(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
11 July 2013 (11.07.2013)

WIPO | PCT

(10) International Publication Number
WO 2013/103554 Al

(51) **International Patent Classification:**
H04L 29/06 (2006.01) ***G06F 3/06*** (2006.01)
G06F 21/60 (20 13.0 1) ***G06F 21/57*** (20 13.0 1)

(21) **International Application Number:**
PCT/US20 12/07 1423

(22) **International Filing Date:**
2 1 December 2012 (21 .12.2012)

(25) **Filing Language:** English

(26) **Publication Language:** English

(30) **Priority Data:**
13/339,457 29 December 201 1 (29. 12.201 1) US

(71) **Applicant** *(for all designated States except US):* **UNISYS CORPORATION** [US/US]; 801 Lakeview Dr., Suite 100, M/S 2NW, Blue Bell, PA 19422 (US).

(72) **Inventor: OBLIGACION, Eric;** 981 1 Lake Shore Drive, Montgomery Village, MD 20886 (US).

(74) **Agent: GREGSON, Richard, J.;** Unisys Corporation, 801 Lakeview Dr., Suite 100, M/S 2NW, Blue Bell, PA 19422 (US).

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

(84) **Designated States** *(unless otherwise indicated, for every kind of regional protection available):* ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report (Art. 21(3))*

(54) **Title:** METHOD AND DEVICE FOR ADDING OR REPLACING STORAGE

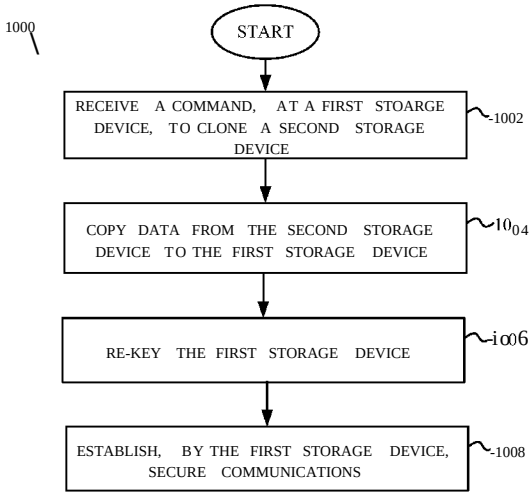


FIG. 10

(57) **Abstract:** In a network of multiple storage devices, a storage device may be¬ come faulty and need to be replaced or additional capacity may need to be ad¬ ded through additional storage devices. When the storage devices communicate through a secure communications link using an encryption key for cryptograph - ically splitting data, replacement or new storage devices may be re-keyed using an encryption key from an existing or prior storage device on the secure data network. After the storage device is re-keyed, the new or replacement storage device may continue to function on the secure data network without requiring changes to clients accessing the secure data network.

WO 2013/103554 A1

METHOD AND DEVICE FOR ADDING OR REPLACING STORAGE DISKS

[0001] The instant disclosure relates to network communications. More specifically, this disclosure relates to securing network communications.

BACKGROUND

[0002] Security is conventionally maintained in organizations by segregating physical networks used by each group of users. This acts to restrict access to data available on computers and databases used in such networks. For example, the physical segregation prevents a user in engineering from gaining access to data in the payroll department's network and vice versa. While separate local network infrastructures help to maintain security of data, superfluous equipment and maintenance is required to maintain these segregated networks. This increases expenses and complexity to the data infrastructures of organizations.

[0003] Regardless of the organizational structure of networks used in commercial, governmental, and other settings, there is an ever increasing security concern that sensitive data transmitted or stored on local networks will be accessed by an unauthorized individual or accidentally accessed or disclosed outside of a group of users, which would compromise the security of the data. Whether a security threat is intentional or unintentional, transmitting data exclusively in one security level partitioned network or another does not protect the data if it is in plaintext format. This is because even strict physical segregation of a network by security level is no guarantee that data will not be disseminated to end-users outside that security level.

SUMMARY

[0004] According to one embodiment, a method includes reading information from a secure boot device. The method also includes executing an operating system contained on the secure boot device. The method further includes establishing a secure connection to a server. The method also includes receiving a request for data from the server. The method further includes cryptographically splitting the requested data. The method also includes transmitting the cryptographically split data to the server.

[0005] According to another embodiment, a computer program product includes a non-transitory computer readable medium having code to read information from a secure boot device. The medium also includes code to execute an operating system contained on the secure boot device. The medium further includes code to establish a secure connection to a server. The medium also includes code to receive a request for data from the server. The medium further includes code to cryptographically split the requested data. The medium also includes code to transmit the cryptographically split data to the server.

[0006] According to a further embodiment, an apparatus includes a processor, a memory coupled to the processor, and a secure boot device coupled to the processor. The processor is configured to read information from the secure boot device. The processor is also configured to execute an operating system contained on the secure boot device. The processor is further configured to establish a secure connection to a server. The processor is also configured to receive a request for data stored in the memory from the server. The processor is further configured to cryptographically split the requested data. The processor is also configured to transmit the cryptographically split data to the server.

[0007] According to one embodiment, a system includes a switch. The system also includes a first storage device connected to the switch over a first communications link through a first communications protocol. The system further includes a secure boot device connected to the first storage device. The system also includes a second storage device connected to the switch over a second communications link through a second communications protocol.

[0008] According to another embodiment, a method includes connecting a first storage device to a network through a first connection. The method also includes connecting a second storage device to a network through a second connection, in which at least one of the first connection and the second connection comprises a fibre channel over Ethernet (FCoE) connection. The method further includes connecting a secure boot device to the first storage device, in which the first storage device accesses the secure boot device to establish a secure connection with a server.

[0009] According to a further embodiment, an apparatus includes a processor, a memory connected to the processor, a secure boot device connected to the processor, and a network adapter connected to the processor. The processor is configured to establish secure communications through the network adapter with the secure boot device. The processor is also configured to communicate through the network adapter with a fibre channel over Ethernet (FCoE) protocol.

[0010] According to one embodiment, a method includes receiving a request, through a secure communications link, from a user for a file in a logical partition. The method also includes identifying a physical location for the file. The method further includes obtaining a copy of the file. The method also includes transferring the copy of the file, through the secure communications link, to the user.

[0011] According to another embodiment, a computer program product includes a non-transitory computer readable medium having code to receive a request, through a secure communications link, from a user for a file in a logical partition. The medium also includes code to identify a physical location for the file. The medium further includes code to obtain a copy of the file. The medium also includes code to transfer the file, through the secure communications link, to the user.

[0012] According to a further embodiment, an apparatus includes a processor, a memory connected to the processor, a secure boot device connected to the processor, and a first network adapter connected to the processor. The processor is configured to receive a request, at the first network adapter through a secure communications link, from a user for a file in a logical partition. The processor is also configured to identify a physical location for the file. The processor is further configured to store a copy of the file in the memory. The processor is also configured to transfer the copy of the file, through the secure communications link with the secure boot device, to the user.

[0013] According to one embodiment, a method includes receiving a command, at a first storage device, to clone a second storage device. The method also includes copying data

from the second storage device to the first storage device. The method further includes re-keying the first storage device with an encryption key matching an encryption key of the second storage device. The method also includes establishing, by the first storage service, secure communications with the encryption key.

[0014] According to another embodiment, a computer program product includes a non-transitory computer readable medium having code to receive a command, at a first storage device, to clone a second storage device. The medium also includes code to copy data from the second storage device to the first storage device. The medium further includes code to re-key the first storage device with an encryption key matching an encryption key of the second storage device. The medium also includes code to establish, by the first storage service, secure communications with the encryption key.

[0015] According to a further embodiment, an apparatus includes a processor, a memory connected to the processor, a secure boot device connected to the processor, and a network adapter connected to the processor. The processor is configured to receive a command, through the network adapter, to clone a second storage device. The processor is also configured to copy data from the second storage device to the memory. The processor is further configured to re-key the secure boot device with an encryption key matching an encryption key of the second storage device. The processor is also configured to establish, through the network adapter, secure communications with the encryption key.

[0016] According to one embodiment, a method includes establishing a secure communications link between a first storage device and a second storage device. The method also includes generating a data stream of data stored on the first storage device. The method further includes splitting the data stream cryptographically. The method also includes transmitting the cryptographically split data.

[0017] According to another embodiment, a computer program product includes a non-transitory computer readable medium having code to establish a secure communications link between a first storage device and a second storage device. The medium also includes code to

generate a data stream of data stored on the first storage device. The medium further includes code to split the data stream cryptographically. The medium also includes code to transmit the cryptographically split data.

[0018] According to a further embodiment, an apparatus includes a processor, a memory connected to the processor, a secure boot device connected to the processor, and a network adapter connected to the processor. The processor is configured to establish a secure communications link between a first storage device and a second storage device with the secure boot device. The processor is also configured to generate a data stream of data stored on the first storage device. The processor is further configured to split the data stream cryptographically. The processor is also configured to transmit the cryptographically split data through the network adapter.

[0019] The foregoing has outlined rather broadly the features and technical advantages of the present invention in order that the detailed description of the invention that follows may be better understood. Additional features and advantages of the invention will be described hereinafter which form the subject of the claims of the invention. It should be appreciated by those skilled in the art that the conception and specific embodiment disclosed may be readily utilized as a basis for modifying or designing other structures for carrying out the same purposes of the present invention. It should also be realized by those skilled in the art that such equivalent constructions do not depart from the spirit and scope of the invention as set forth in the appended claims. The novel features which are believed to be characteristic of the invention, both as to its organization and method of operation, together with further objects and advantages will be better understood from the following description when considered in connection with the accompanying figures. It is to be expressly understood, however, that each of the figures is provided for the purpose of illustration and description only and is not intended as a definition of the limits of the present invention.

BRIEF DESCRIPTION OF THE DRAWINGS

[0020] For a more complete understanding of the disclosed system and methods, reference is now made to the following descriptions taken in conjunction with the accompanying drawings.

[0021] FIGURE 1 is a block diagram illustrating a client with a secure boot device for accessing a secure web application over an unsecure communications link according to one embodiment of the disclosure.

[0022] FIGURE 2 is a flow chart illustrating an exemplary method for transmitting secure data over an unsecure communications link according to one embodiment of the disclosure.

[0023] FIGURE 3 is a flow chart illustrating an exemplary method for transmitting secure data over an unsecure communications link according to another embodiment of the disclosure.

[0024] FIGURE 4 is a block diagram illustrating an exemplary method for splitting and reassembling secure data according to one embodiment of the disclosure.

[0025] FIGURE 5 is a block diagram illustrating a network having a client with a secure boot device according to one embodiment of the disclosure.

[0026] FIGURE 6 is a block diagram illustrating a back end of a secure web application according to one embodiment of the disclosure.

[0027] FIGURE 7 is a flow chart illustrating an exemplary method for transmitting data securely between storage devices in the back end according to one embodiment of the disclosure.

[0028] FIGURE 8 is a block diagram illustrating a back end of a secure web application implementing a hybrid configuration including fiber channel over Ethernet (FCoE) connections according to one embodiment of the disclosure.

[0029] FIGURE 9 is a flow chart illustrating an exemplary method for cloning a disk using Network Data Management Protocol (NDMP) according to one embodiment of the disclosure.

[0030] FIGURE 10 is a flow chart illustrating an exemplary method for replacing one storage device using re-key processing according to one embodiment of the disclosure.

[0031] FIGURE 11 is a block diagram illustrating an network having multiple storage devices presented as a single storage device according to one embodiment of the disclosure.

