

AUTHENTICATION DEVICE AND METHOD

FIELD OF INVENTION

5 The present invention relates broadly to an authentication device and method, for example in the field of the use of encryption for information technology (IT) security and electronic security.

BACKGROUND

10 In the context of IT security, many end user devices, including desktop computers, tablet computers, smart-phones, laptop computers, portable hard disk drives, USB flash drives and various other mobile devices, as well as servers, process and exchange large amount of information in wired as well as wireless environments. Some of this information is highly sensitive, such as private personal information and perpetrate corporation information. Information that can benefit a user or organization can also be used against the user or the
15 organization if it falls into wrong hands. Industrial espionage agents among highly competitive businesses are resorting to electronic means to stealing corporate information.

Encryption is the most pervasive solution for providing data confidentiality. Most data encryption software products install and store the encryption key which is used to encrypt, protect the data inside the same device where the data is stored. If the device is lost or
20 hacked, both the encrypted data as well as the encryption key fall into the same hands and the data security is compromised accordingly.

Using an external portable token which is physically separated from the device storing the data, to store the encryption key for encrypting and accessing secured data, for example, from a virtual secure disk, is a preferred approach to securing sensitive data on a host computer, as
25 it can separate the encryption key from the encrypted data. The communication link between the token and the host computer can be via any communication module or media such as radio frequency (RF) channels, or wired connections. The external portable token can be any peripheral device such as a USB flash drive, a mobile phone or even another computer. The host computer can be any computing machine such as a server, a desktop computer, a portable
30 computer or smart-phones. The host computer can contain a data security manager (DSM).

A password protection mechanism is incorporated together with the token to prevent unauthorized usage of the token. As an example, a user requesting to access the secured data is prompted to plug in his token and key in his password. The correctness of the password is checked. If the password is correct, the token (and hence the user) is allowed to access the
35 secured data. A user is allowed access to the secure data only if the token is plugged in and the

login password is correct. Data are encrypted using, for example, a symmetric key encryption algorithm.

While the above approach can secure sensitive data on a host computer, a problem exists in that a person may forget the password associated with the portable token. While the password typically can be re-set by the manufacturer of the portable token, ensuring that the password is only re-set for the authorized user or owner of the portable token can practically be difficult, if not impossible. For example, even where the manufacturer offers and maintains a product registration website, customers often do not bother with such registration, and therefore this approach does not offer a satisfactory solution. Similar problems exist when the token is damaged, lost or stolen.

SUMMARY

In accordance with a first aspect of the present invention, there is provided an authentication device for authenticating the user or owner of an electronic security device, wherein the authentication device is separate from the security device and is configured for connecting to a computing device via a first communication link for authentication processing.

In accordance with a second aspect of the present invention, there is provided an authentication method for authenticating the user or owner of an electronic security device, the method comprising connecting an authentication device separate from the security device to a computing device via a first communication link for authentication processing.

BRIEF DESCRIPTION OF THE DRAWINGS

Embodiments of the invention will be better understood and readily apparent to one of ordinary skill in the art from the following written description, by way of example only, and in conjunction with the drawings, in which:

Fig. 1a) shows a schematic diagram illustrating an assembly for creating and accessing a secured virtual disk, according to an example embodiment.

Fig. 1b) shows a schematic diagram illustrating a host computer which interfaces with an external portable token of the assembly of Fig.1a, according to an example embodiment.

Fig. 2 shows a schematic diagram illustrating the configuration of the external portable token of the assembly of Fig.1a, according to an example embodiment.

Fig. 3a) shows a schematic diagram illustrating authentication processing for re-setting the token password, according to an example embodiment.

Fig. 3b) shows a flowchart illustrating authentication processing for re-setting the token password, according to an example embodiment.

Fig. 4a) shows a schematic diagram illustrating authentication processing for creating a duplicate token for replacement of a lost, damaged or stolen token, according to an example embodiment.

Fig. 4b) shows a flowchart illustrating authentication processing for creating a duplicate token for replacement of a lost, damaged or stolen token, according to an example embodiment.

