Abstract: A method for performing double domain encryption is provided. In one embodiment, a memory device receives content encrypted with a transport encryption key. The memory device decrypts the content with the transport encryption key and then re-encrypts the content with a key unique to the memory device. The memory device then stores the re-encrypted content in the memory device.

Title: METHOD FOR PERFORMING DOUBLE DOMAIN ENCRYPTION A MEMORY DEVICE

[Continued on next page]
SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, Published: GW, ML, MR, NE, SN, TD, TG). — *with international search report (Art. 21(3))*
Method for Performing Double Domain Encryption in a Memory Device

Background

[0001] To distribute content to optical discs and other storage devices, a content owner, such as a studio, releases content to a replication facility, which replicates the content onto the storage devices. Since the content owner does not have much control as to what happens in the replication facility, the content owner relies on trust and process control of each particular replication facility to make sure illegal or unauthorized copies of the content are not taking place. Accordingly, content providers do not have precise control over how many copies of the content are being made once the content is released to the replication facility. As a result, content owners do not know if unauthorized copies of the content are being made. Further, content is often delivered to a memory device in encrypted form and stored in the memory device in that encrypted form. Unfortunately, if an unauthorized party gains access to the key used to encrypt the content, the unauthorized party would have access to the content.

Summary

[0002] Embodiments of the present invention are defined by the claims, and nothing in this section should be taken as a limitation on those claims.

[0003] By way of introduction, the embodiments described below generally relate to a method for performing double domain encryption. In one embodiment, a memory device receives content encrypted with a transport encryption key. The memory device decrypts the content with the transport encryption key and then re-encrypts the content with a key unique to the memory device. The memory device then stores the re-encrypted content in the memory device.

[0004] Other embodiments are provided, and each of the embodiments can be used alone or together in combination. Various embodiments will now be described with reference to the attached drawings.

Brief Description of the Drawings

[0005] Figure 1 is a representation of a content replication control system of an embodiment.
[0006] Figure 2 is a flow chart of a method of content replication control of an embodiment.

[0007] Figure 3 is a representation of a memory device of an embodiment for performing double domain encryption.

[0008] Figure 4 is an illustration of a double domain encryption technique of an embodiment.

[0009] Figure 5 is a flow chart of a method for performing double domain encryption in a memory device of an embodiment.

[0010] Figure 6 is an illustration of a content replication control system of an embodiment using a memory device operative to perform double domain encryption.

[0011] Figure 7 is a flowchart of a method of content replication control of an embodiment using a memory device operative to perform double domain encryption.

[0012] Figure 8 is an illustration of a content replication control system of an embodiment using a memory device operative to perform double domain encryption.

Detailed Description of the Presently Preferred Embodiments

[0013] **Introduction**

[0014] The following embodiments provide a method and system for content replication control, as well as a memory device and method for double domain encryption. While these embodiments can be used with one another, it is important to note that the content replication control embodiments can be used with memory devices other than those that provide double domain encryption and that memory devices with double domain encryption can be used with applications other than content replication control.

[0015] The following sections provide a discussion of content replication control, followed by a discussion of a memory device with double domain encryption features and a discussion of content replication control using a memory device with double domain encryption features.
[0016] **Content Replication Control**

[0017] Turning now to the drawings, Figure 1 is a representation of a content replication control system 50 of an embodiment. This system 50 comprises a content replication system 100 in communication with a transport encryption key ("TEK") server 110, a content server 120, and a plurality of memory devices 130. As will be described in more detail below, the content replication system 100, TEK server 110, and content server 120 can be located at the same site as the content replication system 100 (e.g., all three components in one manufacturing center or in a kiosk), or one or both of the TEK server 110 and the content server 120 can be remotely located from the site of the content replication system 100. Further, in some circumstances, the content replication system 100 can also function as the TEK server 110. Also, as will be discussed in more detail below, there can be a connection between the TEK server 110 and the content server 120, where the TEK is requested by the content server 120 based on a replication ID or other information.