[0032] FIGURE 12 is a flow chart illustrating an exemplary method for presenting multiple storage device as a single storage device according to one embodiment of the disclosure.

[0033] FIGURE 13 is block diagram illustrating a computer network according to one embodiment of the disclosure.

[0034] FIGURE 14 is a block diagram illustrating a computer system according to one embodiment of the disclosure.

DETAILED DESCRIPTION

[0035] Methods and systems described below improve network security by securing data from both casual and sophisticated eavesdroppers. Further, the disclosure outlines methods and systems for improving the flexibility of networks without compromising security in the network. For example, networks may be partitioned logically without affecting the user. In another example, network devices, such as storage devices, may be located remote to a private cloud network and still have secure access to resources within the private cloud network.

[0036] FIGURE 1 is a block diagram illustrating a client with a secure boot device for accessing a secure web application over an unsecure communications link according to one embodiment of the disclosure. A distributed system 100 may include a server 104, such as a database server, an application server, or a file server. The distributed system 100 may also

include a remote computing systems 106, which may be under control by a different operator than the server 104. In alternative embodiments, the system 100 may include more than one server 104 and/or one computing system 106. Additionally, additional remote computing systems may include mobile devices such as smart phones, cellular phones, laptop computers, and table computers.

[0037] In certain embodiments, the remote computing system 106 may interact with a user through a web page 108, or a dedicated application. The web page 108 may display to a user details of a transaction 105 occurring between the remote computing system 106 and the server 104. For example, the transaction 105 may be a financial transaction or other transaction involving secure communications. A secure connection 110 may be created between the remote computing system 106 and the server 104 to allow transmission of details regarding the transaction 105 over a public network, such as the internet.

[0038] The secure connection 110 may be created when the remote computing system 106 boots into an operating system stored on a secure boot device 102 connected to the remote computing system 106. This secure boot device 102 may store a trusted version of an operating system software and secure communications software for use when the remote computing system 106 establishes the secure connection 110 with the server 104. According to one embodiment, the secure boot device 102 may correspond to a universal system bus (USB) storage device. The remote computing system 106 may boot a USB-bootable operating system from the secure boot device 102. The USB-bootable operating system may provide software capable of communicating with the server 104 over the secure connection 110 and may include software programs capable of cryptographic splitting of data, such as in the methods described below with reference to FIGURES 2-4.

[0039] The secure boot device 102 may additionally provide secure storage that prevents tampering with the software loaded onto the device. This secure storage may be accessed and/or verified by the server 104 during initiation and/or communications over the secure connection 110. The secure boot device 102 may also include other trusted software

modules that may limit the possible operations that a remote computing system 106 may perform when the remote computing system 106 boots from the secure boot device 102.

[0040] For example, the software modules may be configured to prevent the remote computing system 106 from accessing on-secured network resources by limiting access to communication channels, such as Bluetooth, serial connections, and/or other peripheral device connections. The software modules may further prevent the remote computing system 106 from executing application programs stored in a memory of the system itself, such as a local hard drive installed in the remote computing system 106. By operating the remote computing system 106 from the secure boot device 102, the transactions 105 may be trusted by the user at the client computer system 106 and the server 104.

[0041] According to one embodiment, the secure boot device 102 may also include an identity module for providing authentication information to the server 104. For example, the secure boot device 102 may include a smart card (not shown) or a smart card reader (not shown) where a user may insert their individual identification information. According to another embodiment, upon establishment of the secure connection 110 between the remote computing system 106 and the server 104, the user of the remote computing system 106 may be authenticated using identification information stored upon secure storage, such as a community-of-interest key.

[0042] The secure connection 110 may include multiple paths for transmission of data between the remote computing system 106 and the server 104. The paths may be fixed paths or random paths generated by conventional packet-routing networks. Multiple paths for data over the secure connection 110 may improve security by reducing the likelihood of eavesdropping on the secure connection 110.

[0043] FIGURE 2 is a flow chart illustrating an exemplary method for transmitting secure data over an unsecure communications link according to one embodiment of the disclosure. A method 200 may be implemented in a software module included in the secure boot device 102 of FIGURE 1. The method 200 securely transmits cryptographic data set among

logically partitioned data paths. The cryptographic data set may include, for example, one or more encryption keys, filters, and other information useable at an endpoint of a communications link to enable that device to establish a secure communication with a remote system (e.g., another endpoint, a gateway, or any other remote device configured to receive or transmit cryptographically split communication).

[0044] The method 200 begins at block 202 with dividing a cryptographic data set into a plurality of portions. Tag values may be assigned to each portion of the cryptographic data set. Each portion may encapsulated into separate packets, frames, cells, or another unit of data depending on the type of communications link. At block 204, the portions of cryptographic data set may be transmitted from an network interface card of a computing device, such as the client computer system 102 of FIGURE 1.

[0045] At block 206, each portion of cryptographic data is received by a target computing device, such as the server 104 of FIGURE 1. According to one embodiment, the packets received include a new encryption key identifier instructing the target computing device to change keys. In another embodiment, packets received by the target computing device do not include the key identifier. Instead, the target computing device restores a cryptographic data portion encapsulated in a payload portion of the packet using an encryption key stored locally on the target computing device. In some embodiments, the target computing device may store multiple encryption keys, such as for communicating with multiple devices or for rotating encryption keys during a single transaction over a secure communications link.

[0046] When no encryption key matches the received data at block 206, the method 200 continues to block 208. At block 208, packets not matching the encryption key may be discarded, erased, dropped, and/or ignored. Block 208 may be reached when a user of does not have authorization to view a message, because the user (or the user's computing device) lacks the required encryption key, or if the transmitting computing device is not included in a listing of permitted devices at the target device.

[0047] When a matching encryption key is identified at block 206, then each portion of the cryptographic data set is temporarily stored for eventual reassembly at block 210. At this point a tunnel can be established between the sending and receiving computing devices.

[0048] At block 212, the cryptographic data set may be decrypted. Then, the cryptographic data set may be reconstructed by decrypting each portion of the cryptographic data set using the encryption key identified at block 210 and reassembling the decrypted cryptographic data sets. Once all portions of the cryptographic data set are received, the cryptographic data set may be completely reassembled.

[0049] FIGURE 3 is a flow chart illustrating an exemplary method for transmitting secure data over an unsecure communications link according to another embodiment of the disclosure. A method 300 may allow for securely transmitting a message among logically partitioned data paths. The method 300 may be executed in a software module after a secure communications link has been created. At block 302, a message, or a set of data, may be divided into portions and tag values may be assigned to each portion of the set. Each portion may be encapsulated in separate packets using a cryptographic data set at the sending computing device. According to one embodiment, each tag may include metadata indicating a traffic path a particular portion of a message is to follow to a target computing device within a network.

[0050] At block 304, the portions of cryptographic data set are transmitted from network interface of a computing device. According to one embodiment, the different portions of data may be transmitted on different data communication paths. Tag values assigned to each portion of cryptographic data may correspond to a particular communication data path, to transmit the portion of cryptographic data set. For example, a first portion may be transmitted over a communications link towards a first gateway, and a second portion may be transmitted over a communications link towards a second gateway. The portions may be transmitted from a single communications link in different paths, or the portions may be transmitted over two different communications links to achieve different paths. At block 306, each portion of the message sent is received and temporarily stored for later reassembly.

[0051] At block 308, the message may be reconstructed by decrypting each portion of the message and reassembling the portions to recreate the cryptographic data set. When all portions of the message are received, it is possible to fully reassemble the message in a usable form on the target computing device.

[0052] FIGURE 4 is a block diagram illustrating an exemplary method for splitting and reassembling secure data according to one embodiment of the disclosure. An original message 402 is combined with a header 404 and split into data portions 406, by a splitting function 408. The splitting function 408 may also encrypt each of the data portions 406, such that each of the data portions 406 may contain an obfuscated portion of the original message 402. Each of the data portions 406 may then be appended with network layer header 410. The network layer header 410 of each of the data portions 406 may identify the set of data to which the data portion 406 belongs. The data portions 406 may then be passed from a first computing system to a second computing system through different network paths. The second computing system may reassemble the original message 402 with a reassembly function 412. According to one embodiment, the splitting function 408 and the reassembly function 412 may be performed, for example, by a security engine implemented in a software module or in computer hardware. The splitting function 408 and the reassembly function 412 may implement encryption, such as AES-256 encryption.

[0053] FIGURE 5 is a block diagram illustrating a network including a client with a secure boot device according to one embodiment of the disclosure. A network 500 provides for communication over a clear text network, a virtual private network, and/or secure connections. A user at a client device 506 may connect to a private cloud 502, such as a corporate network, via a public network 504, such as the internet. The connection to the private cloud through the public network 504 may be secured through a virtual private network (VPN) connection and/or cryptography as described above with reference to FIGURES 2-4. The client device 506 of the network 500 may be configured to connect to a private cloud 502 with an adapter capable of cryptographic splitting.

[0054] The private cloud 502 may include servers such as a DHCP server 508, a domain server 510, a stealth server 512, and an application server 514, such as an Exchange server. Other network resources may be included in the virtual private network as well. From the internet 504, the private cloud 502 ay be accessed through a VPN server 516 or a secure appliance 518a-b. Additionally, one or more public internet sites 520 may be available to the client device 506 through the public network 504.

[0055] The stealth technology implemented on the client computer device 106 by the boot device 102 may be implemented in storage devices in the back end of a secure web application. Implementing the stealth technology for transmitting data securely over unsecured communications links improves security in the back end services. For example, an eavesdropper present in the data center hosting the back end may no longer be able to eavesdrop on data transmitted in the back end. Further, with stealth technology implemented in back end components, the back end components may be hosted at various locations or on various networks without affecting the security of the data.

[0056] FIGURE 6 is a block diagram illustrating a back end of a secure web application according to one embodiment of the disclosure. A network 600 includes a client computing device 604 coupled to a secure boot device 604a, such as a USB flash drive. The client computing device 604 is coupled through a public network 602, such as the Internet, to a web application executing on a server 612 and/or a mainframe server 614. According to one embodiment, a web application may execute on the server 612 and access data in a database on the mainframe server 614. A router 610 separates a secure portion of the network 600, including the servers 612 and 614, from the public network 602. The server 612 and 614 may be connected to secure boot devices 612a and 614a, respectively. The secure boot devices may be integrated as a processor, such as with the secure boot device 612a connected to the server 612. The secure boot devices may also be external devices, such as with the secure boot device 614a connected to the server 614.

[0057] A hub 620 may be coupled to the router 610 for connecting additional devices to the secure network, including storage devices 622, 624, and 626. Each of the storage devices

622-626 may also be connected to a secure boot device 622a-626a, respectively. The secure boot devices may be modified for operation on the storage devices 622-626. For example, the secure boot devices 622a-626a may include different operating system components and/or security components than the secure boot devices 612a, 614a, and 604a. The different components may be tailored to execute on the different hardware available in the storage devices 622-626 than available on the servers 612-614 and the client computing device 604.

[0058] Additionally, a remote storage device 606 may be coupled to the public network 602. The remote storage device 606 may also include stealth technology embedded in a boot device 606a connected to or integrated in the remote storage device 606. The boot device 606a may allow the remote storage device 606 to establish a secure connection to the servers 612 and 614 located on a secure network behind the router 610. Thus, the remote storage device 606 may be available to applications executing on the servers 612-614 similar to the storage device 622-626, despite being located remote to the servers 612-614. According to one embodiment, the remote storage device 606 may be an auxiliary device to supplement the storage devices 622-626 when demand on the storage devices 622-626 exceeds their capacity. In another embodiment, the remote storage device 606 may be a backup device to replace the storage devices 622-626 during failures or maintenance of the storage devices 622-626.

