

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
24 August 2006 (24.08.2006)

PCT

(10) International Publication Number
WO 2006/088681 A2

(51) International Patent Classification:
G06F 12/14 (2006.01)

(21) International Application Number:

PCT/US2006/004069

(22) International Filing Date: 6 February 2006 (06.02.2006)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
11/060,451 16 February 2005 (16.02.2005) US

(71) Applicant (for all designated States except US): **CY-PRESS SEMICONDUCTOR CORPORATION** [US/US]; 198 Champion Court, San Jose, California 95134-1709 (US).

(72) Inventors; and

(75) Inventors/Applicants (for US only): **LUTTMANN, Eric** [US/US]; 1821 N. Longridge Place, Eagle, Idaho 83616 (US). **MILLER, Jeff** [US/US]; 369 W. River Trail Drive, Eagle, Idaho 83616 (US).

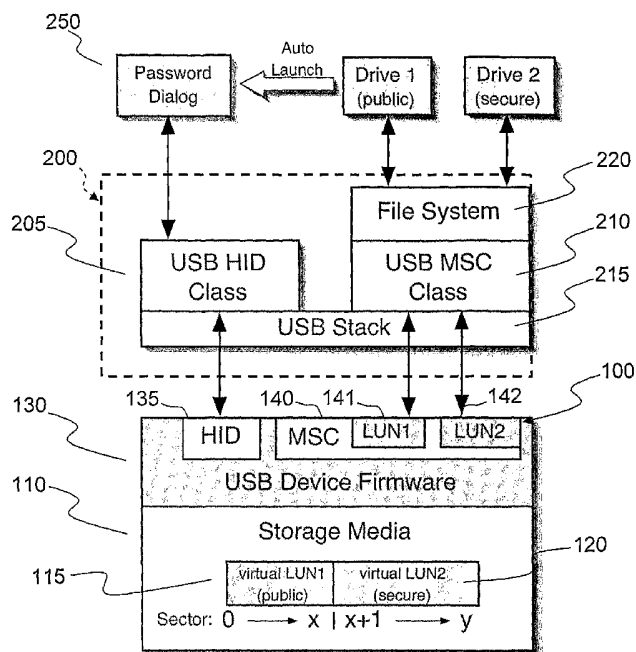
(74) Agents: **FORD, Stephen, S.** et al.; Marger Johnson & McCollom, P.C., 210 Sw Morrison Street, Suite 400, Portland, Oregon 97204 (US).

(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, LY, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SM, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

(84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, LV, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

[Continued on next page]

(54) Title: USB SECURE STORAGE APPARATUS AND METHOD



(57) Abstract: A secure media device preferably includes a Universal Serial Bus (USB) Mass Storage Class (MSC) interface and a USB Human Interface Device (HID) interface. A storage media area is also preferably provided. The storage media is preferably divided into a secure and nonsecure area by arranging the storage media into multiple Logical Units (LUNs). The nonsecure area is preferably accessed in a conventional manner using a host USB MSC driver through the USB MSC interface on the storage device. A password dialog application can be located in the nonsecure area of the storage device. A preferred method of configuring a secured storage device includes dividing storage media of the storage device into a secure and a nonsecure area. A password for accessing the secure area is preferably set using a host USB HID driver and a USB HID interface on the storage device. A preferred method of accessing a secure area of a USB secure storage device includes verifying a password using a host USB HID driver and a USB HID interface of the storage device. Once the password is verified, the secure area is preferably unlocked. After the secure area is unlocked, the storage device preferably responds to a host USB MSC driver poll with an indication that the secure area is

accessible. Until the secure area is unlocked, the storage device preferably responds to a host USB MSC driver poll with an indication that the secure area is inaccessible. The secure area of the USB secured storage device is preferably locked via firmware in response to a removal or other locking event. According to various aspects of this invention, therefore, there is preferably no need for a custom driver or software package to be installed on a host computer to facilitate access to the data contained in the secure media area.