[0018] As used herein, "content" can take any suitable form, such as, but not limited to, digital video (with or without accompanying audio) (e.g., a movie, an episode of a TV show, a news program, etc.), audio (e.g., a song, a podcast, one or a series of sounds, an audio book, etc.), still or moving images (e.g., a photograph, a computer-generated display, etc.), text (with or without graphics) (e.g., an article, a text file, etc.), a video game, and a hybrid multi-media presentation of two or more of these forms. A "memory device" can also take any suitable form. In one embodiment, a memory device takes the form of a solid-state (e.g., flash) memory device and can be one-time programmable, few-time programmable, or many-time programmable. However, other forms of memory, such as optical memory and magnetic memory, can be used. In one embodiment, the memory device takes the form of a handheld, removable memory card, an embedded memory card, a universal serial bus (USB) device, or a removable non-removable hard drive, such as a solid-state drive.

[0019] In general, the content replication system 100 is used to replicate content received from the content server 120 onto the plurality of memory devices 130. The content stored in each of the memory devices is received encrypted with a transport encryption key from the TEK server 110, and any authorized memory device needs this
transport encryption key in order to decrypt and use the content. (Even thought the TEK is called a "transport" encryption key, it should be noted that the content could be ciphered before transport with that key.) In this embodiment, each memory device is associated with a respective unique identifier, and the content replication system 100 supplies a given memory device with the TEK needed to decrypt the content only if the unique identifier of the memory device is authorized to receive the TEK. (In some embodiments, the unique identifier is part of a certificate, and the TEK is securely received (e.g., ciphered using a public key from the certificate or loaded with a secure channel resulting from authentication involving that a certificate).) This connection between memory device identifier and TEK allows a content owner to have precise control over how many copies of content are being made once the content image is released to the replication facility. As compared to replication techniques that rely on trust and process control of each particular replication facility to make sure illegal or unauthorized copies of content are not taking place, these embodiments provide content owners with precise replication control of their content.

[0020] As shown in Figure 1, the content replication system 100 of this embodiment comprises user input device(s) 140 (e.g., a keyboard, a mouse, etc.) and a display device 150, through which a user can input and review data to initiate a content replication session. Although shown as separate components, the user input device(s) 140 and the display device 150 can be integrated, such as when the display device 150 takes the form of a touch-screen display. The user input device(s) 140 and the display device 150 are in communication with a controller 160. In one embodiment, the content replication system 100 takes the form of a computer with a WinXP card reader.

[0021] In this embodiment, the controller 160 comprises a central processing unit ("CPU") 163, a crypto-engine 364 operative to provide encryption and/or decryption operations, read access memory (RAM) 365, and read only memory (ROM) 366. The controller 160 also comprises memory device interface(s) 161, which contain the necessary hardware and/or software for placing the controller 160 in communication with the plurality of memory devices 130. (As used herein, the phrase "in communication with" could mean directly in communication with or indirectly in communication with through one or more components, which may or may not be shown or described herein.)
For example, the memory device interface(s) 161 can contain the physical and electrical connectors to simultaneously host the plurality of memory devices 130, or it can contain the physical and electrical connectors to host a separate card reader, which can simultaneously host the plurality of memory devices 130. The controller 160 further comprises server interface(s) 162, which contain the necessary hardware and/or software for placing the controller 160 in communication with the TEK server 110 and the content server 120. For example, the server interface(s) 162 can contain one or more network jacks.

Figure 2 is a flow chart 200 of a method of content replication control using the content replication system 100 of Figure 1. First, the content replication system 100 receives a request to replicate content in the plurality of memory devices 130 (act 210). This request can be received from a user via the user input device(s) 140 and can contain, for example, a replication session ID, a manufacture ID, the title of the content to be replicated, and the number of memory devices to receive the content.