[0059] Back end devices, located in the secure network or connected to the secure network, having secure boot devices may operate on data over communications links as described for a client device with reference to FIGURES 2-4. That is, the back end devices may cryptographically split data and transmit the data across different network paths. A method for establishing the communications link between a back end device having a secure boot device is described with reference to FIGURE 7. FIGURE 7 is a flow chart illustrating an exemplary method for transmitting data securely between storage devices in the back end according to one embodiment of the disclosure.

[0060] A method 700 begins at block 702 with a back end component, such as a storage device or a server, reading information from the secure boot device. At block 704, the back end component launches an operating system contained on the secure boot device. At

block 706, the back end component establishes a secure connection to a server. At block 708, the back end component receives a request for data transmission to the server, and at block 710, the back end component cryptographically splits the requested data and transmits the data over the secure connection.

[0061] A back end of a network, which hosts web applications or data storage, may including a combination of technologies for accessing and storing data. According to one embodiment, fiber channel over Ethernet (FCoE) may be used in combination with other technologies to achieve improved performance of applications executing on the back end. FIGURE 8 is a block diagram illustrating a back end of a secure web application implementing a hybrid configuration including fiber channel over Ethernet (FCoE) connections according to one embodiment of the disclosure. A network 800 includes several switches and/or hubs 810, 820, and 830. Each of the switches 810, 820, and 830 may be coupled through Ethernet, or another networking technology. Storage devices 812-816, 822-826, and 832-836 may be coupled to the switches 810, 820, and 830, respectively. According to one embodiment, the switch 820 and the storage devices 822-826 may be located remote to the secure network behind a gateway 804. The switch 820 may be coupled through a public network 802 to the gateway 804 and other devices on the secure network behind the gateway 804.

[0062] The switches 810, 820, and 830 may implement different technologies for connecting to the storage devices 812-816, 822-826, and 832-836. For example, the storage devices 812-816 may be connected to the switch 810 through Ethernet connections. Although Ethernet connects the switch 810 to each of the storage devices 812, 814, and 816, each of the storage devices 812, 814, and 816 may communicate through a different protocol to a server 840 connected to the switch 810. For example, the storage device 812 may communicate through advanced technology attachment (ATA) over Ethernet, and the storage devices 814 and 816 may communicate through fibre channel over Ethernet (FCoE). Other protocols available for communication with a storage device may include internet small computer system interface (iSCSI) and fiber channel protocol. According to one embodiment, the storage device 816 may communicate through fiber connections, such as FICON or ESCON, with the switch 810 while the storage devices 812 and 814 communicate with the switch 810 through Ethernet. Regardless

of different physical connections and/or different networking layer protocols employed by the storage devices 812-816, the storage devices 812-816 may employ a common higher-layer protocol for allowing access to files, such as the networking file system (NFS) and/or the common internet file system (CIFS).

[0063] The storage devices 812, 814, 816 may be the same of different physical storage format. For example, the storage device 812 may be a tape drive, the storage device 814 may be a hard disk drive (HDD), and the storage device 816 may be a solid state disk (SSD) comprising flash memory. According to one embodiment, at least one of the storage device 812-816 may comprise multiple storage devices of the same or different type in a redundant array of independent disks (RAID).

[0064] The storage devices 822-826 and 832-836 may be configured similar to the storage devices 812-816 described above. Further, some or all of the storage devices 812-816, 822-826, and 832-836 may be connected to a secure boot device as described above with reference to FIGURE 6 and respond to requests from the server 840 in a method similar to the method described with reference to FIGURE 7. For example, the storage device 826 may include an integrated secure boot device 826a.

[0065] When multiple storage devices are present in a network, whether located locally or remotely through a secure connection, data may be cloned to provide additional copies of data. For example, a new storage device may be attached to a network resulting in a copy of select data from one or more other storage devices. In another example, a replacement storage device may be attached to a network resulting is a cloning of data from an existing storage device to the new storage device. According to one embodiment, the cloning is performed through the network data management protocol (NDMP). FIGURE 9 is a flow chart illustrating an exemplary method for cloning a disk using network data management protocol (NDMP) according to one embodiment of the disclosure.

[0066] A method 900 begins at block 902 with establishing a secure communications link between a first storage device and a second storage device. The secure communications

may be created by booting the first storage device from a secure boot device and establishing a connection to a second storage device using an encryption key contained in the secure boot device. The secure communication may be established over an unsecure communications link when the first storage device is located remote from the second storage device.

[0067] At block 904, a data stream is generated by the first storage device including data stored on the first storage device. According to one embodiment, during a cloning process all of the data stored on the first storage device is accumulated in the data stream. The data stream may be created through a combination of protocols, such as a data stream formed by network data management protocol (NDMP) at a high level and passed to a transmission control protocol (TCP)/internet protocol (IP) at a lower layer.

[0068] According to one embodiment, the data stream may be generated by the first storage device under control of a server. The first storage device may create a second secure connection to the server with the secure boot device. The server may then provide control commands to the first storage device. For example, the first storage device may first establish a secure connection to the server, then, under instruction by the server, begin the method 900 by establishing a second secure connection to a second storage device.

[0069] The server may continue to issue commands to the first storage device after the method 900 begins. For example, the server may instruct the first storage device a selection of files for copy to the second storage device. The data stream may be transmitted directly from the first storage device to the second storage device, without transmission of the data to the server, to improve performance of the copy process. The selection of files for transfer from the first storage device to the second storage device may be specified through a wildcard, a regular expression, and/or a filter, such as by specifying an owner of files to transfer.

[0070] At block 906, the data stream is cryptographically split, such as by the methods described above with reference to FIGURES 2-4. At block 908, the cryptographically split data stream is transmitted to the second storage device.

[0071] Physical storage devices having a secure boot device for operating secure connections may have unique encryption keys or encryption certificates for securing data-at-rest (DAR) on the storage device or securing data-in-motion (DIM) transferred to and/or from the physical storage device. When a physical storage device is replaced in a network, a new physical storage device replacing the old physical storage device should be capable of functioning identical to the old physical storage device to reduce down-time in accessibility of the files. Thus, the unique encryption key and/or certificate should be recreated on the new physical storage device to prevent users from requiring new keys and/or certificates to access the new physical storage device. FIGURE 10 is a flow chart illustrating an exemplary method for replacing one storage device using re-key processing according to one embodiment of the disclosure. Additionally, the flow chart of FIGURE 10 may be implemented for adding a new storage device to a secure data network. Additionally, the flow chart of FIGURE 10 may be implemented for adding a new storage device to a secure data network.

[0072] A method 1000 begins at block 1002 with a replacement physical storage device receiving a command to clone an old physical storage device. At block 1004, the replacement physical storage device clones the old physical storage device by coping all data from the old physical storage device. The cloning process may be proceed as described above with reference to FIGURE 9. At block 1006, the replacement physical storage device is re-keyed to match the encryption certificate and/or key of the old physical storage device. At block 1008, the replacement physical storage device establishes secure communications using the encryption key and/or certificate generated during the re-keying of block 1006. The secure communications may be established through the method described above with reference to FIGURE 1. The replacement physical storage device may then serve data in response to requests from a server by cryptographically splitting data according to the encryption key and/or certificate for transmission as described above with reference to FIGURES 2-4. Re-keying a replacement physical storage device may reduce or eliminate down-time and reconfiguration of users associated with replacement of physical storage devices.

[0073] When data is arranged for storage in several storage devices as illustrated in the diagram of FIGURE 8, data management may become difficult due to the large number of

physical devices. Hosts and applications may require knowledge of which physical device stores a requested file. Tracking files across many physical devices may consume significantly resources and reduce performance of applications or hosts accessing the files stored across multiple physical devices.

[0074] Storage virtualization may be used to separate logical storage from physical storage. Logical storage may be presented to the user through a server, which has a mounted or mapped drive, and physical storage may be the actual location of the storage. Storage virtualization may allow the pooling of physical storage devices into what presents as a single storage device that is managed from a central console.

[0075] FIGURE 11 is a block diagram illustrating an network having multiple storage devices presented as a single storage device according to one embodiment of the disclosure. The network 800 is presented in FIGURE 11 with the physical storage devices 812-816, 822-826, and 832-836 partitioned into separate logical disks for presentation to a user as a single drive. For example, a first logical partition 1110 may include the physical storage devices 822-826 and 836. According to one embodiment, the first logical partition 1110 may include physical storage devices located local to the server 840 and remote to the server 840, in which the remote physical storage devices may have a secure connection to the server 840 through a secure boot device. A second logical partition 1120 may include the physical storage devices 812-816 and 832-834. Although FIGURE 11 shows entire physical storage devices assigned to a logical partition, different files located on a single physical storage device may be assigned to different logical partitions.

[0076] The server 840 may store a table identifying the location on a physical storage device of data within each logical partition. Accesses to data in each logical partition may be made through the server 840. When a user requests a first file in the first logical partition 1110 from the server 840, the server 840 may identify the file as stored on the physical storage device 826. The server may then retrieve the file from the physical storage device 826 and deliver the file to the user. According to one embodiment, the server 840 may redirect the user to the physical storage device 826, rather than cache the file on the server 840 for retrieval by the user.

[0077] According to one embodiment, one of the physical storage devices within a logical partition may be designated as a server for the logical partition and store the look-up table for mapping files within the logical partition to individual physical storage devices. For example, the physical storage device 826 may be designated as a server for the logical partition 1110. Thus, a request for a file in the logical partition 1110 may be made directly to the logical partition 1110, rather than passing the request to the server 840.

[0078] According to one embodiment, storage virtualization may be achieved through block virtualization, which provides seamless data array physical independence and facilitates managing a potentially multi-vendor environment from a single interface within and across datacenters. Block virtualization promotes logical storage flexibility from physical storage such that it may be accessed without regard to physical storage or an underlying heterogeneous structure. This separation allows administrators of the storage system greater flexibility in how they manage storage for end users and also facilitates changing storage vendors should the need arise.

[0079] According to another embodiment, storage virtualization may be achieved through file virtualization, which may be provided by network attached storage (NAS) devices. File virtualization may reduce the dependencies between the data accessed at the file level and the location where the files are physically stored. This provides opportunities to optimize storage use and server consolidation and to perform non-disruptive file migrations. These technologies should provide analytics tools that will help determine what subsystems are prime candidates for virtualization and how they should be consolidated.

[0080] FIGURE 12 is a flow chart illustrating an exemplary method for presenting multiple storage device as a single storage device according to one embodiment of the disclosure. A method 1200 begins at block 1202 with receiving, at a server, a request for a file in a logical partition from a user. At block 1204, the server identifies in a look-up table the location of the file within the logical partition. At block 1206, the server obtains a copy of the file from the physical storage device mapped to the file in the look-up table. At block 1208, the server

transfers the file to the user requesting the file through secure communications, as described above with reference to FIGURES 2-4.

[0081] According to one embodiment, access to storage devices in a logical partition may be accomplished through multipath input/output (MPIO). For example, each of the storage devices may include multiple network adapters. Each network adapter may be coupled through a separate physical connection to the same switch or to a different switch. Likewise, the servers may implement MPIO through multiple network adapters and multiple connections to a switch. MPIO may improve accessibility and reliability of the storage devices and the servers. When one network adapter, switch, or physical connection malfunctions, another path to the storage device or server may be available for establishing secure connections to the storage device or server. Multipath input may allow a storage device or server multiple paths for receiving requests and/or data. Multipath output may allow a storage device or server multiple paths for transmitting requests and/or data. According to one embodiment, multiple paths may be combined, when all paths are functional, to improve performance of a storage device or server by increasing bandwidth available to the server and/or storage device.

