying the password of a dongle, associated computer program, dongle and user terminal

The present invention concerns the field of verification of the password of a dongle.

5 The present invention relates to a method for verifying the password of a dongle, comprising, following the connection of said dongle to an associated user terminal, a step of receipt, by a first electronic processing module for processing the dongle, of a password to be validated, entered by a user requesting access to the user terminal.

10 It was known to store the result, H, of a hash function of the password of a dongle beforehand, in the memory of the latter. When a user, carrying this dongle, later connects the dongle to a user terminal and enters the password, this entered password is communicated by the user terminal to the dongle, which applies the hash function to this entered password and compares the result to the hash H in memory.

15 However, if an attacker retrieves the hash value H from the dongle's memory, then he can do an exhaustive search on his computer, searching for the password whose hash value is H.

20 The attacker can also test passwords, until he finds the right password, by exchanging directly with the dongle if he succeeds in disposing of it. It is certainly possible to put a ratification counter in the dongle that blocks any new attempt to provide a password beyond a threshold; but the attacker can also succeed in changing the threshold of the counter.

There is thus a need for a method to verify the password of a dongle with an increased level of security.

25 To this end, according to a first aspect, the invention proposes a method for verifying the password of a dongle of the above-mentioned type, characterized in that it further comprises the following steps implemented by the electronic processing module of the dongle, a memory of the dongle storing a first cryptographic quantity CHIFF_CST_dongle_i equal to the result of the encryption, in confidentiality only, of a first constant CST_dongle_i using a first symmetric key (Ksym_i) cryptographically derived from the password of the
30 dongle, the memory of the dongle also storing the obtaining function for obtaining the symmetric key DERIVE_Ksym(;):

- determination of a second symmetric key Ksym_i' by applying the said obtaining function DERIVE_Ksym(;) of a key symmetric to the received password to be validated;

- obtaining a second constant CST_dongle_i' by decrypting the first cryptographic quantity CHIFF_CST_dongle_i stored in the memory of the dongle using the second determined symmetric key; and
- delivery by the dongle of a set of data, including a data determined according to the second constant CST_dongle_i' to the user terminal, with a view to the validation by the latter of the password entered according to the said set of data and the first constant CST_dongle_i.

The invention thus makes it possible to provide a control solution for the password of a dongle which, by involving the user terminal, significantly reduces the risks that the password of the dongle may be discovered. In embodiments, a method for checking the password of a dongle according to the invention further comprises one or more of the following features:

- said symmetric key obtaining function DERIVE_Ksym(:) is further a function of an identifier of the dongle, and according to which said second symmetric key is determined by the dongle processing module by applying said symmetric key obtaining function to the received password to be validated and to the dongle id_dongle_i identifier stored in memory (32) of the dongle;
- said data, in the data set delivered to the user terminal, which is determined according to the second constant comprises said second constant encrypted using a session key, and said data set is inserted in a specific data field of a message transmitted by the dongle to the user terminal in accordance with a session key establishment protocol;
- said data, in the data set delivered to the user terminal, which is determined according to the second constant, includes an integrity pattern of the second constant calculated using a session key, and wherein said data set is inserted in a specific data field of a message transmitted by the dongle to the user terminal according to a session key establishment protocol;
- the method further comprises the following steps implemented by a second electronic processing module of the user terminal, data showing said first constant having been previously stored in a memory of the user terminal:
 - receipt of the set of data comprising the data determined as a function of said second constant CST_dongle_i' delivered by the dongle;
 - obtaining the first constant as a function of the data showing said first constant previously stored in the memory of the user terminal;