Published:

— without international search report and to be republished
upon receipt of that report

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

USB SECURE STORAGE APPARATUS AND METHOD

5 BACKGROUND OF THE INVENTION

Field of the Invention

The present invention relates to an apparatus and method of providing a secure area on a media storage device, and more particularly an apparatus and method of providing a storage device with a secure section that does not require custom drivers or a specialized software package for accessing data in the secure section.

Description of Related Art

Although conventional technologies provide secure storage areas, these have several disadvantages. One technology uses a software package to create two partitions, a public
15 partition that anyone can access and an encrypted partition that requires installation of specialized drivers and software for accessing the files on the encrypted partition. Another technology, such as is implemented in the USB MSC drivers supplied by Cypress Semiconductor Corporation, uses a custom driver and software application to implement the ATA Security Feature of hard drives to protect access to the entire device using a password
20 that is stored on the hard drive itself.

Unfortunately, there are disadvantages with each of these conventional methods. For example, the technology that provides multiple partitions with one partition being encrypted requires specialized software and drivers to be installed to access files on the encrypted partition. In addition, the secure data has to be encrypted and decrypted to gain access to the files using the specialized software and drivers. Furthermore, unauthorized users could gain access to the actual data in the encrypted partition using low level disk access (sector read/write), thereby providing them with the opportunity to attempt to decrypt the data on the drive.

The technology that permits an entire disk to be protected via hardware also has disadvantages. For instance, this technology also requires specialized software and drivers to be installed in order to unprotect access to the disk. It further provides an all or nothing approach to accessing files on disk, with no separate public and secure areas. Furthermore, the hard drives have to support the ATA security feature for this method to work, and it is therefore not compatible with typical flash memory or ATAPI devices. In addition, once the

hard drive is unprotected, a non-standard mechanism is required to cause the Operating System (OS) file system to re-mount the media and access the drive contents.

SUMMARY OF THE INVENTION

5 According to one preferred embodiment of the present invention, a method is provided that allows portions of a Universal Serial Bus (USB) media device to be secure, and further allows either the presence or absence of unsecured portions. In this preferred embodiment, existing native USB Mass Storage Class (MSC) drivers can be utilized without requiring any custom drivers or software packages to be installed in order to access the secure
10 area on the media.

 A preferred apparatus according to this invention is preferably constructed to include specialized firmware in the USB storage device. A simple, standalone security application is also preferably provided to communicate directly with the USB storage device firmware through a standard Human Interface Device (HID) interface. Security requests are preferably
15 executed through the HID interface.

 Enabling and disabling access to the secure media is preferably done in the firmware by emulating the insertion and removal of media, respectively. Security features, like locking and unlocking the secure media, would therefore preferably be initiated via the simple, standalone security application. The security application would send the enable or disable
20 instruction through the HID interface to the USB storage device. According to this embodiment, therefore, there would be no need for a custom driver or software package to be installed to allow access to the data contained in the secure media area.

BRIEF DESCRIPTION OF THE DRAWINGS

25 The foregoing and additional aspects and advantages of the present invention will become more readily apparent through the following detailed description of preferred embodiments, made with reference to the attached drawings, in which:

 FIG. 1 is a schematic block diagram of a USB secure storage device providing both a secure and a nonsecure section according to a preferred embodiment of the present invention;

30 FIG. 2 is a flow diagram illustrating a method of setting up and sectioning the USB secure storage device of FIG. 1 into a secure area and a nonsecure area according to a still further aspect of the present invention;

 FIG. 3 is a flow diagram illustrating a method of accessing a secure section of the USB secure storage device of FIG. 1 according to another aspect of the present invention; and

FIG. 4 is a flow diagram illustrating a method of preventing access to a secure section of the USB secure storage device of FIG. 1 according to yet another aspect of the present invention.