[0082] FIGURE 13 illustrates one embodiment of a system 1300 for an information system. The system 1300 may include a server 1302, a data storage device 1306, a network 1308, and a user interface device 1310. The server 1302 may be a dedicated server or one server in a cloud computing system. In a further embodiment, the system 1300 may include a storage controller 1304, or storage server configured to manage data communications between the data storage device 1306 and the server 1302 or other components in communication with the network 1308. In an alternative embodiment, the storage controller 1304 may be coupled to the network 1308.

[0083] In one embodiment, the user interface device 1310 is referred to broadly and is intended to encompass a suitable processor-based device such as a desktop computer, a laptop computer, a personal digital assistant (PDA) or tablet computer, a smartphone or other a mobile communication device having access to the network 1308. When the device 1310 is a mobile device, sensors (not shown), such as a camera or accelerometer, may be embedded in the device

1310. When the device 1310 is a desktop computer the sensors may be embedded in an attachment (not shown) to the device 1310. In a further embodiment, the user interface device 1310 may access the Internet or other wide area or local area network to access a web application or web service hosted by the server 1302 and provide a user interface for enabling a user to enter or receive information.

[0084] The network 1308 may facilitate communications of data, such as authentication information, between the server 402 and the user interface device 1310. The network 1308 may include any type of communications network including, but not limited to, a direct PC-to-PC connection, a local area network (LAN), a wide area network (WAN), a modem-to-modem connection, the Internet, a combination of the above, or any other communications network now known or later developed within the networking arts which permits two or more computers to communicate, one with another.

[0085] In one embodiment, the user interface device 1310 accesses the server 1302 through an intermediate sever (not shown). For example, in a cloud application the user interface device 1310 may access an application server. The application server fulfills requests from the user interface device 1310 by accessing a database management system (DBMS). In this embodiment, the user interface device 1310 may be a computer or phone executing a Java application making requests to a JBOSS server executing on a Linux server, which fulfills the requests by accessing a relational database management system (RDMS) on a mainframe server.

[0086] FIGURE 14 illustrates a computer system 1400 adapted according to certain embodiments of the server 1302 and/or the user interface device 1310. The central processing unit ("CPU") 1402 is coupled to a system bus 1404. The CPU 1402 may be a general purpose CPU or microprocessor, graphics processing unit ("GPU"), and/or microcontroller. The present embodiments are not restricted by the architecture of the CPU 1402 so long as the CPU 1402, whether directly or indirectly, supports the modules and operations as described herein. The CPU 1402 may execute the various logical instructions according to the present embodiments.

[0087] The computer system 1400 also may include random access memory (RAM) 1408, which may be synchronous RAM (SRAM), dynamic RAM (DRAM), and/or synchronous dynamic RAM (SDRAM). The computer system 1400 may utilize RAM 1408 to store the various data structures used by a software application. The computer system 1400 may also include read only memory (ROM) 1406 which may be PROM, EPROM, EEPROM, optical storage, or the like. The ROM may store configuration information for booting the computer system 1400. The RAM 1408 and the ROM 1406 hold user and system data.

[0088] The computer system 1400 may also include an input/output (I/O) adapter 1410, a communications adapter 1414, a user interface adapter 1416, and a display adapter 1422. The I/O adapter 1410 and/or the user interface adapter 1416 may, in certain embodiments, enable a user to interact with the computer system 1400. In a further embodiment, the display adapter 1422 may display a graphical user interface (GUI) associated with a software or web-based application on a display device 1424, such as a monitor or touch screen.

[0089] The I/O adapter 1410 may couple one or more storage devices 1412, such as one or more of a hard drive, a flash drive, a compact disc (CD) drive, a floppy disk drive, and a tape drive, to the computer system 1400. The communications adapter 1414 may be adapted to couple the computer system 1400 to the network 1308, which may be one or more of a LAN, WAN, and/or the Internet. The communications adapter 1414 may also be adapted to couple the computer system 1400 to other networks such as a global positioning system (GPS) or a Bluetooth network. The user interface adapter 1416 couples user input devices, such as a keyboard 1420, a pointing device 1418, and/or a touch screen (not shown) to the computer system 1400. The keyboard 1420 may be an on-screen keyboard displayed on a touch panel. Additional devices (not shown) such as a camera, microphone, video camera, accelerometer, compass, and or a gyroscope may be coupled to the user interface adapter 1416. The display adapter 1422 may be driven by the CPU 1402 to control the display on the display device 1424.

[0090] The applications of the present disclosure are not limited to the architecture of computer system 1400. Rather the computer system 1400 is provided as an example of one type of computing device that may be adapted to perform the functions of a server 1302 and/or the

user interface device 1310. For example, any suitable processor-based device may be utilized including, without limitation, personal data assistants (PDAs), tablet computers, smartphones, computer game consoles, and multi-processor servers. Moreover, the systems and methods of the present disclosure may be implemented on application specific integrated circuits (ASIC), very large scale integrated (VLSI) circuits, or other circuitry. In fact, persons of ordinary skill in the art may utilize any number of suitable structures capable of executing logical operations according to the described embodiments.

[0091] If implemented in firmware and/or software, the functions described above may be stored as one or more instructions or code on a computer-readable medium. Examples include non-transitory computer-readable media encoded with a data structure and computer-readable media encoded with a computer program. Computer-readable media includes physical computer storage media. A storage medium may be any available medium that can be accessed by a computer. By way of example, and not limitation, such computer-readable media can comprise RAM, ROM, EEPROM, CD-ROM or other optical disk storage, magnetic disk storage or other magnetic storage devices, or any other medium that can be used to store desired program code in the form of instructions or data structures and that can be accessed by a computer; disk and disc, as used herein, includes compact disc (CD), laser disc, optical disc, digital versatile disc (DVD), floppy disk and blu-ray disc where disks usually reproduce data magnetically, while discs reproduce data optically with lasers. Combinations of the above should also be included within the scope of computer-readable media.

[0092] In addition to storage on computer readable medium, instructions and/or data may be provided as signals on transmission media included in a communication apparatus. For example, a communication apparatus may include a transceiver having signals indicative of instructions and data. The instructions and data are configured to cause one or more processors to implement the functions outlined in the claims.

[0093] Although the present disclosure and its advantages have been described in detail, it should be understood that various changes, substitutions and alterations can be made herein without departing from the spirit and scope of the disclosure as defined by the appended

claims. Moreover, the scope of the present application is not intended to be limited to the particular embodiments of the process, machine, manufacture, composition of matter, means, methods and steps described in the specification. As one of ordinary skill in the art will readily appreciate from the present invention, disclosure, machines, manufacture, compositions of matter, means, methods, or steps, presently existing or later to be developed that perform substantially the same function or achieve substantially the same result as the corresponding embodiments described herein may be utilized according to the present disclosure. Accordingly, the appended claims are intended to include within their scope such processes, machines, manufacture, compositions of matter, means, methods, or steps.

CLAIMS

What is claimed is:

1. A method, comprising:

receiving a command, at a first storage device, to clone a second storage device;

copying data from the second storage device to the first storage device;

re-keying the first storage device with an encryption key matching an encryption key of the second storage device; and

establishing, by the first storage service, secure communications with the encryption key.

2. The method of claim 1, further comprising receiving a request from a client computing device through the secure communications.

3. The method of claim 1, in which the step of copying the data comprises cloning the second storage device on the first storage device.

4. The method of claim 3, in which the step of cloning comprises reconstructing cryptographically split data from the second storage device.

5. The method of claim 1, in which the first storage device is located remote from the second storage device.

6. The method of claim 1, in which the first storage device is a network attached storage (NAS) device.

7. A computer program product, comprising:

a non-transitory computer readable medium comprising:

code to receive a command, at a first storage device, to clone a second storage device;

code to copy data from the second storage device to the first storage device;

code to re-key the first storage device with an encryption key matching an encryption key of the second storage device; and

code to establish, by the first storage service, secure communications with the encryption key.

8. The computer program product of claim 7, in which the medium further comprises code to receive a request from a client computing device through the secure communications.

9. The computer program product of claim 7, in which the medium further comprises code to clone the second storage device on the first storage device.

10. The computer program product of claim 9, in which the medium further comprises code to reconstruct cryptographically split data from the second storage device.

11. The computer program product of claim 7, in which the first storage device is located remote from the second storage device.

12. The computer program product of claim 7, in which the first storage device is a network attached storage (NAS) device.

13. An apparatus, comprising:

a processor;

a memory connected to the processor;

a secure boot device connected to the processor; and

a network adapter connected to the processor,

in which the processor is configured:

to receive a command, through the network adapter, to clone a second storage device;

to copy data from the second storage device to the memory;

to re-key the secure boot device with an encryption key matching an encryption key of the second storage device; and

to establish, through the network adapter, secure communications with the encryption key.

14. The apparatus of claim 13, in which the processor is further configured to receive a request from a client computing device through the secure communications.

15. The apparatus of claim 13, in which the processor is further configured to clone the second storage device on the first storage device.

16. The apparatus of claim 15, in which the processor is further configured to reconstruct cryptographically split data from the second storage device.

17. The apparatus of claim 13, further comprising a universal serial bus (USB) connected to the secure boot device and the processor.