- comparing said data determined as a function of said second constant and a data determined as a function of the first obtained constant;
 - depending on the result of said comparison, selectively validating or not validating the entered password to be validated;
- 5 - the data of said first constant previously stored in the memory of the user terminal comprise at least one cryptographic function `Derive_CST`, and the receipt of the data set comprising said data determined as a function of said second constant `CST_dongle_i` and delivered by the dongle triggers the application, by the second processing module of the user terminal, of the cryptographic function `Derive_CST`
- 10 to at least one element among an identifier of the dongle extracted from the received data set and a system encryption key `Kbase` previously stored in the memory of the user terminal;
- in a step of prior configuration of a set of equipment comprising at least the dongle and the user terminal, the following steps are implemented by an electronic configuration device:
- 15
 - obtaining the first constant associated with the dongle;
 - determining of the first symmetric key according to the symmetric key obtaining function `DERIVE_Ksym` applied to the password of the dongle;
 - determining of the first cryptographic quantity `CHIFF_CST_dongle_i` equal
 - 20 to the result of encrypting in confidentiality only a first constant applying the first symmetric encryption algorithm using the first symmetric key determined;
 - loading in the dongle memory the first cryptographic quantity and the symmetric key obtaining function `DERIVE_Ksym(;;)`;
 - loading the data showing the first constant into the memory (22) of the user
 - 25 terminal (20).

According to a second aspect, the present invention proposes a computer program comprising software instructions which, when executed by a computer, implement the steps, incumbent on the dongle, of a method as defined above. According to a third aspect,

30 the present invention proposes a dongle adapted to be connected to an associated user terminal, said dongle comprising a first electronic processing module adapted to receive a password to be validated entered by a user requesting access to the user terminal when said user terminal is connected to the dongle,

said dongle being characterized in that it further comprises a memory storing a first

35 cryptographic quantity `CHIFF_CST_dongle_i` equal to the result of alone encrypting in

confidentiality a first constant CST_dongle_i using a first symmetric key Ksym_i cryptographically derived from the password of the dongle, the memory of the dongle further storing the symmetric key obtaining function DERIVE_Ksym(;) and in that the dongle's electronic processing module is adapted to determine a second symmetrical key Ksym_i' by applying said obtaining function DERIVE_Ksym(;) for obtaining a symmetrical key to the password to be validated received;

the electronic module for processing the dongle being adapted to obtain a second constant CST_dongle_i' by decrypting the first cryptographic quantity CHIFF_CST_dongle_i stored in the memory of the dongle using the second symmetric key determined;

the dongle being adapted to deliver a set of data, including a data determined as a function of the second constant CST_dongle_i' to the user terminal, with a view to validation by the latter of the password entered as a function of said set of data and of the first constant CST_dongle_i.

According to a fourth aspect, the present invention proposes a set comprising a user terminal and a dongle according to the third aspect of the invention, wherein the user terminal comprises a second electronic processing module and a memory in which data showing the said first constant have been previously stored; and wherein the second electronic processing module is adapted to receive the set of data, when a dongle is connected to said terminal, comprising the data determined as a function of said second constant CST_dongle_i' delivered by the dongle, to obtain the first constant as a function of the data representative of said first constant previously stored in the memory of the user terminal;

said second electronic processing module being adapted to compare said data determined as a function of said second constant and a data determined as a function of the first constant obtained and to, depending on the result of said comparison, selectively validate or not validate the entered password to be validated.

These features and advantages of the invention will appear upon reading the description which follows, given only as an example, and made with reference to the annexed drawings, in which:

[Fig 1] Figure 1 shows a schematic view of an electronic system in an embodiment of the invention;

[Fig 2] Figure 2 is a view of an assembly comprising a user terminal with which a dongle is connected in an embodiment of the invention;

[Fig 3] Figure 3 is a flowchart of steps implemented in a preliminary step in an embodiment of the invention;

[Fig 4] Figure 4 is a flowchart of steps implemented in an operational step in one embodiment of the invention.

5

Figure 1 shows a system 1 in one embodiment. System 1 comprises M user terminals 20, N dongles 30 and an electronic configuration device 10 associated with a database 11.

10 A single dongle 30 and a single user terminal 20 are shown in figure 1; M and N are two integers greater than or equal to 1.

For example, configuration device 10 contains a processor 12 and a memory 13.

For example, configuration device 10 is arranged within a configuration unit for user terminals with secure dongle access.