Fig. 5a) shows a schematic diagram illustrating re-encrypting access to secured data files, according to an example embodiment.

Fig. 5b) shows a flowchart illustrating re-encrypting access to secured data files, according to an example embodiment.

Fig. 6 shows a schematic diagram illustrating read-only access to the secured virtual disk, according to an example embodiment.

DETAILED DESCRIPTION

Some portions of the description which follows are explicitly or implicitly presented in terms of algorithms and functional or symbolic representations of operations on data within a computer memory. These algorithmic descriptions and functional or symbolic representations are the means used by those skilled in the computer engineering arts to convey most effectively the substance of their work to others skilled in the art. An algorithm is here, and generally, conceived to be a self-consistent sequence of steps leading to a desired result. The steps are those requiring physical manipulations of physical quantities, such as electrical, magnetic or optical signals capable of being stored, transferred, combined, compared, and otherwise manipulated.

Unless specifically stated otherwise, and as apparent from the following, it will be appreciated that throughout the present specification, discussions utilizing terms such as "scanning", "calculating", "determining", "replacing", "generating", "initializing", "outputting", or the like, refer to the action and processes of a computer system, or similar electronic device, that manipulates and transforms data represented as physical quantities within the computer system into other data similarly represented as physical quantities within the computer system or other information storage, transmission or display devices.

The present specification also discloses apparatus for performing the operations of the methods. Such apparatus may be specially constructed for the required purposes, or may comprise a general purpose computer or other device selectively activated or reconfigured by a computer program stored in the computer. The algorithms and displays presented herein are

not inherently related to any particular computer or other apparatus. Various general purpose machines may be used with programs in accordance with the teachings herein. Alternatively, the construction of more specialized apparatus to perform the required method steps may be appropriate. The structure of a conventional general purpose computer will appear from the description below.

In addition, the present specification also implicitly discloses the algorithm of a computer program, in that it would be apparent to the person skilled in the art that the individual steps of the method described herein may be put into effect by computer code. The computer program is not intended to be limited to any particular programming language and implementation thereof. It will be appreciated that a variety of programming languages and coding thereof may be used to implement the teachings of the disclosure contained herein. Moreover, the computer program is not intended to be limited to any particular control flow. There are many other variants of the computer program, which can use different control flows without departing from the spirit or scope of the invention.

Furthermore, one or more of the steps of the computer program may be performed in parallel rather than sequentially. Such a computer program may be stored on any computer readable medium. The computer readable medium may include storage devices such, as magnetic or optical disks, memory chips, or other storage devices suitable for interfacing with a general purpose computer. The computer readable medium may also include a hard-wired medium such as exemplified in the internet system, or wireless medium (for example wi-fi, bluetooth device and the mobile telephone system). The computer program when loaded and executed on such a general-purpose computer effectively results in an apparatus that implements the steps of the preferred method.

The invention may also be implemented as hardware modules. More particular, in the hardware sense, a module is a functional hardware unit designed for use with other components or modules. For example, a module may be implemented using discrete electronic components, or it can form a portion of an entire electronic circuit such as an application specific integrated circuit (ASIC). Numerous other possibilities exist. Those skilled in the art will appreciate that the system can also be implemented as a combination of hardware and software modules.

Fig. 1a) shows a schematic drawing illustrating an assembly 10 for use in encrypting and accessing secured data, for instance in a virtual secure disk, according to an example embodiment. The assembly 10 comprises a portable token 12 for connecting to a host computer via a first communication link for encrypting and accessing the secured data on the host computer, and a portable authentication device 14 associated with the token 12. The authentication device 14 is configured for connecting to a computing device via a second communication link for authentication processing.

Fig. 1b) illustrates a high-level schematic block diagram of a host computer 100 connected with the portable token 12 using communication link 120, according to an example embodiment. The communication link 120 represents the media through which data is communicated between the portable token 12 and the host computer 100 via the token's interface 140 and the computer's interface 130. The communication link includes, but is not limited to, any communication module or media such as radio frequency (RF) channels, or wired connections. The portable token 12 could be any peripheral device such as USB flash drive, a mobile phone or even another computer. The host computer 100 can be any computing machine such as a server, a desktop computer, a smart-phone, a notebook computer or portable computer. The host computer contains a data security manager (DSM) 150, which may be separately installed, e.g. from a CD-ROM or may be installed from the token 12 or through the internet.