18. The apparatus of claim 13, in which the secure boot device is integrated into the apparatus.

19. The apparatus of claim 13, in which the apparatus is located remote from the second storage device.

20. The apparatus of claim 13, in which the apparatus is a network attached storage (NAS) device.

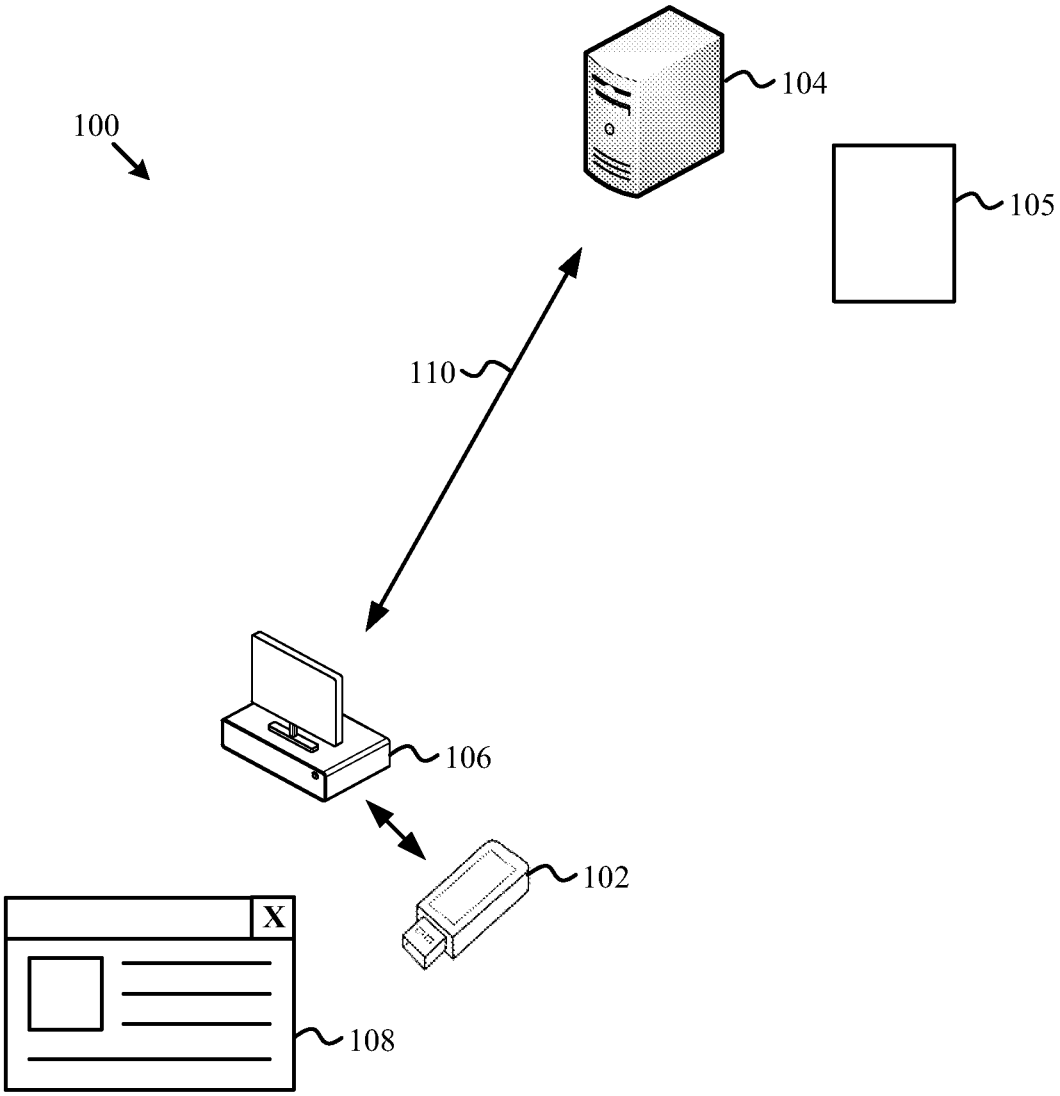


FIG. 1

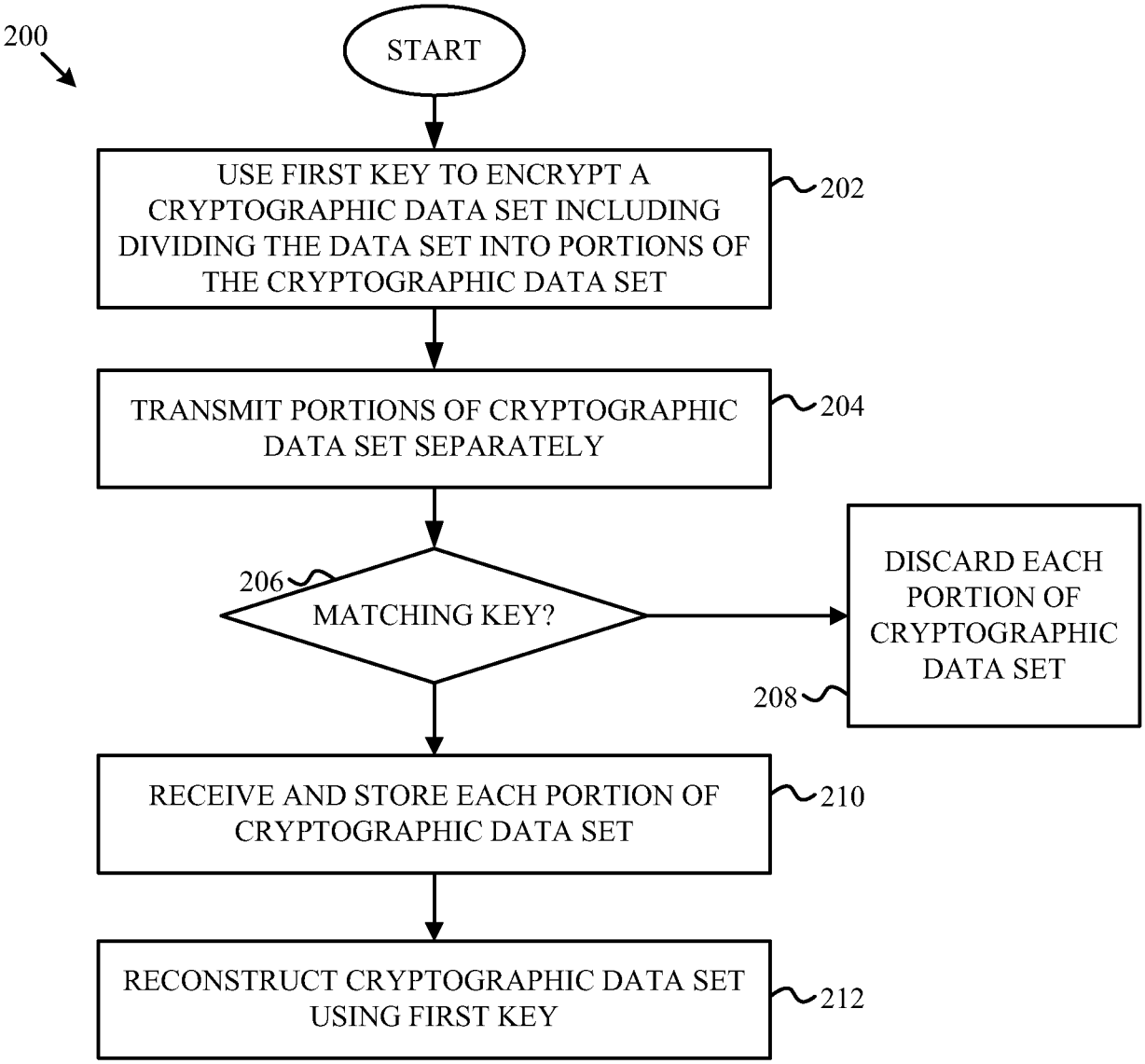

200
START
USE FIRST KEY TO ENCRYPT A CRYPTOGRAPHIC DATA SET INCLUDING DIVIDING THE DATA SET INTO PORTIONS OF THE CRYPTOGRAPHIC DATA SET
202
TRANSMIT PORTIONS OF CRYPTOGRAPHIC DATA SET SEPARATELY
204
206
MATCHING KEY?
DISCARD EACH PORTION OF CRYPTOGRAPHIC DATA SET
208
RECEIVE AND STORE EACH PORTION OF CRYPTOGRAPHIC DATA SET
210
RECONSTRUCT CRYPTOGRAPHIC DATA SET USING FIRST KEY
212

FIG. 2

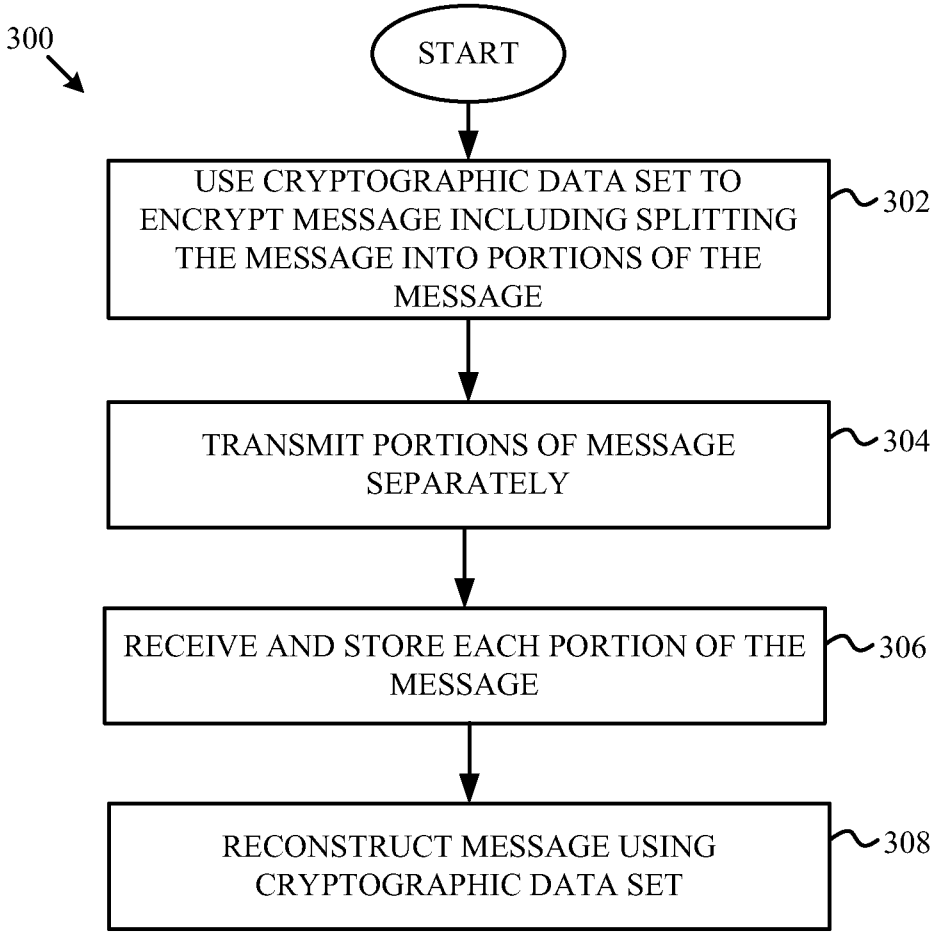

300
START
USE CRYPTOGRAPHIC DATA SET TO ENCRYPT MESSAGE INCLUDING SPLITTING THE MESSAGE INTO PORTIONS OF THE MESSAGE
302
TRANSMIT PORTIONS OF MESSAGE SEPARATELY
304
RECEIVE AND STORE EACH PORTION OF THE MESSAGE
306
RECONSTRUCT MESSAGE USING CRYPTOGRAPHIC DATA SET
308