As mentioned above, in this embodiment, each memory device is associated with a respective unique identifier, and the content replication system 100 provides a given memory device with the TEK needed to decrypt the content only if the unique identifier of the memory device is authorized to receive the TEK. It is this connection between memory device identifier and TEK that allows a content owner to have precise control over how many copies of content are being made once the content image is released to the replication facility. Acts 220 and 230 relate to the process of providing a memory device with a TEK, when appropriate. Specifically, for each of the plurality of memory devices, the content replication system 100 sends a request to the TEK server 110 for the TEK (act 220). The request including the unique identifier of the memory device. In one embodiment, the unique identifier of the memory device is passed through authentication (mutual or otherwise), although other mechanisms can be used. The TEK server 110 would then determine if the unique identifier present in the request is authorized by the content owner to receive the TEK. If the unique identifier is not authorized, that memory device will not receive a TEK and, therefore, not be able to decrypt the content. However, if the unique identifier is authorized to receive the TEK, the content replication system 100 will receive the TEK and send it to the memory device
As mentioned above, acts 220 and 230 would be performed for each memory device in the plurality of memory devices 130. These acts can be performed for each memory device one-at-a-time, or the TEK can be sent to all memory devices in parallel, e.g., if the content replication system 100 is a certified device with the credentials to authenticate to the TEK server 110 and to memory devices, such as when the content replication system 100 generates a secure channel key to all memory devices in order to broadcast a TEK encrypted by the secure channel key (e.g., using a parallel replication machine for gang programming).

[0024] Either before or after the authorized memory devices receive the TEK, the content replication system 100 receives content encrypted with the TEK from the content server 120 (act 240) and sends the encrypted content to the plurality of memory devices 130 (act 250). If a memory device did not receive a TEK (because it was not authorized to receive a TEK), that memory device will not be able to decrypt the content. It is because of this that these embodiments provide a "best of both worlds" situation. A content owner can ensure that only authorized memory devices receive content by establishing a point-to-point secure connection with the content server 120 for the duration of the loading of the content into an authorized memory device. However, this approach would be costly and impractical because of the relatively-long time needed to load the content into memory devices in a serial fashion. Because these embodiments use a point-to-point secure connection only to load a TEK based on a unique identifier that is bound to the memory device, the content owner can achieve precise content control over how many copies of content will be made without paying the cost (financial and time) of providing point-to-point loading for the size of the content. Further, because the distributed content is encrypted with a closely-controlled TEK, the content itself can be distributed in a broadcast fashion - even to unauthorized memory devices - because only those memory devices that have the TEK will be able to decrypt and use the content.

[0025] As will be discussed in more detail below, if the memory device receiving the TEK and the content encrypted with the TEK is capable of performing double domain encryption, after receiving the TEK and the encrypted content, the memory device can decrypt the encrypted content with the TEK, re-encrypt the content with a key unique to
the memory device, and store the re-encrypted content in memory. As used herein, a key "unique" to the memory device can be a key that is purposefully chosen to be truly unique, so as not to be used by other memory devices in a set. A key can also be "unique" to the memory device if the key is a value that is randomized by the memory device (or randomized by another entity and delivered to the memory device). Such a randomized value can be considered "unique" as that term is used herein even if it is theoretically possible that another memory device can generate the same random value.

[0026] Before turning to a discussion of the use of double domain encryption in content replication control, the following section discusses an exemplary memory device that is capable of performing double domain encryption. As mentioned above, it is important to note that this exemplary memory device can be used in applications other than those related to content replication control.

(0027) Memory Device with Double Domain Encryption

[0028] Returning to the drawings, Figure 3 is an illustration of an exemplary memory device 300 that is operative to perform double domain encryption. As noted above, while this memory device 300 finds particular use in content replication control embodiments, this memory device 300 can be used in applications that are unrelated to content replication control. Accordingly, to the extent that the claims herein are directed to a memory device or a method for use therewith, details of the content replication control embodiments should not be read into those claims unless those details are explicitly recited in those claims. As will be discussed in more detail below, "double domain encryption" is a process by which data is ciphered with one key, deciphered, and then enciphered with another key (e.g., on the fly while the data is being received). The key to re-cipher the data could be generated by the memory device. Double domain encryption keeps distribution of content simple where the content could be ciphered once and received as a regular file, and where it is distributed with a unique storage key, thus decreasing the value of attacking the storage CEK. It should be noted that, at any given time, the content is ciphered only by one key (either the TEK or the CEK).