Fig. 2 is a high-level schematic block diagram showing some important components internal to the portable token 12 and the authentication device 14 according to an example embodiment. The token 12 and authentication device 14 each contains rewritable nonvolatile memory 200, 201 respectively, an embedded controller (EC) 210, 211 respectively, and the interface 140, 141 respectively to the computer.

The nonvolatile memory 200 contains data items such as password, counter value and token key. The password is used to control user access to the token 12. The counter value can be a small positive integer used to control the number of consecutive failed password retries. In one embodiment, the token key is a random secret key used for file encryption and decryption under a symmetric key encryption algorithm. The nonvolatile memory 201 contains data items such as an authentication key.

Returning to Fig. 1b), before a token 12 is used on a computer system, the DSM 150 needs to be installed and the token 12 needs to be initialised. The token 12 is connected to the computer 100 and a user password for the token 12 is entered. The token key is then generated. Alternatively the token key can be pre-installed at the factory.

The authentication device 14 (Fig. 1a)) has a unique matching authentication key to the token key. In one embodiment, "matching" means "equal", but it is noted that, in different embodiments "matching" can include that one key can be matched to the other using for example an algorithm for transforming one key into the other. The unique authentication key for the authentication device 14 can be pre-stored in the authentication device 14 during production or can be generated and stored during user installation.

In another embodiment, the association or pairing of the token 12 and the associated authentication device 14 may be implemented using unique identifiers in the token 12 and in the authentication device 14 respectively. The pair of unique identifiers can be "equal", but may alternatively be configured such that one unique identifier can be matched to the other using for example an algorithm for transforming one unique identifier into the other. The pair

of unique identifiers may be pre-stored during production or can be generated and stored during user installation

The DSM 150 performs access control to the secured data based on four events: token plug-in, token removal, user login and user logout. A user is allowed access to the secured data only if the token 12 is plugged in and the login password is correct. Otherwise, the secured data is not accessible. During user access to the secured data, the DSM 150 constantly detects if the token 12 is present. If DSM 150 detects that the token 12 is removed from the host computer, the user is immediately denied access to the secured data. Accessibility is only restored upon plug-in of the token 12 and successful user log-in.

Data are encrypted using a symmetric key encryption algorithm in one example embodiment. Encryption of the data is performed by the EC 210 (Fig. 2) inside the token 12.

In the following, it will be described how the authentication device 14 can advantageously be used for authentication processing prior to resetting the password on the token 12, for example for regaining full access to the secured data if the user has forgotten the original password, in a first scenario.

Fig. 3a) shows a schematic diagram illustrating authentication processing using the token 12 and the authentication device 14. In one example, the user presents the token 12 together with the authentication device 14 to a computer 300 at, for example, a service center, a station of the manufacturer, an internet web interface of the manufacturer or an authorized dealer. This computer 300 may also be the user's host computer. The authentication device 14 and the token 12 are connected to the computer 300 via respective communication links 304, 302. The token 12 and the authentication device 14 may be simultaneously connected to the computer 300, or may be connected in sequence, for example where only one communication interface for connecting to the computer 300 may be available. An authentication manager (AM) 306 is running on the computer 300.

As illustrated in Fig. 3b), the user requests to "reset password" at step 350, for example from a menu list presented on the computer screen under the control of the AM 306 running on the computer 300, in response to the plug-in of the token 12 and the authentication device 14. Upon confirmation of the request, the authentication device 14 encrypts its authentication key at step 352, and sends the encrypted authentication key to the token 12, for example via the AM 306, at step 354. Methods such as secret key cryptography or other suitable methods may be used in example embodiments. It is noted that therefore, the AM 306 in this example embodiment advantageously cannot "see" the actual authentication key.