5

DETAILED DESCRIPTION OF PREFERRED EMBODIMENTS

The principles of the present invention will be described more fully hereinafter with reference to preferred embodiments thereof. It should be noted, however, that these embodiments are provided by way of example, and not by way of limitation, and that the embodiments disclosed herein may be modified in form and detail without departing from the principles and scope of the invention.

As noted previously, FIG. 1 is a schematic block diagram of a USB secure storage device 100 providing both a nonsecure section 115 and a secure section 120 in a storage media area 110, according to a preferred embodiment of the present invention. Referring to FIG. 1, the USB secure storage device 100 includes USB device firmware 130. A host OS 200 preferably includes a USB HID Class driver 205 and a USB MSC driver 210. The host USB MSC driver 210 preferably communicates with a USB MSC interface 140 in the firmware 130 of the USB storage device 100, while the host USB HID Class driver 205 preferably interacts with an HID interface 135 in the firmware 130 of the USB storage device 100. The host OS 200 preferably also includes a USB stack 215 and file system 220 to facilitate communicate between the host OS 200 and the USB storage device 100.

As noted, the USB device firmware 130 preferably includes an HID interface 135 and a MSC interface 140. The MSC interface 140 preferably provides the ability to utilize multiple Logical Units (LUNs) 141, 142 to communicate separately with the secure section 120 and the nonsecure section 115. Most preferably, the size of the nonsecure and secure sections 115, 120 may be user configured by mapping storage areas on the physical media 110 to virtual LUNs corresponding to the secure 120 and nonsecure sections 115.

More particularly, a first physical storage area, including sectors 0 – x, is preferably mapped to a first virtual LUN (virtual LUN1) corresponding to the public or nonsecure storage area 115. A second physical storage area, including sectors x+1 – y, is preferably mapped to a second virtual LUN (virtual LUN2) corresponding to the secure section 120. The standard USB storage LUN mechanism can then be used to present the nonsecure (public) and secure storage areas 115, 120 to a host device.

First Time Use or Operation Without Security Enabled

First time use of the USB secure storage device 100, or operation without the security feature enabled, is preferably similar to use of conventional USB mass storage devices. In particular, the USB Storage device 100 is preferably USB Mass Storage Class (MSC) compliant. Accordingly, in the configuration described above, the host Operating System (OS) USB MSC driver 210 permits communication with the USB Storage device 100. The USB Storage device 100 preferably provides access to the storage media 110 as a single Logical Unit (LUN). The USB MSC driver 210, in this configuration, mounts and interacts with the media 110 in a conventional manner.

The Secure Storage Feature

The primary difference between the USB secure storage device 100 and a conventional USB MSC device, is that the secure storage device 100 additionally presents an USB HID Class interface 135. The USB HID Class driver 205 provided by the Host OS matches to the HID interface 135 of the storage device 100. The secure section 120 can be protected from unauthorized access using the USB HID interface as will be described in further detail below with reference to FIGS 1-4.

FIGS. 2-4 illustrate through flow diagrams preferred methods of configuring and using the secure storage device 100 of FIG. 1. FIG. 2 illustrates a method of setting up and sectioning the USB secure storage device 100 into secure and nonsecure areas. FIG. 3 illustrates a method of accessing a secure section of the USB secure storage device 100. And FIG. 4 illustrates a method of preventing access to a secure section of the USB secure storage device 100.

Enabling and Configuring the Secure Storage Device

Referring specifically to FIGS. 1 and 2, first time setup of the secure storage section 120 is preferably initiated by the end-user. More particularly, a custom secure storage configuration application is preferably launched by the end-user, or can be automatically launched from the USB storage device 100, to create the secure storage section 120. This application preferably communicates with the USB storage device 100 by way of the USB HID driver 205. The application can use the USB HID driver 205 and the USB HID interface 135 of the USB Storage device 100 to send commands to setup the security configuration of the device. This includes setting a security password as well as setting up the size of the public and secured sections 115, 120 of the media 110. The security password

can be stored in non-volatile memory such as NV-RAM, EEPROM, or other flash memory, for example, and can be located, for instance, in the secured section 120 of the media 110.