15 In this particular case, the user terminal 20 is for example a radio terminal adapted to implement ad hoc radio communications with other similar radio user terminals, but the user terminal can be any other user terminal with secure access: rack-mounted fixed computer, laptop, tablet, TV, a personal digital assistant (PDA) or a smartphone, etc.

20 The dongle 30, also known as hardware key or electronic key, is a hardware component whose function is to protect a user terminal against hacking. A dongle 30 typically takes the form of a smart card, or a key; it is adapted to be held by a user, and to be interfaced by this user to a user terminal 20, in a removable and temporary way, when the user wishes to use (specific functions of) this user terminal 20.

In this case, the dongle 30 has a microprocessor 31 and a memory 32.

25 The dongle 30 obviously includes an interface block (not shown) with the user terminal 20 allowing the implementation of exchanges between the dongle and the user terminal.

The user terminal 20 has an access control block 24, a human/machine interface block MMI 25, which in this case has a display and keyboard with numbers instead of letters, a processing block 21 and a connection interface 27 to the dongle.

30 The processing block 21 carries out a set of functionalities delivered by the user terminal (waveform, telephony, data transmission, etc.).

The access control block 24 is adapted to validate a password, or not, entered by a user after connection of the dongle 30 and the user terminal 30.

35 The connection of the dongle 30 with the user terminal 20 is made via a connection interface 27 of the user terminal 20.

Depending on the embodiments, the connection interface 27 has at least one slot in which the dongle is inserted: for example an integrated smart card reader, if the dongle is a smart card, or an input-output port of the user terminal 20 in which the dongle is directly connected (for example, a parallel port, USB port if the dongle is a USB key) or in which a peripheral device, showing a removable user terminal extension such as a smart card reader, is connected, allowing a connection to the dongle 30.

In the particular case considered here, the dongle 30 is a smart card and a user equipment 28, shown in figure 2, comprises the user terminal 20, a peripheral 29 of the smart card reader type connected via a cable 40 in a connector 26 of the user terminal 20 and the dongle 30 which is shown inserted in the peripheral 29, so as to allow exchanges between the user terminal 20, the reader 29 and the dongle 30.

Thus, in this particular case, the connection interface 27 includes the reader 29, the connector 26, the cable 40.

The reader 29 has a human-machine interface, for example a keyboard, containing the letters of the alphabet and numbers, allowing the entry of a password. This man-machine interface of the reader is considered as an extension of the interface MMI 25 of the radio terminal 20. The reader 29 also has a communication interface with the dongle 30 once the dongle is inserted in the reader and a communication interface with the terminal 20.

As it is known, such a dongle 30 allows secure access to the user terminal, or at least to some of the functions accessible from this user terminal using a dongle associated with a password: only a user who both owns dongle 30 and provides the dongle password will be authorized by the user terminal to use F functionalities accessible from the user terminal and provided by the processing block 21.

In one embodiment, the memory 13 contains software instructions, which, when executed on processor 12, automatically implement the steps incumbent on the configuration device 10 described in reference to figure 3.

In one embodiment, the memory 22 contains software instructions which, when executed on processor 23, automatically implement the steps for the access control block 24 described with reference to Figure 4.

In one embodiment, the memory 32 contains software instructions which, when executed on processor 31, automatically implement the steps incumbent on dongle 30 described with reference to Figure 4.

In another embodiment, the configuration device 10 and/or the access control block 24 and/or dongle 30 is implemented as a programmable logic component, such as an FPGA (*Field Programmable Gate Array*), or as a dedicated integrated circuit, such as an ASIC (*Applications Specific Integrated Circuit*).

5

Figures 3 and 4 show the steps of a method for controlling dongle passwords in an embodiment of the invention.

First, in Figure 3, a set of 100 configuration steps for each user terminal 20 and each dongle 30 is first implemented by the configuration system 10. System 1 is thus a mastered system, in that it is adapted to be able to configure user terminals and dongles beforehand.