At step 356, the token 12 then decrypts the encrypted authentication key received from the authentication device 14, to check the authentication key at step 358. If the authentication key "matches" the token key stored in the token 12, then the match between the authentication device and the token is successfully established in this example embodiment. The successful establishment of the match is communicated to the AM 306. In one alternative embodiment,

the token 12 can perform encryption of its token key and send the encrypted token key to the authentication device 14 for decryption and checking of the match. As will be appreciated by a person skilled in the art, the final result, i.e. that the match is checked, can advantageously be the same in such an alternative embodiment.

- 5 The AM 306 will then allow the user to re-input a new password for the token 12. The AM 306 will replace the token's old password with the new password, at step 360 or will reset the password to a default password.

Accordingly, only if a person presents the "matching" token key and authentication device, will a reset of the password be allowed. This can advantageously overcome the problems
10 associated with existing solutions described in the background section above. In an alternative embodiment, the AM 306 parses the authentication device 14 and the token 12 to authenticate an association there between. In other words, the AM 306 automatically determines whether the token 12 and the authentication device "belong together". If they do, this serves as evidence that the person presenting the token is the authorized person/owner of the token 12.
15 If not, re-setting of the password will be refused.

In one example, the token 12 and authentication device 14 which belong together both contain an identical unique data string which is very long, say, for example, more than 100 characters, as the token key and matching authentication key respectively. The AM 306 will compare the data strings in the token 12 and in the authentication device 14. When the two strings are the
20 same, a match is established. In another example, the data strings in the token 12 and authentication device 14 that belong together could be different. In this case, the AM 306 will use an authentication algorithm to process the two data strings to establish whether they are matched to each other. Method such as secret key cryptography or other suitable methods may be used in example embodiments.

- 25 If the authentication of the association between the token 12 and the authentication device 14 is successful, the AM 306, either automatically or prompted by user input, executes a password re-setting operation. In an example embodiment, when a match between a token 12 and an authentication device 14 is established, the user will be prompted to confirm that he wants to reset the password to the token 12. If password reset is confirmed, the user will be
30 asked to enter his new password. After the new password is entered, AM 306 will remove the old password from the token 12 and replace it with the newly entered password.

As mentioned before, unique identifiers stored in the token 12 and the authentication device 14 respectively may be used in the checking of the match between the token 12 and the authentication device 14, in the same manner as described for the key matching above with
35 reference to Fig. 3.

In the following, it will be described how the authentication device 14 can advantageously be used for authentication processing prior to creating a duplicate token of the token 12.

With reference to Fig. 4a), in one example, the user presents a “blank” token 15 together with the authentication device 14 to a computer 400 at, for example, a service center, a station of the manufacturer, an internet web interface of the manufacturer or an authorized dealer. A “blank” token 15 is a token which is not associated with a security token (yet), and e.g. does not contain the unique token key, and which allows a new unique token key to be placed inside. It is noted that a normal token can also be converted to a “blank” token by stripping away the token key. For example, the user plugs in the token, inputs the password to show he is the owner, and then remove or replace the token key. Accordingly, a normal token can be used instead of a “blank” token 15 in a different embodiment.

The authentication device 14 and the token 15 are connected to the computing device 400 via respective communication links 404, 402. The token 15 and the authentication device 14 may be simultaneously connected to the computer 400, or may be connected in sequence, for example where only one communication interface for connecting to the computer 400 may be available.

As illustrated in Fig. 4b), the user chooses to create a duplicate token at step 450, for example from a menu list presented on the computer screen under the control of the AM 406 running on the computer 800, in response to the plug-in of the blank token 15 and the authentication device 14. Upon confirmation of the request, the authentication device 14 generates the matched token key from its authentication key and encrypts the token key at step 452.

Methods such as secret key cryptography or other suitable methods may be used in example embodiments.

The encrypted token key is sent to the “blank” token 15, for example via the AM 406, at step 454. The token 15 decrypts the token key received from the authentication device 14 and installs the decrypted token key, at step 456. Accordingly, the “blank” token 15 is now converted into a token that is matched to, i.e. associated with, the authentication device 14.