As noted previously, configuration of the public and secured areas 115, 120 preferably takes place by setting them up as multiple virtual LUNs. The public area 115 is preferably defined by mapping a first physical storage area (e.g., sectors 0 – x) to a first virtual LUN (virtual LUN1). Likewise, the secure area 120 is preferably defined by mapping a second physical storage area (e.g., sectors x+1 – y) to a second virtual LUN (virtual LUN2). The standard USB storage LUN mechanism can then be used to present the nonsecure (public) area 115 and the secure storage areas 120 to a host device in a manner as will be further described below.

Additionally, using the HID Interface 135, later reconfiguration of the secure media area 120 is preferably also able to be performed. This could include, for example, permitting a user to enable and disable the secure media option, permitting a user to define and change the size of public media area 115 and secure media area 120, and allowing a user to change the secure media password. A user could also be allowed to reconfigure the device 100 in case of a lost password, but preferably only after a complete low-level (firmware) wipe of the secure area 120.

Accessing the Secured Media

Referring now to FIGS. 1 and 3, after setting up the secure section 120, the secured USB storage device 100 remains USB MSC compliant. The host OS USB MSC driver 210 therefore matches to the secured USB storage device 100. The USB MSC driver 210 interrogates the secured USB storage device 100. The USB secure storage device 100 reports back to the USB MSC driver 210 that it has multiple removable media drives (representing the nonsecure section 115 and the secure section 120). The USB MSC driver 210 mounts the nonsecure media area 115 in the conventional manner.

Again, the primary difference between the USB secure storage device 100 and conventional USB MSC devices is that the USB secure storage device 100 presents an additional USB HID class interface 135. The USB HID driver 205 provided by the host OS 200 matches to the HID interface 135 of the storage device 100. When the password dialog 250 is launched automatically, or initiated by the end-user, that password application 250 communicates with the USB secure storage device 100 by way of the USB HID driver 205. The dialog preferably uses the USB HID driver 205 and the USB HID interface 135 of the USB Storage device 100 to send the password to unlock the secured “drive” area 120.

The USB secure storage device 100 therefore preferably includes an auto-loading or otherwise easily accessible software application that provides a password dialog to permit access to the secure section 120. In one embodiment, for example, the password dialog application 250 is configured to autorun from the nonsecure section 115 upon insertion of the storage device 100 into the host 200, providing a method to unlock the secure media area 120 without having pre-installed software on the host computer.

Until the password application is run and the appropriate password is entered, when the USB MSC driver 210 polls the secured removable media drive 100, the storage device 100 responds with an indication that the media in the secured drive area 120 is not available. This operation is provided by leveraging the protocol for a Multiple-LUN device with removable media. The USB MSC driver 210 will continue to poll the secure storage device 100 for accessible media and the storage device 100 will continue to respond with an indication that no media is available as long as the secured media section 120 remains password protected.

The firmware 130 acts as a gatekeeper to control access to the media. When the secured area 120 is locked by the firmware 130, it blocks access to the secured area 120 by responding to an access request as if a secured drive had been physically removed. When a correct password has been sent and verified, the secured area 120 is then unlocked. The next time the USB MSC driver 210 polls the secured device 100 for accessible media, the device 100 responds with an indication that the secured media area 120 is now available. The USB MSC driver 210 is then able to mount the secure media area 120 in a conventional manner. Again, the firmware 130 acts as the gatekeeper. Once the correct password is received and the secure media area 120 is unlocked, the firmware 130 allows the host requests to pass through to the secure area 120 of the device 100. For instance, the host 200 queries the secure area 120 as removable media and, once unlocked, receives a reply from the secure area 120, and is then able to mount it. Thus, during locking and unlocking the secure area 120, the firmware 130 doesn't modify the media or the media's response to commands from the host 200. Rather, the firmware 130 preferably either blocks access to the locked media by returning requests without passing them through, or opens the "gate" to permit queries to pass through to the unlocked media.