[0029] As shown in Figure 3, the memory device 300 comprises a controller 310 and a memory 320. The controller 310 comprises a memory interface 311 for interfacing with the memory 320 and a host interface 312 for interfacing with the host 350. (The host 350
can be the content replication system 100 of Figure 1 or can be another device, such as, but not limited to, a dedicated content player, a mobile phone, a personal computer, a game device, a personal digital assistant (PDA), a kiosk, a set-top box, and a TV system.) The controller 310 also comprises a central processing unit (CPU) 313, a crypto-engine 314 operative to provide encryption and/or decryption operations (the crypto-engine 314 can be implemented in hardware or software), read access memory (RAM) 315, read only memory (ROM) 316 which stores firmware for the basic operations of the memory device 300, and a non-volatile memory (NVM) 317 which stores a device-specific key used for encryption/decryption operations. In this embodiment, the memory device 300 takes the form of a handheld, removable memory card (or hard drive) that can be interchangeably used in a wide variety of host devices. However, other form factors can be used, such those used for a USB device or a solid-state drive.

The memory 320 can take any suitable form. In one embodiment, the memory 120 takes the form of a solid-state (e.g., flash) memory and can be one-time programmable, few-time programmable, or many-time programmable. However, other forms of memory can be used. In this embodiment, the memory 320 comprises a public partition 325 that is managed by a file system on a host and a hidden protected system area 335 that is internally managed by the controller 310. The hidden protected system area 335 stores firmware (FW) code 342 which is used by the controller 310 to control operation of the memory device 300, as well as a transport encryption key (TEK) 344 and a content encryption key (CEK) 346, which will be described below. (In an alternate embodiment, one or both of the TEK 344 and CEK 346 can be stored in the NVM 317.)

The public partition 325 and the hidden protected system area 335 can be part of the same memory unit or can be different memory units. The hidden protected system area 335 is "hidden" because it is internally managed by the controller 310 (and not by the host's controller) and is "protected" because objects stored in that area 335 are encrypted with the unique key stored in the non-volatile memory 317 of the controller 310. Accordingly, to access objects stored in that area 335, the controller 310 would use the crypto-engine 314 and the key stored in the non-volatile memory 317 to decrypt the encrypted objects. Preferably, the memory device 300 takes the form of a secure product
from the family of products built on the TrustedFlash™ platform by SanDisk Corporation.

[0032] The public partition 325 of the memory stores protected content files 330A, 330B. The content 330A, 330B can be preloaded, side-loaded, or downloaded into the memory 320. While the public partition 325 of the memory 320 is managed by a file system on the host, objects stored in the public partition 325 (such as the content files 330A, 330B) may also be protected by the memory device 100. In this embodiment, both stored content files 330A, 330B are protected by respective content encryption keys 340 stored in the hidden protected system area 335, and those keys 340 are themselves protected by the memory-device unique key stored in the non-volatile memory 317 of the controller 310. Accordingly, to unprotect one of the protected content files (say, content file 330A), the crypto-engine 314 would use the memory-device unique key stored in the non-volatile memory 317 of the controller 310 to decrypt the appropriate content encryption key 340 and then use the decrypted content encryption key 340 to decrypt the protected content 330A.

[0033] The memory device 300 and a host (e.g., a server) can communicate with each other via a host interface 312. In one embodiment, for operations that involve the secure transfer of data, the crypto-engine 314 in the memory device 300 and the crypto-engine in a server can be used to mutually authenticate each other and provide a key exchange. The mutual authentication process calls for the server and memory device 300 to exchange unique certification IDs. The server and the memory device 300 can perform a mutual authentication based on PKI, where each memory device has a unique certificate ID. After mutual authentication is complete, it is preferred that a session key be used to establish a secure channel for communication between the memory device 350 and the server. It should be noted that single authentication can also be performed, where a server authenticates the memory device in order to load the TEK. This saves time for each memory device given that the memory device is blank and does not care to validate the server. A secure session key can be generated after single-side authentication.

(0034) The controller 310 can be implemented in any suitable manner. For example, the controller 310 can take the form of a microprocessor or processor and a computer-readable medium that stores computer-readable program code (e.g., software or
firmware) executable by the (micro)processor, logic gates, switches, an application
specific integrated circuit (ASIC), a programmable logic controller, and an embedded
microcontroller, for example. Examples of controllers include, but are not limited to, the
following microcontrollers: ARC 625D, Atmel AT91SAM. Microchip PIC18F26K20,
and Silicon Labs C8051F320. Examples of various components that can be used in a
controller are described in the embodiments discussed herein and are shown in the
associated drawings. The controller 310 can also be implemented as part of the memory
320 control logic.