In a step 101, each dongle 30 $n^o i$, $i = 1$ to N , being identified by an identifier named `id_dongle_i` and stored in its memory 32, the database 11 of the configuration system 10 storing a symmetrical cryptographic key, named `Kbase`, and the `id_dongle_i` identifiers of each dongle $i = 1$ to N , the configuration device 10 calculates for each dongle i a unique pseudo-contingency, denoted `CST_dongle_i`, by applying a cryptographic algorithm, named `Derive_CST(;)`.

The `Derive_CST` algorithm is a function of the cryptographic key, named `Kbase`, and of the identifier, `id_dongle_i`:

20
$$\text{CST_dongle_i} = \text{Derive_CST}(\text{Kbase}, \text{id_dongle_i})$$

The cryptographic function using the cryptographic key `Kbase` is a HMAC function, for example.

25 In a step 102, the configuration device 10 generates (or obtains) a password, noted `mdp_dongle_i`, for each dongle 30 $n^o i$.

Then the configuration device 10 calculates a cryptographic quantity `CHIFF_CST_dongle_i`, using an encryption algorithm named `CHIFF_CST()` in confidentiality only of the `CST_dongle_i` data using a symmetrical key, named `Ksym_i` derived from the password `mdp_dongle_i` (and optionally from the `id_dongle_i` identifier):

30
$$\text{CHIFF_CST_dongle_i} = \text{Derive_CLE}(\text{CST_dongle_i}):$$

wherein `Derive_CLE` is the cryptographic function considered, in confidentiality only.

The encryption algorithm in confidentiality is only, for example, the CTR (Counter) or CBC (Cipher Block Chaining) mode (see NIST Special Publication 800-38A Recommendation for Block 2001 Edition. Cipher Modes of Operation), using for example the AES (Advanced Encryption Standard) type block algorithm; in this case, AES-CTR and
 5 AES-CBC are generally used.

The symmetric key Ksym_i is here a function of mdp_dongle_i (and here optionally of the id_dongle_i identifier): Ksym_i = DERIVE_Ksym_i (mdp_dongle_i; id_dongle_i).

In an embodiment, the Ksym_i function is for example:

- DERIVE_Ksym_i (mdp_dongle_i) = mdp_dongle_i; or
- 10 - DERIVE_Ksym_i (mdp_dongle_i; id_dongle_i) = mdp_dongle_i; id_dongle_i; or
- DERIVE_Ksym_i (mdp_dongle_i) = SHA (mdp_dongle_i); or
- DERIVE_Ksym_i (mdp_dongle_i; id_dongle_i) = SHA(mdp_dongle_i; id_dongle_i).

The configuration device 10 loads in memory 32 of dongle 30 n° i:

- the quantity CHIFF_CST_dongle_i;
- 15 - the algorithmic function DERIVE_Ksym_i;
- the decryption algorithm in confidentiality only, noted DECHIFF_CST(), of the CHIFF_CST_dongle_i data associated with the encryption algorithm in confidentiality only of step 102).

The generation of the password mdp_dongle_i is performed according to alphabet
 20 and size criteria for dongle n°i.

In step 103, the Kbase key, together with the algorithmic function Derive_CST (;) is loaded into memory 24 of each user terminal 20 from configuration device 10 (via a wired or wireless connection), thus configuring the user terminals with this secret system key common to the user terminals.

25 Now suppose that each user terminal 20 and each dongle 30 has been configured beforehand according to the set of steps 100.

With reference to Figure 4, which illustrates the steps implementing the user terminal 20 (and its card reader 29) and dongle 30, let us now consider a user U who has a dongle
 30 30 in his possession, for example dongle 30 n°i, and who wishes to access a set of F functions that are only available, via the user terminal 20, upon possession of a dongle 30 and an access control implementing this dongle.

In step 201, user U inserts dongle 30 in reader 29; following the detection of this connection via reader 29, terminal 20 requires, via MMI 25, that user U enter the dongle
 35 password (at this stage, the user cannot access the F functions). In this case, via the

keyboard of reader 29, the user enters a password, noted here mdp'. The password entered mdp' is transmitted to dongle 30 in step 202.