In an alternative embodiment, the AM 406 on the computer 400 parses the authentication device 14 and generates the token key for the token 15. This token key is matched to the authentication device's 14 authentication key, is then entered into the token 15 by the AM 406. The token 15 and the authentication device 14 now "belong together", i.e. become a matched or associated pair. In this example, the creation of a duplicate token is for the replacement of a damaged token. It may be desirable in this case for the AM 406 to be hosted in an authorised environment.

Where the authentication device 14 also contains a unique identifier for the association with a token, the authentication device 14, or the AM 406 in the alternative embodiment, generate a matching unique identifier for storing in the token 15 in the same way as the token key.

In the examples described above with reference to Figures 3 and 4, it is assumed that the authentication device 14 is kept safe by the user/owner of the token 12, as it is not required for

normal use of the token 12. Accordingly, being in possession of the authentication device 14 can be considered as evidence that the person is "authorized".

In other instances, the user may desire to re-encrypt his secured data which have been encrypted by a token that has been lost. In this next example, the process to re-encrypt the secured data so that a lost token no longer has accessibility to these data is described with reference to Figures 5a) and b), according to an embodiment.

As shown in Fig. 5a), the user connects the authentication device 14 and the token 12 to e.g. the host computer 500, via respective communication links 504, 502. This token 12 may be a new replacement token for a lost token. As illustrated in Fig. 5b) the user enters the token password at step 550, and the password is checked and verified to be correct at step 552.

Optionally, the association between the token 12 and the authentication device 14 may be checked, as described above with reference to Fig. 3.

The AM 506 then accepts the user instruction to change the authentication key and the token key and to re-encrypt all the secured files with the new token key, at step 554.

At step 556, the secured data are decrypted by the authentication device 14 using the current authentication key. The authentication device 14 then generates a new token key and matched authentication key at step 558. The authentication device 14 re-encrypts the data using the new token key at step 560.

The authentication device 14 replaces its old authentication key with the newly generated authentication key at step 562. . The new token key is encrypted by the authentication device and sent to the token 12, for example via the AM 506, at step 564. The token 12 decrypts the received new token key, and replaces the old token key with the new token key at step 566.

The token key and the authentication key have now been replaced with a newer version and the secured files have been re-encrypted with the new key. Accordingly, the lost or stolen key, for which token 12 is the replacement for example, will advantageously no longer be able to open the secured files.

In one alternative embodiment, the token 12 can perform the key replacement and re-encryption functions of the authentication device, as described for the above example embodiment. As will be appreciated by a person skilled in the art, the final result, i.e. that the token key and the authentication key have been replaced with a newer version and the secured files have been re-encrypted with the new key, can advantageously be the same in such an alternative embodiment.

In another alternative embodiment, upon plug-in of the token 12 and the authentication device 14, the user enters the password for the token 12. The AM 506 reads the password input. The AM 506 reads and parses the token's 12 stored password and the unique token key. The result

is checked against the read password input to ascertain that the read password input is correct. If the password input is correct the user is now allowed to proceed to the next stage.

Optionally, the association between the token 12 and the authentication device 14 may be checked, as described above with reference to Fig. 3.

- 5 At the next stage, the user instructs the AM 506 to re-encrypt his secured data files. The AM 506 acknowledges this user's instruction and asks for confirmation. When confirmation is obtained, the AM 506 will generate a new authentication key for the authentication device 14 and a new matching token key for its matched token 12, in this example embodiment. Using the old token key, the AM 506 will decrypt the secured data. It will then re-encrypt the
10 decrypted data using the new token key. Therefore, the lost or stolen token now cannot access these data with its old token key. The AM 506 will write the new authentication key into the authentication device 14 to replace the old one, and the new token key into the token 12 to replace the old one. The token 12 will now be able to access the secured data, using the new token key.
- 15 In another example embodiment, the authentication device 14 can be used to replace the data encryption before creating a duplicate token. Such an embodiment can be readily implemented by a person skilled in the art through the steps already disclosed above.