[0035] As mentioned above, in this embodiment, the crypto-engine 314 in the memory
device 300 is capable of performing double domain encryption. The "double domain" in
"double domain encryption" refers to the transport domain (the encryption used to protect
the content during transmission to the memory device 300) and the storage domain (the
encryption used to protect the content when it is stored in the memory device 300).
Figure 4 illustrates the concept of double domain encryption and will be discussed in
conjunction with the flow chart 500 of Figure 5.

[0036] First, content (data) encrypted with a TEK is received from a host 400 (act
510). This data is ciphered with the transport domain in that the content is encrypted
with the TEK to protect the content during transmission from the host 400 to the memory
device 300. When the content is received at the memory device 300, the crypto-engine
314 in the controller 310 of the memory device 300 decrypts the content with the TEK
344 stored in the memory device 300 (act 520). This converts the content from the
transport domain to clear content. (The transport domain uses the TEK 344 to cipher data
coming into or going out of the memory device 300.) The crypto-engine 314 then takes
the clear content and re-encrypts it with a key unique to the memory device, here the
CEK 346 (act 530). This decryption and re-encryption can take place on-the-fly as the
content is being received by the memory device 300. This places the content in the
storage domain. (The storage domain uses the CEK 346 to cipher data written into or
read out of the flash memory 320.) The memory device 300 then stores the data ciphered
in the flash storage domain in the memory (the flash storage) 320 (act 540).

[0037] Double domain encryption enables host/memory device transfer of encrypted
data without actually encrypting the channel between them while still achieving memory-
device-unique content encryption for storage. This enables the host and memory device 300 to pass data securely between them without having to encrypt the whole session and to achieve uniquely-encrypted content stored in the flash memory 320. In one embodiment, an API that uses this feature is called by an "open stream command," which is available only when the memory device 300 is not engaged in a secure session. The open stream command sets up a security services module for data stream transfers to read or write data. This command determines the characteristics of the data stream and whether to read or write data with or without domain information along with other required data. In one embodiment, one of the arguments in this command specifies the domain for flash encryptions, while another specifies the domain for host/memory device data transport encryption.

[0038] As mentioned above, a memory device capable of performing double domain encryption finds particular use with the content replication control embodiments described above. Consider, for example, the situation in which content encrypted by the TEK is stored in the memory device instead of being re-encrypted. In this situation, if an unauthorized party were somehow able to obtain the TEK, that party would have unauthorized access to the content stored in the memory device. By using double domain encryption, a memory device effectively "changes the lock" on the received content, since the stored content would be protected with a different key from the one that protected the content during transport. Accordingly, with double domain encryption, even if an unauthorized party were somehow able to obtain the TEK, that party would not be able to access to the content because the content would no longer be protected by the TEK. This provides an additional layer of content replication control, which may be desired by content owners.

[0039] It is important to note that a memory device with double domain encryption can be used in applications other than those relating to content replication control. One of the reasons to use a setup such as "double domain" is to pass a secret/valuable object between two authenticated parties without resorting to a strenuous encryption and secure channel method. A secure channel, which encrypts every piece of information going back and forth between two parties, may take a lot of resources to implement, slow down applications, and consume considerably more power from hosts, such as cell-phones.
Double domain alleviates these concerns because it is used to secure specific objects and not the entire communication line. Also, instead of using a single key for all objects being transferred, several different keys can be used for different objects to be transferred. Additionally, there can be multiple entities on one end and a single entity on the other end, separating the users on a single authenticated communication line.

[0040] In an alternate embodiment, double domain can be used with an SSL session where the content/data is stored protected with a first key and delivered to the other party with SSL using another key. Similarly, the content could also be delivered with SSL and stored with another key using double domain. The case where the content is delivered with SSL and where the content is stored as-is and the SSL session key is saved for later use may not be practical if (a) it is too computer intensive to handle so much ciphered data and (b) content provider requirements require content to be kept protected.

[0041] **Content Replication Control Using a Memory Device with Double Domain Encryption**

[0042] As noted above, the double domain embodiments described in the previous section find particular use with the content replication control embodiments described above. This section provides several examples of how these embodiments can work together.