In a step 203, dongle 30 determines the symmetrical key Ksym_i' according to the password mdp' and the function DERIVE_Ksym_i(;) showing in its memory 32 (and optionally the id_dongle_i identifier): Ksym_i' = Ksym_i(mdp'; id_dongle_i).

Then dongle 30 extracts the quantity CHIFF_CST_dongle_i from memory 32 and decrypts this quantity using the decryption algorithm, in confidentiality only (named DECHIFF_CST and corresponding to the encryption algorithm, in confidentiality only, of step 102) using this symmetrical key Ksym_i' derived from the password mdp_dongle_i':

CST_dongle_i' = DECHIFF_CST(CHIFF_CST_dongle_i)

In an embodiment, the user terminal 20 and the dongle 30 also have a common secret session key Ksession, single-use for example, in their respective memories 22, 32 (dynamically negotiated between them or previously defined). In this case, in a step 204, dongle 30 performs a cryptographic operation by an F() function, for example an encryption or integrity pattern calculation of the quantity CST_dongle_i' with the session key Ksession. The calculated quantity is noted F(Ksession; CST_dongle_i'). In the case of encryption, the data sent is the encrypted one; in the case of an integrity pattern calculation, it is only the calculated pattern that is sent.

In a step 205, dongle 30 delivers, via reader 29, to terminal 20, in addition to its id_dongle_i identifier, the quantity F(Ksession; CST_dongle_i').

On receipt of this data, in step 206, the user terminal 20 determines CST_dongle_i, in this case by calculating CST_dongle_i in turn, from the dongle identifier received, the key Kbase and the cryptographic function Derive_CST(;) which uses this key Kbase (Kbase and the algorithm of the cryptographic function Derive_CST having been stored in its memory 22 in the prior configuration phase):

CST_dongle_i = Derive_CST(Kbase, id_dongle_i)

Then in a step 207, knowing CST_dongle_i and Ksession, the user terminal can check that the value F(Ksession; CST_dongle_i') is consistent. For example, if it is an integrity pattern calculation, the user terminal compares the received data with its own calculation of F(Ksession; CST_dongle_i). Alternatively, if F() is an encryption function, the user terminal decrypts the quantity F(Ksession; CST_dongle_i') and compares CST_dongle_i' to CST_dongle_i.

In step 208, if the quantities compared by terminal 30 in step 207 are equal, then for example a message confirming that the password provided mdp' by the user in step 201 is correct is provided by terminal 30 to the user via MMI 25, for example, by displaying the message on the display of MMI 25 of terminal 20.

5 This validation of the password has the effect that the user then has access to the functions offered by the user terminal 20, such as:

- access to sensitive data stored in User Terminal 20 or accessible via a telecommunications network to which User Terminal 20 belongs;
- access to (part of) a telecommunications network of which User Terminal 20 is a
- 10 part;
- access to specific sensitive commands ...

If the quantities compared in step 207 are not equal, then a message, indicating for example that the password provided mdp' is incorrect is issued by terminal 30 to the user via the display of MMI 25 and user terminal 20 continues to prevent the user from accessing

15 the functions.

Thus, according to the invention, the user terminal plays an active role in validating the password, making it impossible to retrieve the password by simply possessing the dongle: the invention offers a robust solution, based on symmetrical cryptography, with a low memory footprint, to strongly secure access to functions of a user terminal using a

20 dongle associated with a password. According to the invention, the password can also be chosen (i.e., it is not necessarily the result of a mathematical function).

The password mdp_dongle_i is thus not stored in dongle 30, either in decrypted or encrypted form.

An attacker who wishes to retrieve the password can try to intercept the encrypted

25 CHIFF_CST_dongle_i in clear text only in the dongle's memory, but he can do nothing about it, since any password can retrieve a different CST_dongle_i' quantity whose relevance cannot be tested via the dongle alone. According to the invention, an attacker who wishes to retrieve the password of a dongle must necessarily have access to a user terminal to have the validity or invalidity status of a password.