Blocking Access to the Secured Media

A preferred method of blocking access to the secure media area 120 will now be described in more detail with reference to FIGS. 1 and 4. When the host OS file system 220

ejects the secure media area 120 (or in response to another locking event), the USB device firmware 130 preferably locks the secure media area 120. The secure media area 120 then remains locked until the security application is used to unlock it again. Events causing the firmware 130 to lock access to the secure media area 120 include, for instance, powering up
5 the storage device 100 for the first time, receiving an eject command from the host 200, receiving a lock command from a security application over the HID interface 135, or other selected events. If a user forgets the password, the password could preferably be reset and the media reused, but not until after the firmware 130 completes a low-level wipe of the secured data in the secure media area 120.

10

This invention provides numerous benefits over conventional secure storage systems. Some of the primary advantages of various aspects of this invention include the provision of hardware level secure access while still using the existing USB Mass Storage Class and File System drivers of current operating systems to access the secure data once unlocked. No
15 specialized software or drivers are therefore required to gain access to the secured files. Furthermore, a standard USB HID interface can be used to implement the security interface with the firmware. Therefore, no custom drivers are required to support the interface that initiates the security features.

According to other aspects of this invention, multiple LUNs can be used to distinguish
20 between the public and secure media areas. By leveraging the characteristics of multiple-LUN devices, access to the secure area can be prevented until an appropriate password has been entered. The removable-media mass storage technique can further be utilized to enable and disable access to the secure media area "on-the-fly." Accordingly, a simple, standalone security application is all that is required to unlock access to the secure media area. Also, no
25 encryption and decryption is required to secure the media and there is no risk of unauthorized low level access to the secure media area data. The data remains unavailable until access to the secure media area is unlocked by the firmware on the USB device itself.

Various preferred aspects and embodiments of this invention have been described above. While the principles of this invention have been shown and described with reference
30 to preferred embodiments thereof, it will be understood by those skilled in the art that various changes in form and details may be made without departing from those principles. The invention should therefore be interpreted to encompass all such variations coming within the spirit and scope of the appended claims.

CLAIMS

What is claimed is:

- 5 1. A Universal Serial Bus (USB) storage device, comprising:
a USB Mass Storage Class (MSC) interface;
a USB Human Interface Device (HID) interface; and
storage media accessible through the USB MSC interface.
- 10 2. A storage device according to claim 1, wherein the storage media is arranged into a public area and a secure area as multiple virtual Logical Units (LUNs).
3. A storage device according to claim 2, wherein the USB HID interface is used to unlock the secure area.
- 15 4. A storage device according to claim 2, further comprising firmware configured to lock the secure area in response to a removal event.
5. A storage device according to claim 2, further comprising a password dialog
20 application stored in the public area.
6. A storage device according to claim 5, wherein the password dialog is configured to autorun in response to an insertion event.
- 25 7. A storage device according to claim 5, wherein the password dialog is configured to verify a password and unlock the secure area using a host USB HID driver that communicates with the storage device HID interface.
8. A storage device according to claim 2, wherein the storage device is configured to
30 respond to a host USB Mass Storage Class (MSC) driver interrogation with an indication that the storage device contains multiple drives.