[0043] Returning to the drawings, Figure 6 is an illustration of a system of content replication control of an embodiment using a memory device operative to perform double domain encryption. As with the system 50 shown in Figure 1, this system comprises a content replication system 600, a TEK server 610, and a content server 620. In this embodiment, these components are placed in communication with each other over the Internet. Also in this embodiment, both the TEK server 610 and the content server 620 are remotely located from the site of the content replication system 600. As explained above and as will be further illustrated below, different arrangements can be used. The operation of this system will now be described in conjunction with the flow chart 700 in Figure 7.

[0044] As shown in the flow chart 700, an operator would initiate a replication session by entering information, such as replication session ID, manufacture ID, title of the content, and the number of memory devices (here, memory cards) to be replicated into
the content replication system 600 (act 705, 710). The content replication system 600 and the TEK server 610 then mutually authenticate each other (acts 715, 720). (As noted above, single authentication can also be used.) In this embodiment, authentication and a secure session are established with each memory device to receive content in order to provision the TEK directly to the memory device, and the content replication system 600 facilitates a secure channel and provides a secure pipe of communication between the TEK server 610 and the memory device 130 (act 725). Here, the content replication system 600 never knows of any of the credential information (secrets). The content replication system 600 only facilitates the communications channel. Once authentication takes place, commands and vital data are encrypted and are not sent in the clear.

[0045] The TEK server 610 then provides a replication-session unique TEK (e.g., a AES 128 TEK) directly and securely into each authenticated memory device via a secure session with memory device controller 640. The TEK server 610 can also log the memory device's unique certificate ID to eliminate duplication and for other uses. Next, the content server 620 synchronizes with the TEK server 610, with the content server 620 retrieving the TEK based on the replication session ID and validating the content loading right of the target memory card and manufacturer against a database and rights policy (act 735). This act can be triggered by the content replication system 600 during, before, or after TEK loading. The TEK server 610 then provides the replication-session-unique TEK to the content replication system 600 to send to the memory device 630 (act 730). Upon receipt, the memory device controller 640 encrypts and stores the TEK in memory 650 (act 740). The memory device 630 will then acknowledge completion of the TEK load process to the content replication system 600 (act 745). The content replication system 600 then sends the replication session ID and the requested content title to the content server 620 (act 750) and receives the content title encrypted with the TEK from the content server 620 (act 755). As noted above, these acts can be done in parallel to act 735. The sequence of acts performed by the content server 620 and the TEK server 610 can be interchangeable as long as the authenticity is validated and the replication session and its TEK are assigned. After this, the content replication system 600 parallel programs the content into multiple memory cards (act 760). The crypto-engine 645 of each memory card then performs double domain encryption by first decrypting the
content title with the preloaded TEK (act 770), then re-encrypting the content title with
the pre-generated CEK (e.g., the storage encryption key randomized by the memory
device), and then storing the re-encrypted content in memory 650 (act 775). (While the
memory is shown as being NAND memory in Figure 6, any type of storage technology
can be used, and the memory can be a separate device from the controller 640 performing
the double domain encryption.) As discussed above, domain double prevents third
parties from cloning an image from one memory device to another since each memory
device would have its own content encryption key that makes the image unique. In this
embodiment, both the TEK and CEK are encrypted in memory 650 and integrity
protected, so those items cannot be changed.

[0046] There are many different alternatives that can be used with these embodiments.
For example, while the TEK server 610 and the content server 620 were both remotely
located from the site of the content replication system 600, the location of these
components can vary. This alternative is shown in Figure 8, with the TEK server 810
being located at the same site as the content replication system 800, while the content
server 820 is remotely located. This alternative also includes a replication management
server 815, which receives the TEK server ID, the session ID, and the memory device
IDs from the TEK server 810 for processing and coordination with the content server
820. Otherwise, the operation of the memory device 830, controller 840, and memory
850 is as described above.