30 In one embodiment, step 205 takes place, for example, during the implementation of a standardized protocol for interactive session key exchange, for example according to the standard "ISO/IEC 11770-3:2015 Information technology - Security techniques - Key management - Part 3: Mechanisms using asymmetric techniques" in which the dongle performs the key confirmation step. In this key confirmation step, the Ksession then

35 corresponds to the symmetric key used in the key confirmation of the standard and the

quantity CST_dongle_i corresponds to the "optional field" that can be added in the key confirmation (this standard suggests the calculation of an integrity check value).

In the embodiment described above, the user terminal and the dongle have a common session key Ksession, for single use: if the attacker has recorded the communications, he can do nothing with it.

In another embodiment, this session key is not single-use.

In another embodiment, the exchange between the user terminal and the dongle at step 205 is not secured by the Ksession: the user terminal and the dongle do not use a session key.

In the embodiment described above, the quantity CST_dongle_i is a cryptographic quantity depending on the system key Kbase and the dongle identifier and the system key Kbase only present in terminals 20 (i.e., not in dongles 30). In another embodiment, CST_dongle_i may or may not be a cryptographic quantity, may or may not depend on a system key and/or the dongle identifier. The user terminal must store the CST_dongle_i quantity or have the means to recalculate it when it has to validate a password entered.

In a simple implementation of the invention, each user terminal 20 has an exhaustive listing ("a truth table") in which the CST_dongle_i quantities of each dongle are written directly; the terminal 20 then does not need to recalculate them by a Derive_CST function.

In this case, the listing might not always be up to date if new dongles are created and the CST_dongle_i quantities do not all have the same value. A procedure to update the listing is therefore to be expected in the fleet of terminals.

Alternatively, the fact that the constants can actually be derived from a cryptographic function using a Kbase and/or id_dongle avoids storing the exhaustive listing and list updates.

In this case, the exchanges between the user terminal 20 and the dongle 30 are implemented on wired telecommunication links, but wireless telecommunication technologies, such as NFC type, can be used.

CLAIMS

1. A method for verification of the password of a dongle (30), comprising, following the connection of said dongle to an associated user terminal (20), a step of receipt, by a first electronic processing module for processing the dongle, of a password to be validated (mdp'), entered by a user requesting access to the user terminal, said method being characterized in that it further comprises the following steps implemented by the electronic processing module (31) of the dongle, a memory (32) of the dongle storing a first cryptographic quantity CHIFF_CST_dongle_i equal to the result of encrypting, in confidentiality only, a first constant CST_dongle_i using a first symmetric key Ksym_i cryptographically derived from the password of the dongle, the memory (32) of the dongle (30) further storing the obtaining function DERIVE_Ksym(;) of the symmetric key:

- determination of a second symmetric key Ksym_i' by applying the said obtaining function DERIVE_Ksym(;) of a key symmetric to the received password to be validated;
- obtaining a second constant CST_dongle_i' by decrypting the first cryptographic quantity CHIFF_CST_dongle_i stored in the memory of the dongle using the second determined symmetric key; and
- delivery by the dongle of a set of data, including a data determined according to the second constant CST_dongle_i' to the user terminal, with a view to the validation by the latter of the password entered according to the said set of data and the first constant CST_dongle_i.

2. The method for verifying the password of a dongle (30) according to claim 1, wherein said symmetric key obtaining function DERIVE_Ksym(;) is further dependent on a dongle identifier, wherein said second symmetric key is determined by the processing module of the dongle by applying said symmetric key obtaining function to the received password to be validated and to the identifier id_dongle_i of the dongle stored in the memory (32) of the dongle.

3. The method for verifying the password of a dongle (30) according to claim 1 or 2, wherein said data in the data set delivered to the user terminal (20), which is determined according to the second constant comprises said second constant encrypted with a session key, and wherein said data set is inserted in a specific data field of a message

transmitted by the dongle to the user terminal in accordance with a session key establishment protocol.

4. The method for verifying the password of a dongle (30) according to claim 1 or 2, wherein said data in the data set delivered to the user terminal (20), which is determined according to the second constant, comprises an integrity check value of the second constant calculated using a session key, and wherein said data set is inserted into a specific data field of a message transmitted by the dongle to the user terminal in accordance with a session key establishment protocol.