9. A storage device according to claim 8, wherein the storage device is configured to reply to a host USB MSC poll with an indication that the virtual LUN corresponding to a locked secure area is unavailable.
- 5 10. A method of configuring a secure storage device comprising:
configuring a secure area and a nonsecure area within a storage media of the storage device; and
setting a password for accessing the secure area of the storage device using a USB HID interface of the storage device.
- 10 11. A method according to claim 10, wherein configuring a secure area and a nonsecure area comprises configuring a virtual LUN for each of the secure and nonsecure areas.
12. A method according to claim 10, wherein access to the secure area is obtained using
15 the USB HID interface of the storage device and a host USB HID driver.
13. A method according to claim 10, wherein a password dialog application is stored in the nonsecure area of the storage device.
- 20 14. A method of accessing a secure storage device, comprising:
verifying a password using an USB HID interface on a storage device and a host USB HID driver; and
unlocking a secure area of the storage device using device firmware in response to verification of an appropriate password entry.
- 25 15. A method according to claim 14, further comprising responding to a host USB MSC driver poll with an indication that the secure area is accessible after the secure area is unlocked.
- 30 16. A method according to claim 15, further comprising accessing data in the secure area using the host USB MSC driver and a USB MSC interface on the storage device.
17. A method according to claim 14, further comprising responding to a host USB MSC driver poll that the secure area is inaccessible when the secure area is locked.

18. A method of locking a secure area on a secure storage device, comprising:
detecting a locking event; and
locking the secure area using firmware arranged on the storage device in response to
5 the locking event.
19. A method according to claim 18, wherein locking the secure area comprises
configuring the secure storage device to respond to a host USB MSC driver poll that the
secure area is unavailable.
10
20. A method according to claim 18, wherein the locking event is removal of the secure
storage device from a host.

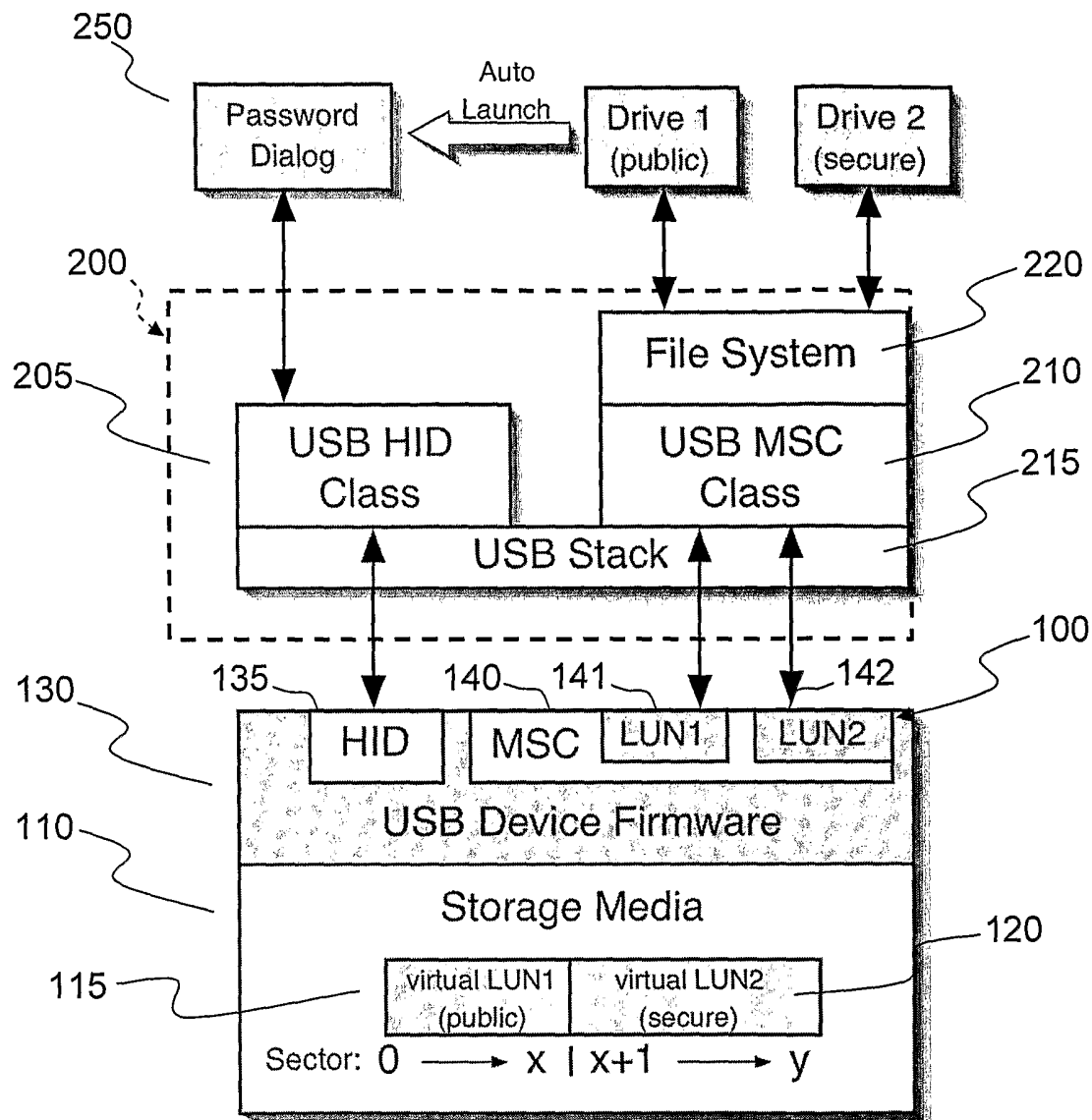
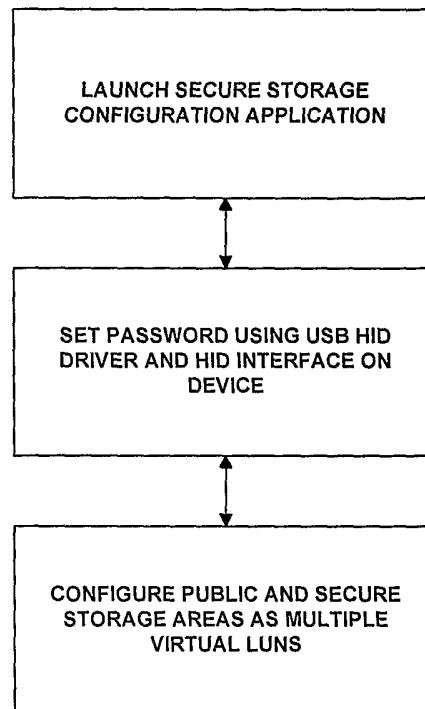
$1/4$ 