[0047] In another alternative, instead of the TEK server being located at the same site
as the content replication system or remotely located from the site of the content
replication system, the TEK server can be located in the content replication system. For
example, the content replication system can double as the TEK server if the content
replication system is certified and trusted to do so. If this is the case, the TEK can be
loaded into memory devices in parallel with a single TEK protected by the same
encryption key provided to all memory devices by the content replication system.
Controlling the TEK provides the ability to control and log devices that can be loaded
with usable content. This is a desirable element of the production log for content
providers.
It is intended that the foregoing detailed description be understood as an illustration of selected forms that the invention can take and not as a definition of the invention. It is only the following claims, including all equivalents, that are intended to define the scope of the claimed invention. Finally, it should be noted that any aspect of any of the preferred embodiments described herein can be used alone or in combination with one another.
What is claimed is:

1. A method for performing double domain encryption, the method comprising:
   performing the following in a memory device:
   (a) receiving content encrypted with a transport encryption key;
   (b) decrypting the content with the transport encryption key;
   (c) re-encrypting the content with a key unique to the memory device;
   and
   (d) storing the re-encrypted content in the memory device.

2. The method of Claim 1 further comprising receiving the transport encryption key from a transport encryption key server.

3. The method of Claim 2, wherein the transport encryption key is received over a secure channel.

4. The method of Claim 1, wherein the content is received over an open channel.

5. The method of Claim 1, wherein (a)-(d) are performed by a controller of the memory device.

6. The method of Claim 1, wherein the memory device is a memory card.

7. The method of Claim 1, wherein the re-encrypted content is stored in flash memory in the memory device.

8. The method of Claim 1, wherein the transport encryption key and the key unique to the memory device are stored in a part of memory that is hidden to a host device.

9. The method of Claim 1 further comprising receiving additional content encrypted with the transport encryption key.
10. The method of Claim 9, wherein the content and the additional content are received from a single entity.

11. The method of Claim 9, wherein the content and the additional content are received from multiple entities.

12. The method of Claim 1 further comprising receiving additional content encrypted with a different transport encryption key.

13. The method of Claim 12, wherein the content and the additional content are received from a single entity.

14. The method of Claim 12, wherein the content and the additional content are received from multiple entities.

15. The method of Claim 1 further comprising, with the memory device, generating the key unique to the memory device.

16. The method of Claim 15, wherein the key unique to the memory device is a random value.

17. The method of Claim 15, wherein the key unique to the memory device is a value not used by any other memory devices in a set.

18. The method of Claim 1, further comprising, with the memory device, receiving the key unique to the memory device from an entity outside of the memory device.

19. The method of Claim 18, wherein the key unique to the memory device is a random value.
20. The method of Claim 18, wherein the key unique to the memory device is a value not used by any other memory devices in a set.

21. The method of Claim 1, wherein the decrypting and the re-encrypting are performed on-the-fly as the content is received by the memory device.
Figure 1
200

Receive a request to replicate content in a memory device

210

Send a request to a transport encryption key server for a transport encryption key

220

Receive the transport encryption key and send the transport encryption key to the memory device

230

Receive encrypted content from a content server

240

Send the encrypted content to the memory device

250

Figure 2
Receive content encrypted with a transport encryption key

Decrypt the content with the transport encryption key

Re-encrypt the content with a key unique to the memory device

Store the re-encrypted content in the memory device

Figure 5
Figure 7
INTERNATIONAL SEARCH REPORT

A. CLASSIFICATION OF SUBJECT MATTER
INV. H04L29/06
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC.

B. RELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
H04L G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched.

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

EPO-Internal, COMPENDEX, INSPEC, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

<table>
<thead>
<tr>
<th>Category</th>
<th>Citation of document, with indication, where appropriate, of the relevant passages</th>
<th>Relevant to claim No</th>
</tr>
</thead>
</table>

D. Further documents are listed in the continuation of Box C

X See patent family annex

F. See patent family annex

Date of the actual completion of the international search

8 July 2010

Date of mailing of the international search report

15/07/2010

Name and mailing address of the ISA

European Patent Office, P B 5818 Patentlaan 2
NL - 2280 HV RIESWIJK
Tel (+31-70) 340-2040
Fax (+31-70) 340-3016

Authorized officer

Ghomrasseni Z
<table>
<thead>
<tr>
<th>Patent document cited in search report</th>
<th>Publication date</th>
<th>Patent family member(s)</th>
<th>Publication date</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>WO 2009045665 A1</td>
<td>09-04-2009</td>
</tr>
</tbody>
</table>