In the following, it will be described how the authentication device 14 can advantageously be used for read-only access to the secured data without requiring a password according to an
20 example embodiment.

With reference to Figure 11, if the authentication device 14 is plugged into the host computer 1100 via communication link 1104, the DSM 1102 performs read-only access control to the secured data based on authentication device 14 plug-in and authentication device 14 removal. To enable access, the DSM 1102 sends the encrypted data to the authentication device 14 and
25 the EC of the authentication device 14 decrypts the data. During user read-only access to the secured data, the DSM 1102 constantly detects if the authentication device 14 is present. If DSM 1102 detects that the authentication device 14 is removed from the host computer 1100, the user is immediately denied access to the secured data.

In an alternative embodiment, the DSM 1102 reads and parses the authentication key from the authentication device 14. With the authentication key, the DSM 1102 can derive the token key to decrypt the secured data. In one embodiment, the authentication key from the authentication device 14 is equal to the token key. A user is allowed read-only access to the secured data if the authentication device 14 is plugged in. During user read-only access to the secured data, the DSM 1102 constantly detects if the authentication device 14 is present. If DSM 1102
35 detects that the authentication device 14 is removed from the host computer 1100, the user is immediately denied access to the secured data.

In that way, the example embodiments advantageously provide that, should a user forget the password or lose the token, the user is still able to read the data using the authentication device 14.

5 In one embodiment, an authentication device for authenticating the user or owner of an electronic security device is provided, wherein the authentication device is separate from the security device and is configured for connecting to a computing device via a first communication link for authentication processing. The authentication processing can comprise authenticating an association between the security device and the authentication device. The authentication processing can comprise matching a first key and/or a first unique
10 identifier stored in the authentication device with a second key and/or a second unique identifier stored in the security device.

In one embodiment, the authentication device is configured for generating a third key and/or a third unique identifier for configuring another security device as associated with the authentication device.

15 The authentication device can be configured for enabling resetting a password stored in the security device upon successful authentication.

The authentication device can further be configured for connecting to a host computer via a second communication link, for reading secured data stored in the host computer. The authentication device can further be configured for connecting to the host computer via the
20 second communication link, for reading secured data stored in the host computer without requiring a password.

In one embodiment the authentication device is configured for decrypting secured data stored in a host computer, generating a fourth key, and encrypting the data using the fourth key.

25 The computing device can be a host computer for the security device or a computing device residing in one or more of a group consisting of a service center, a station of the manufacturer, an internet web interface of the manufacturer or an authorized dealer. The security device can be a data encryption token.

In one embodiment, an authentication method for authenticating the user or owner of an electronic security device is provided, the method comprising connecting an authentication
30 device separate from the security device to a computing device via a first communication link for authentication processing. The method can further comprise connecting the security device to the computing device via a second communication link, for the authentication processing. The authentication processing can comprise authenticating an association between the security device and the authentication device. The authentication processing can comprise matching a
35 first key and/or a first unique identifier stored in the authentication device with a second key and/or a second unique identifier stored in the security device.

In one embodiment, the method can further comprise resetting a user password stored in the security device upon successful authentication of the association between the authentication device and the security device.

5 The authentication processing can comprise obtaining a third key and/or a third unique identifier from the authentication device for configuring an un-associated security device as associated with the authentication device.

The method can further comprise connecting the authentication device to a host computer via a third communication link for reading secured data stored in the host computer. The reading of the secured data stored in the host computer is without requiring a password.

10 In one embodiment, the method further comprises decrypting secured data stored in a host computer; encrypting the decrypted data using a new key; and storing the new key in the token and a matching authentication key in the authentication device.

The security device can be a data encryption token.

15 It will be appreciated by a person skilled in the art that numerous variations and/or modifications may be made to the present invention as shown in the specific embodiments without departing from the spirit or scope of the invention as broadly described. The present embodiments are, therefore, to be considered in all respects to be illustrative and not restrictive. Also, the invention includes any combination of features, in particular any combination of features in the patent claims, even if the feature or combination of features is
20 not explicitly specified in the patent claims or the present embodiments.