5. The method for verifying the password of a dongle (30) according to any of the preceding claims, further comprising the following steps implemented by a second electronic processing module (24) of the user terminal (20), data representatives of said first constant having been previously stored in a memory (22) of the user terminal:

- receipt of the data set comprising the data determined as a function of said second constant CST_dongle_i' delivered by the dongle (30);
- obtaining the first constant as a function of the data showing said first constant previously stored in the memory (22) of the user terminal;
- comparison of said data determined as a function of said second constant and a data determined as a function of the first obtained constant;
- depending on the result of said comparison, selectively validating or not validating the entered password to be validated.

6. The method for verifying the password of a dongle (30) according to claim 5, according to which the data of said first constant previously stored in the memory (22) of the user terminal (20) comprise at least one cryptographic function Derive_CST, and the receipt of the set of data comprising said data determined as a function of said second constant CST_dongle_i' and delivered by the dongle (30) triggers the application, by the second processing module (24) of the user terminal (20) of the cryptographic function Derive_CST to at least one element among an identifier of the dongle extracted from the data set received and a system encryption key Kbase previously stored in the memory (22) of the user terminal.

7. The method for verifying the password of a dongle (30) according to any one of claims 4 to 6, wherein, in a step of preconfiguring a set of equipment comprising at least

the dongle and the user terminal, the following steps are implemented by an electronic configuration device (10):

- obtaining the first constant associated with the dongle;
- determining the first symmetric key according to the symmetric key obtaining function DERIVE_Ksym applied to the password of the dongle;
- determining the first cryptographic quantity CHIFF_CST_dongle_i equal to the result of encrypting a first constant applying the first symmetric encryption algorithm using the first symmetric key determined;
- loading the memory of the dongle with the first cryptographic quantity and the symmetric key obtaining function DERIVE_Ksym(;;
- loading the data representing the first constant into the memory (22) of the user terminal (20).

8. A computer program comprising software instructions which, when executed by a computer, implement the steps incumbent on the first dongle processing module (30), of a method according to any of the preceding claims.

9. A dongle (30) adapted to be connected to an associated user terminal (20), said dongle comprising a first electronic processing module (31) adapted to receive a password to be validated (mdp') entered by a user requesting access to the user terminal when said user terminal is connected to the dongle,

said dongle being characterized in that it further comprises a memory (32) storing a first cryptographic quantity CHIFF_CST_dongle_i equal to the result of encrypting, in confidentiality only, a first constant CST_dongle_i using a first symmetric key Ksym_i cryptographically derived from the password of the dongle, with the memory (32) of the dongle (30) further storing the symmetric key obtaining function DERIVE_Ksym(;; and in that the electronic module for processing the dongle (31) is adapted to determine a second symmetrical key Ksym_i' by applying said obtaining function DERIVE_Ksym(;; for obtaining a symmetrical key to the password received to be validated;

the electronic processing module of the dongle (31) being adapted to obtain a second constant CST_dongle_i' by decrypting the first cryptographic quantity CHIFF_CST_dongle_i stored in the memory of the dongle using the second symmetric key determined;

the dongle being adapted to deliver a set of data, including a data determined as a function of the second constant CST_dongle_i' to the user terminal, with a view to validation by the

latter of the password entered as a function of said set of data and of the first constant CST_dongle_i.

- 5 **10.** An user assembly comprising a user terminal (20) and a dongle (30)
according to claim 9, wherein the user terminal comprises a second electronic processing
module (24) and a memory in which data showing said first constant have been previously
stored; and wherein the second electronic processing module (24) is adapted to, when a
dongle is connected to said terminal, receive the set of data comprising the data determined
as a function of said second constant CST_dongle_i' delivered by the dongle (30), to obtain
10 the first constant as a function of the data representative of said first constant previously
stored in the memory (22) of the user terminal;
said second electronic processing module (24) being adapted to compare said data
determined as a function of said second constant and a data determined as a function of
the first constant obtained and to, depending on the result of said comparison, selectively
15 validate or not validate the entered password to be validated.