FIG. 3

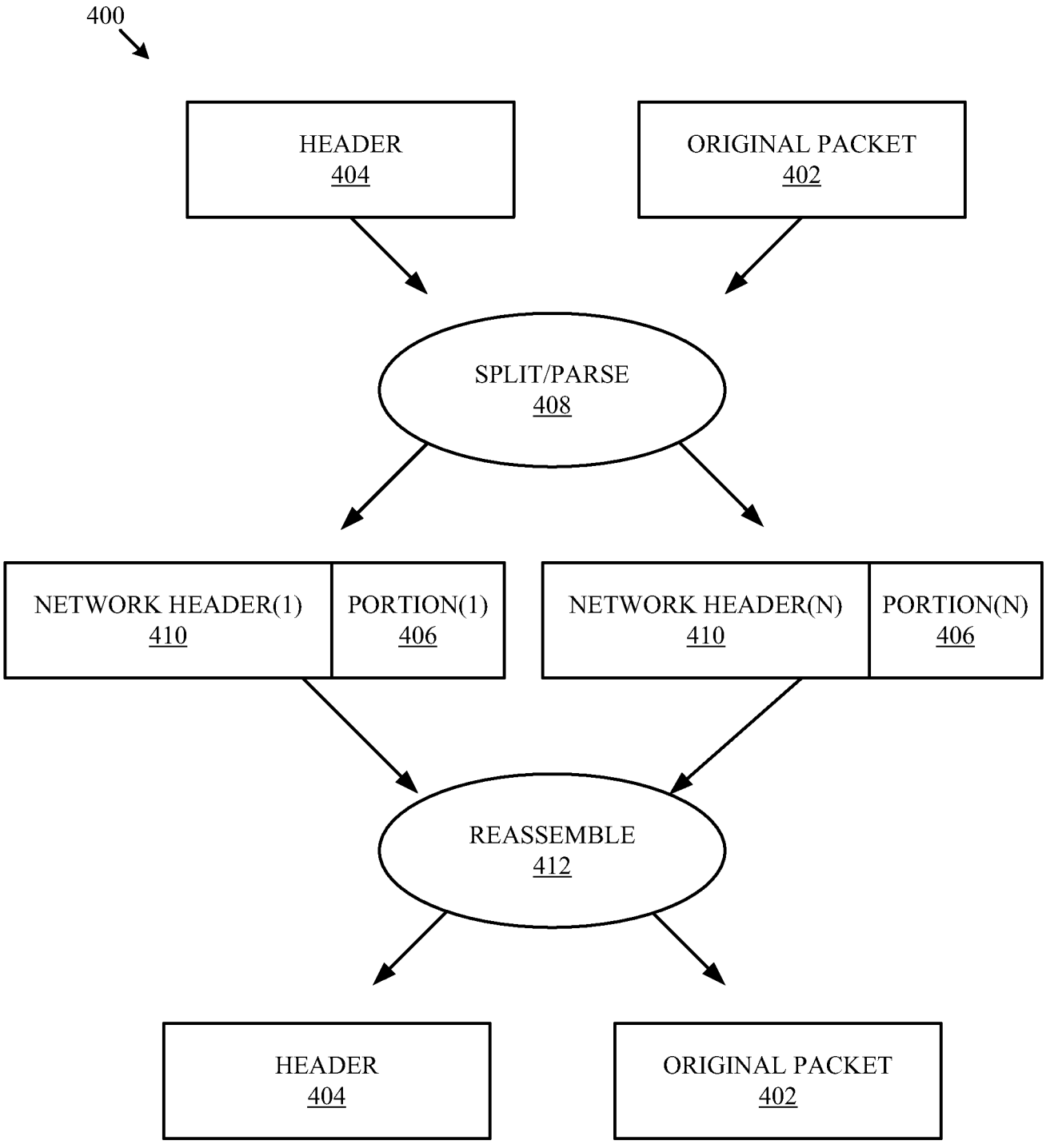

400
HEADER
404
ORIGINAL PACKET
402
SPLIT/PARSE
408
NETWORK HEADER(1)
410
PORTION(1)
406
NETWORK HEADER(N)
410
PORTION(N)
406
REASSEMBLE
412
HEADER
404
ORIGINAL PACKET
402