For example, the functionality of the DSM and the AM described in the example embodiments may be implemented in one Security Control Manager (SCM) in different embodiments.

25 Also, while the embodiments have been described in the context of a security token for data encryption, it will be appreciated that the present invention can be applied to different electronic security devices such as electronic secure access cards to buildings or secure areas.

As another example, while in the described embodiments the security token is implemented as a single portable device, it will be appreciated that the present invention can be used in conjunction with different implementations of a security token. For example, the security
30 token can be implemented using one or more interacting devices, which may interact wirelessly and/or wired to perform the desired security functionality. The devices may include one or more of a peripheral device such as a USB flash drive, a mobile phone or any other computing machine/device.

Also, while in the described embodiments the security token is implemented for data
35 encryption using a virtual disk, it will be appreciated that in different embodiments File Encryption, Folder Encryption or Whole Disk Encryption etc. can be implemented.

Claims

1. An authentication device for authenticating the user or owner of an electronic security device, wherein the authentication device is separate from the security device and is configured for connecting to a computing device via a first communication link for authentication processing.
5
2. The authentication device as claimed in claim 1, wherein the authentication processing comprises authenticating an association between the security device and the authentication device.
3. The authentication device as claimed in claim 2, wherein the authentication processing
10 comprises matching a first key and/or a first unique identifier stored in the authentication device with a second key and/or a second unique identifier stored in the security device.
4. The authentication device as claimed in any one of claims 1 to 3, wherein the authentication device is configured for generating a third key and/or a third unique identifier for configuring another security device as associated with the authentication device.
- 15 5. The authentication device as claimed in any one of claims 1 to 4, wherein the authentication device is configured for enabling resetting a password stored in the security device upon successful authentication.
6. The authentication device as claimed in any one of claims 1 to 5, wherein the authentication device is further configured for connecting to a host computer via a second
20 communication link, for reading secured data stored in the host computer.
7. The authentication device as claimed in claim 6, wherein the authentication device is further configured for connecting to the host computer via the second communication link, for reading data stored in the virtual secured disk on the host computer without requiring a password.
- 25 8. The authentication device as claimed in any one of claims 1 to 7, wherein the authentication device is configured for decrypting secured data stored in a host computer, generating a fourth key, and encrypting the data using the fourth key.
9. The authentication device as claimed in any one of claims 1 to 8, wherein the computing device is a host computer for the security device or a computing device residing in one or more of a group consisting of a service center, a station of the manufacturer, an internet
30 web interface of the manufacturer or an authorized dealer.
10. The authentication device as claimed in any one of claims 1 to 9, wherein the security device is a data encryption token.

11. An authentication method for authenticating the user or owner of an electronic security device, the method comprising connecting an authentication device separate from the security device to a computing device via a first communication link for authentication processing.
12. The method as claimed in claim 11, further comprising connecting the security device to the computing device via a second communication link, for the authentication processing.
13. The method as claimed in claim 12, wherein the authentication processing comprises authenticating an association between the security device and the authentication device.
14. The method as claimed in claim 13, wherein the authentication processing comprises matching a first key and/or a first unique identifier stored in the authentication device with a second key and/or a second unique identifier stored in the security device.
15. The method as claimed in any one of claims 11 to 14, further comprising resetting a user password stored in the security device upon successful authentication of the association between the authentication device and the security device.
16. The method as claimed in any one of claims 11 to 15, wherein the authentication processing comprises obtaining a third key and/or a third unique identifier from the authentication device for configuring an un-associated security device as associated with the authentication device.
17. The method as claimed in any one of claims 11 to 16, further comprising connecting the authentication device to a host computer via a third communication link for reading secured data stored in the host computer.
18. The method as claimed in claim 17, wherein the reading of the secured data stored in the host computer is without requiring a password.
19. The method as claimed in any one of claims 11 to 18, further comprising:
- decrypting secured data stored in a host computer;
- encrypting the decrypted data using a new key; and
- storing the new key in the token and a matching authentication key in the authentication device.
20. The method as claimed in any one of claims 11 to 19, wherein the security device is a data encryption token.