FIG. 1

2/4

ENABLING AND CONFIGURING SECURE STORAGE DEVICE

**FIG. 2**

3/4

ACCESSING SECURE MEDIA

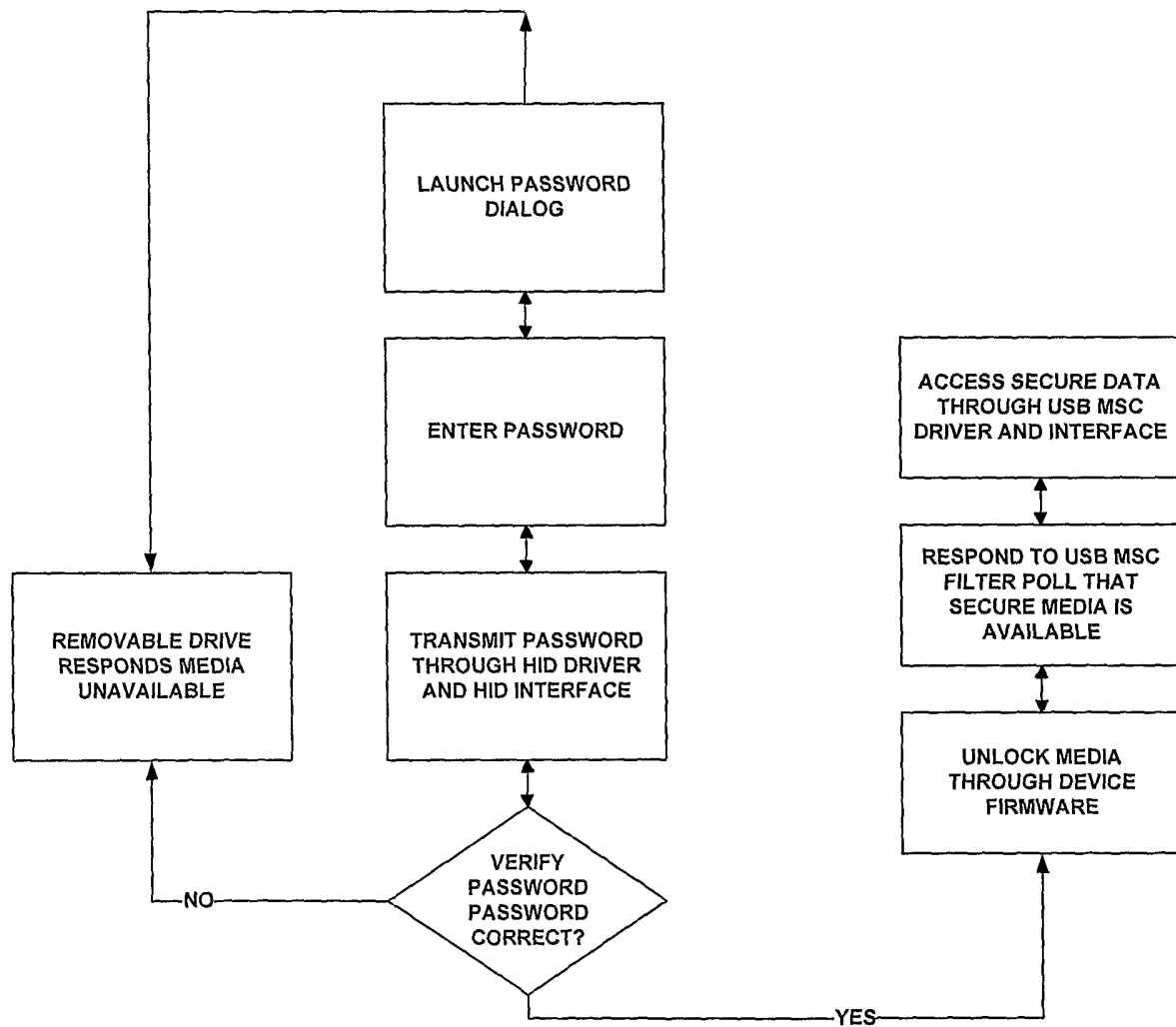
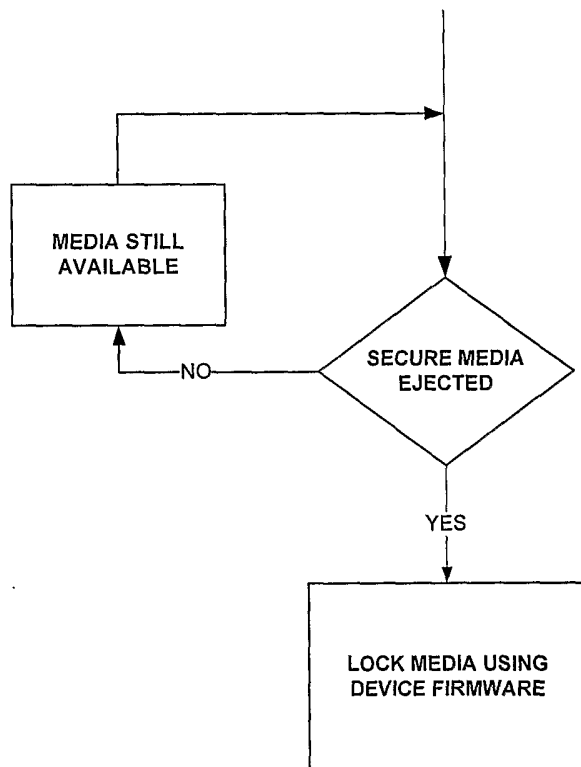


FIG. 3

4/4

LOCKING SECURE MEDIA**FIG. 4**