FIG. 4

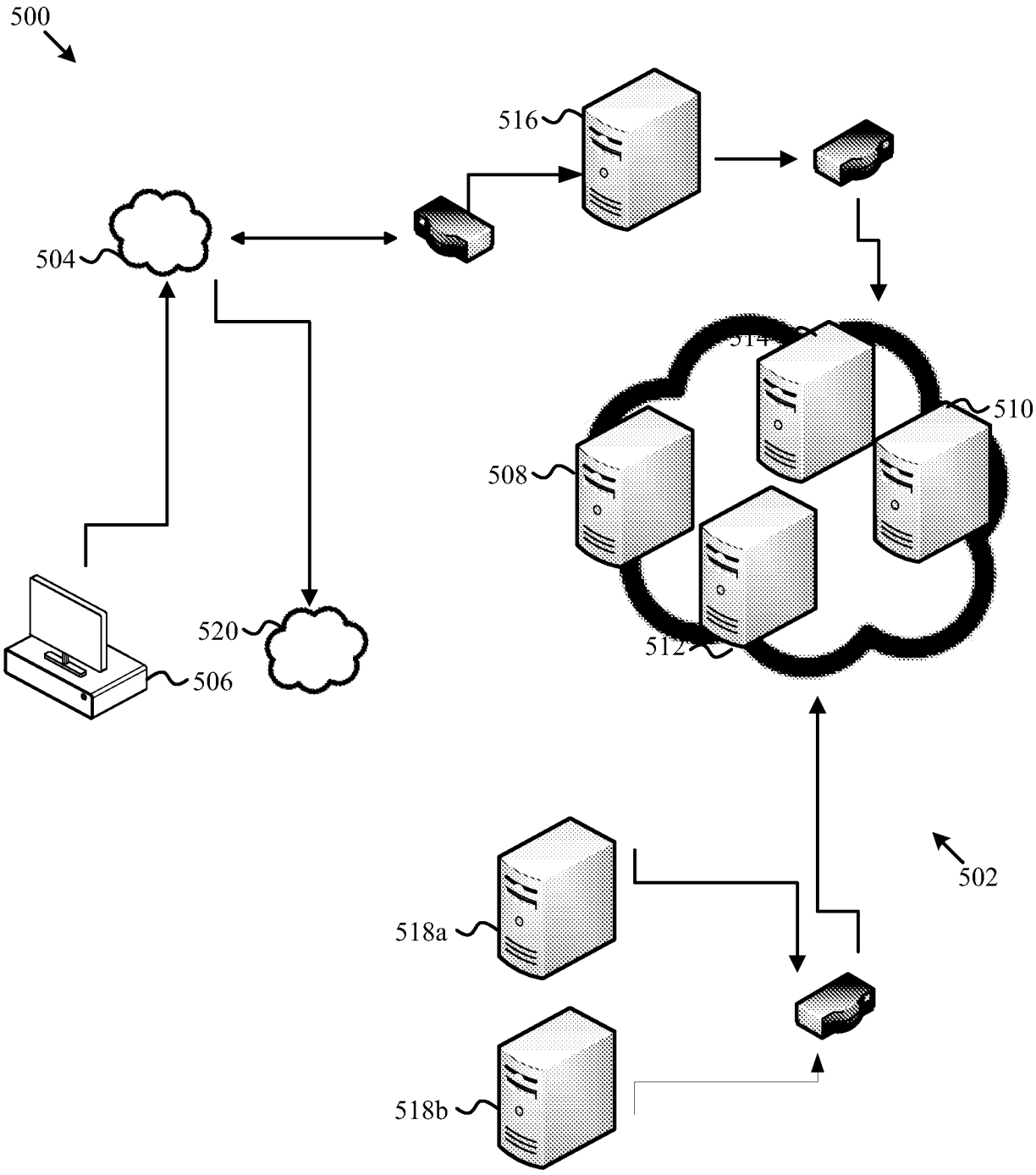


FIG. 5

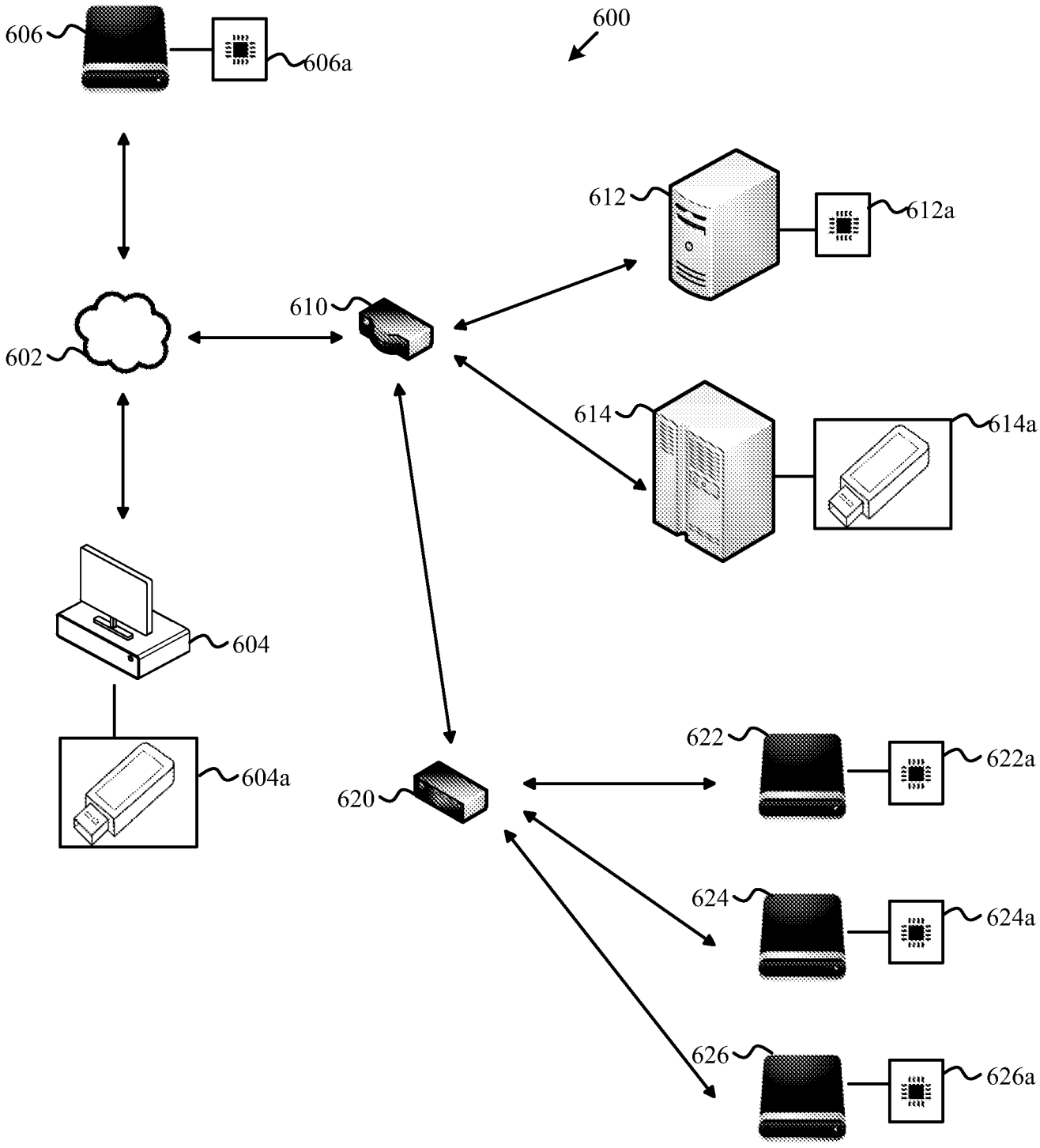

606
606a
600
612
612a
602
610
614
614a
604
604a
620
622
622a
624
624a
626
626a

FIG. 6

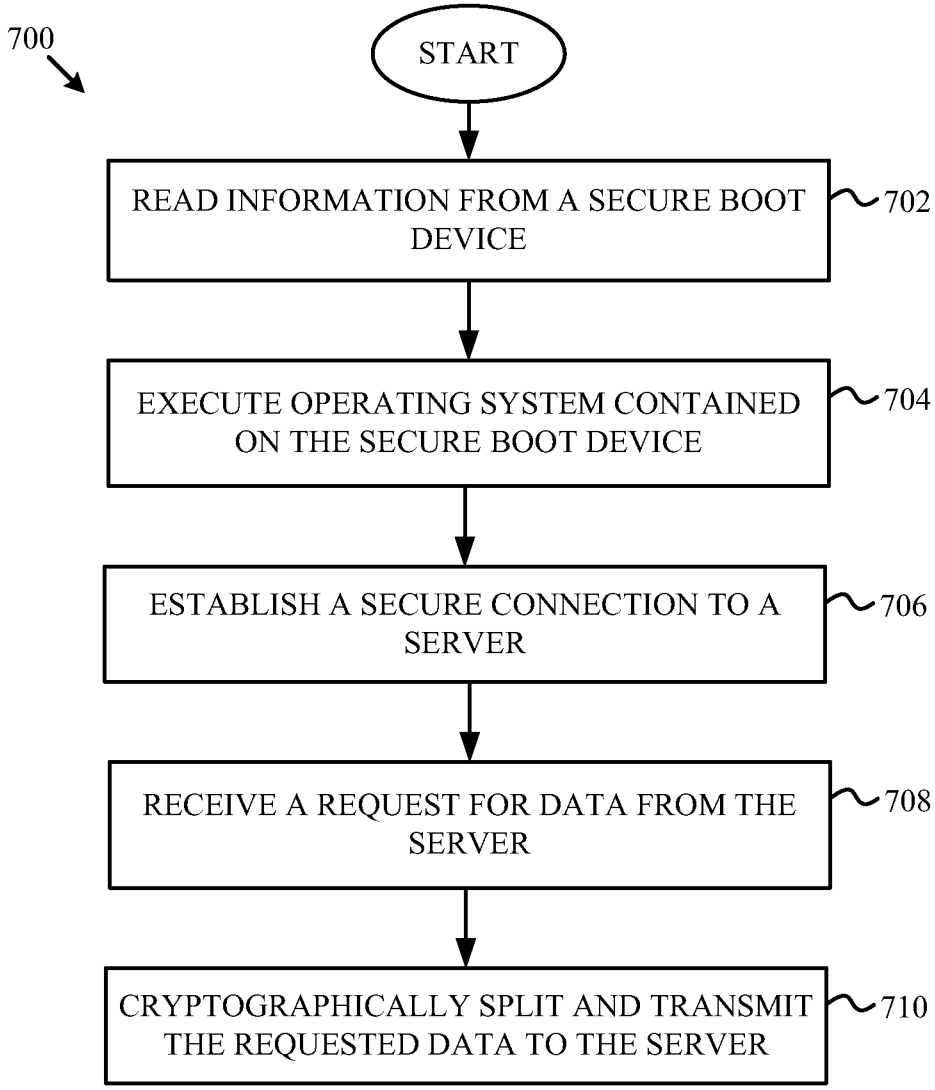

700
START
READ INFORMATION FROM A SECURE BOOT DEVICE
702
EXECUTE OPERATING SYSTEM CONTAINED ON THE SECURE BOOT DEVICE
704
ESTABLISH A SECURE CONNECTION TO A SERVER
706
RECEIVE A REQUEST FOR DATA FROM THE SERVER
708
CRYPTOGRAPHICALLY SPLIT AND TRANSMIT THE REQUESTED DATA TO THE SERVER
710

FIG. 7

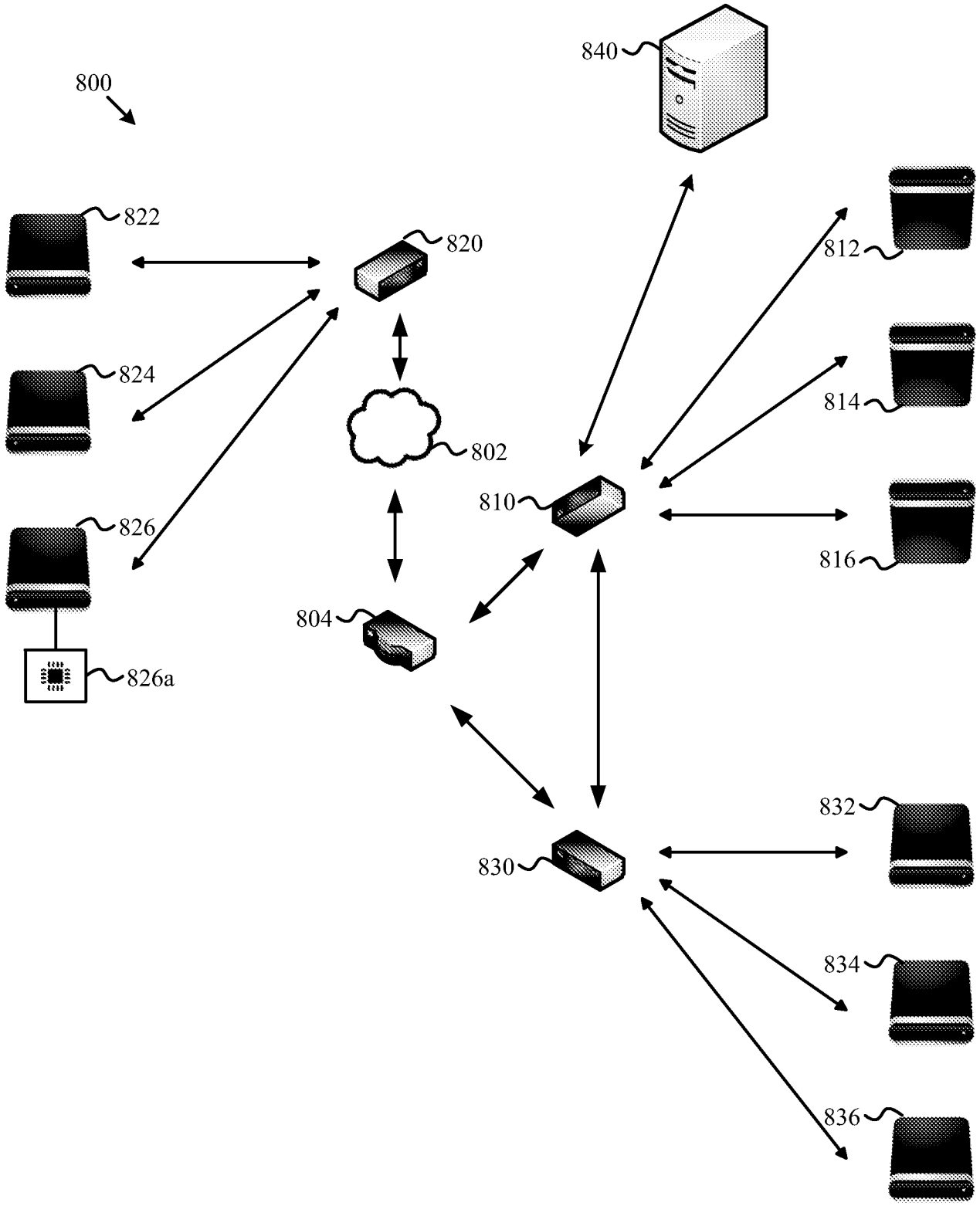

800
840
822
820
812
824
814
802
826
810
816
804
826a
832
830
834
836

FIG. 8

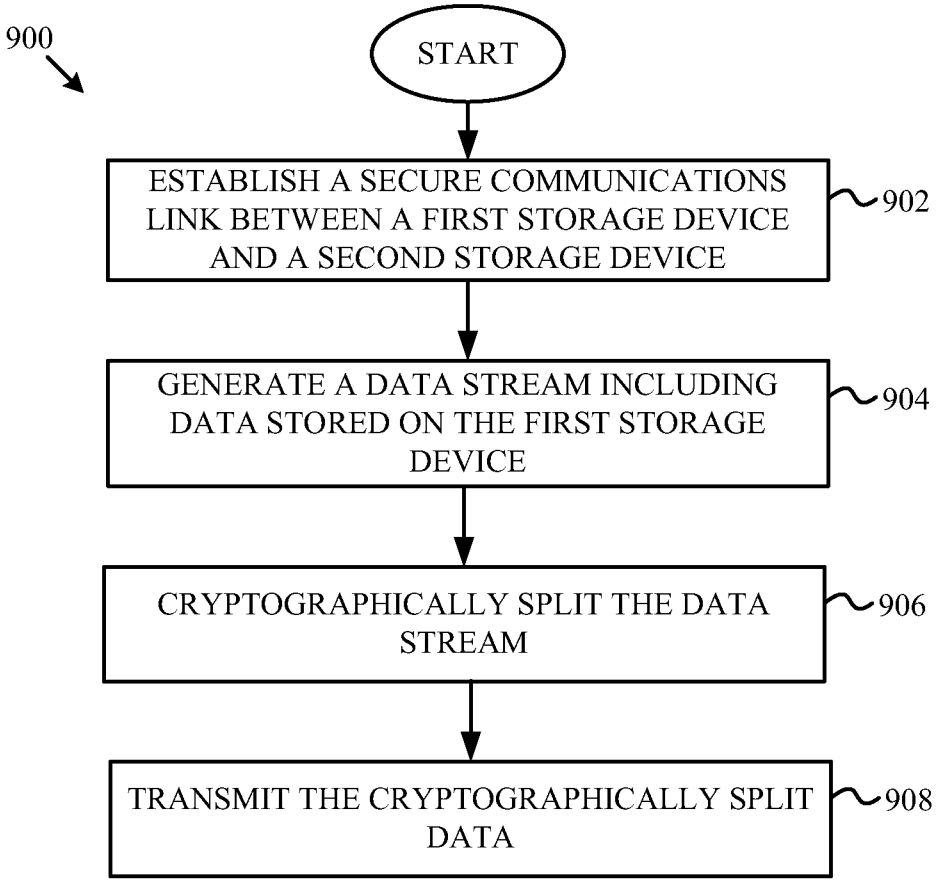

900
START
ESTABLISH A SECURE COMMUNICATIONS LINK BETWEEN A FIRST STORAGE DEVICE AND A SECOND STORAGE DEVICE
902
GENERATE A DATA STREAM INCLUDING DATA STORED ON THE FIRST STORAGE DEVICE
904
CRYPTOGRAPHICALLY SPLIT THE DATA STREAM
906
TRANSMIT THE CRYPTOGRAPHICALLY SPLIT DATA
908

FIG. 9

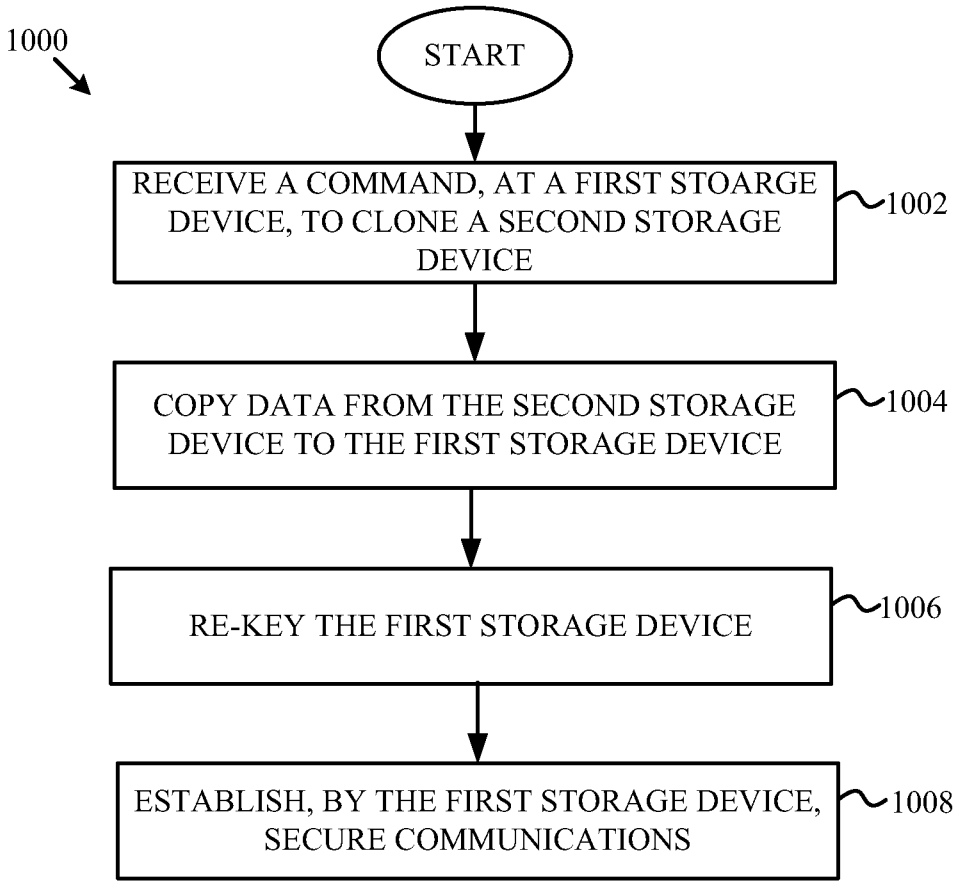

1000
START
RECEIVE A COMMAND, AT A FIRST STOARGE DEVICE, TO CLONE A SECOND STORAGE DEVICE
1002
COPY DATA FROM THE SECOND STORAGE DEVICE TO THE FIRST STORAGE DEVICE
1004
RE-KEY THE FIRST STORAGE DEVICE
1006
ESTABLISH, BY THE FIRST STORAGE DEVICE, SECURE COMMUNICATIONS
1008

FIG. 10

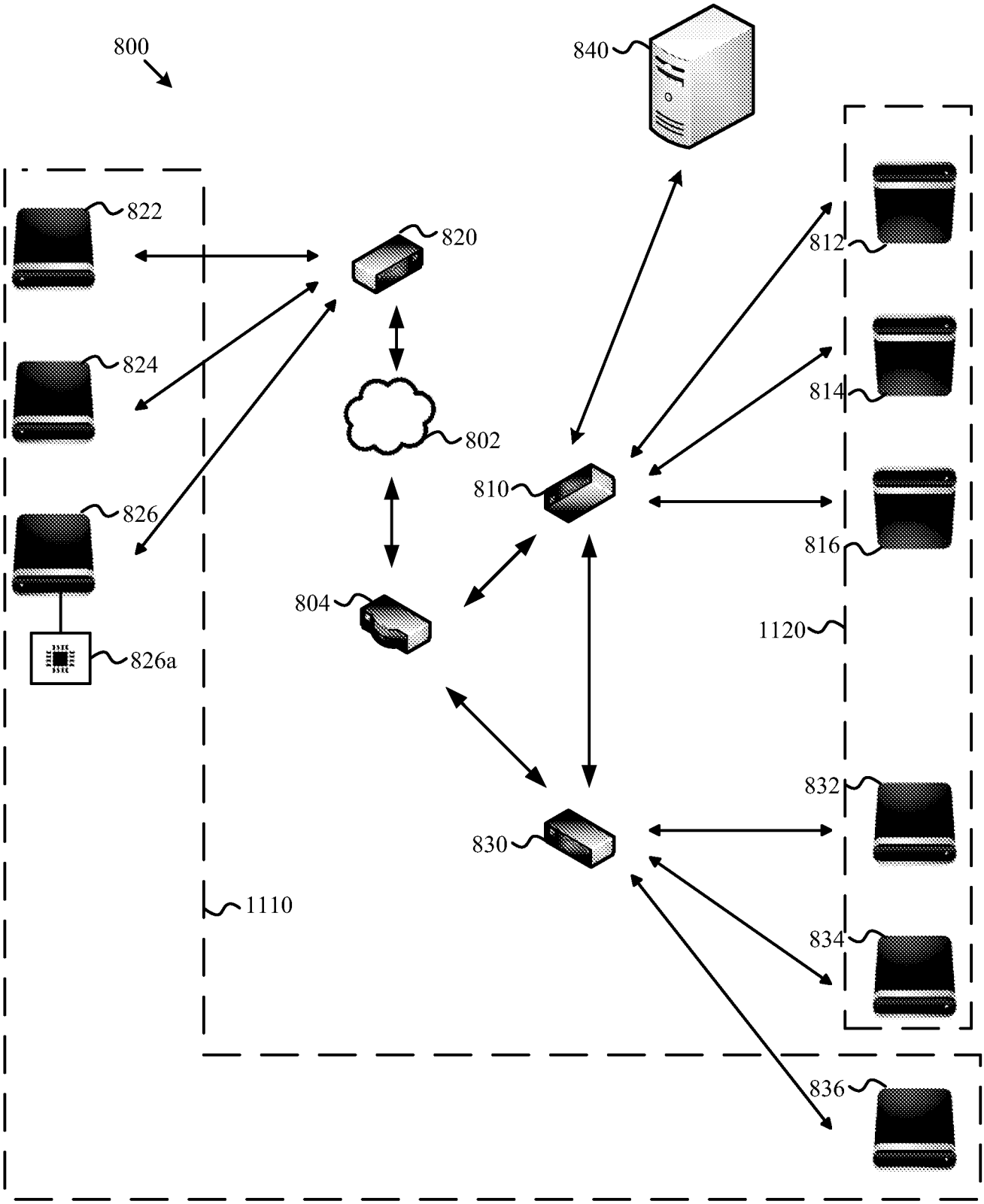

800
840
822
820
812
824
814
802
826
810
816
804
1120
826a
832
830
1110
834
836

FIG. 11

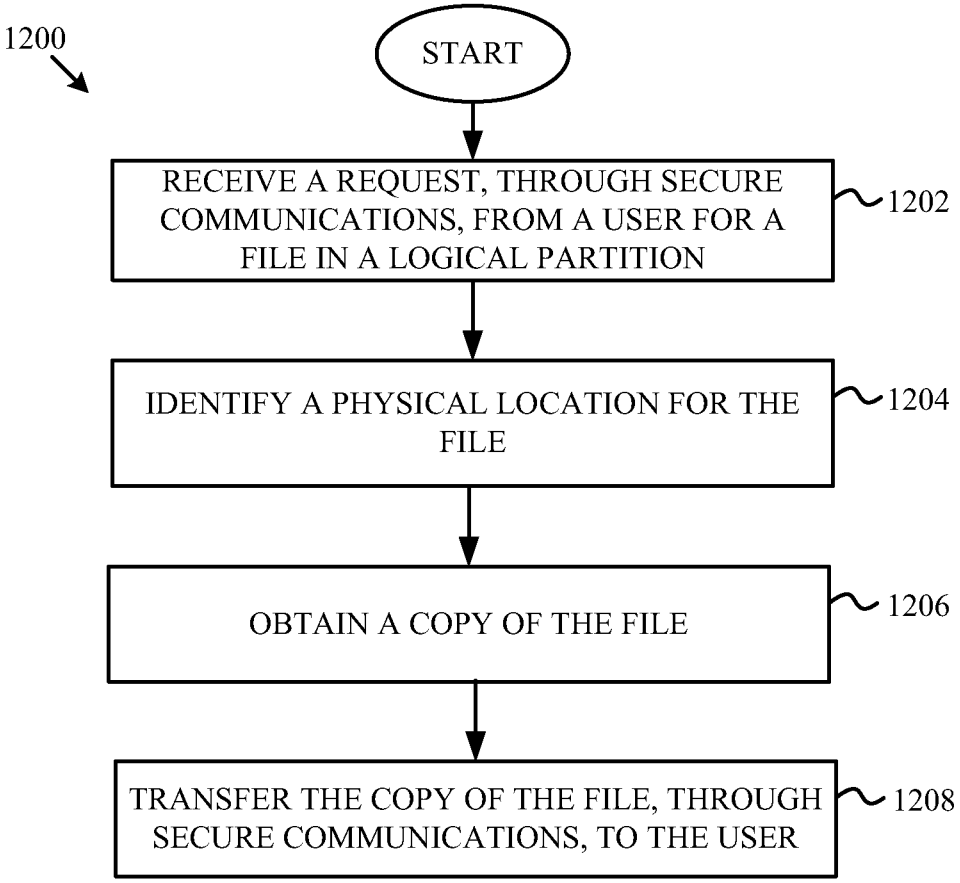

1200
START
RECEIVE A REQUEST, THROUGH SECURE COMMUNICATIONS, FROM A USER FOR A FILE IN A LOGICAL PARTITION
1202
IDENTIFY A PHYSICAL LOCATION FOR THE FILE
1204
OBTAIN A COPY OF THE FILE
1206
TRANSFER THE COPY OF THE FILE, THROUGH SECURE COMMUNICATIONS, TO THE USER
1208

FIG. 12

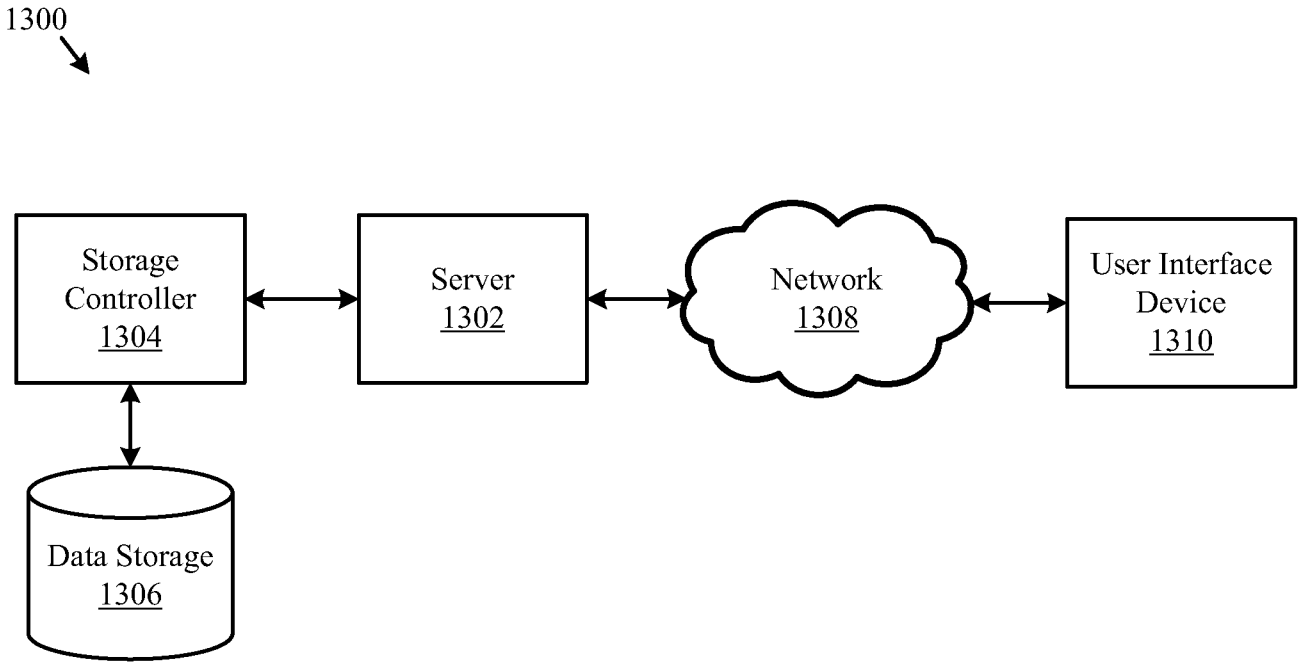

1300
Storage Controller 1304
Server 1302
Network 1308
User Interface Device 1310
Data Storage 1306

FIG. 13

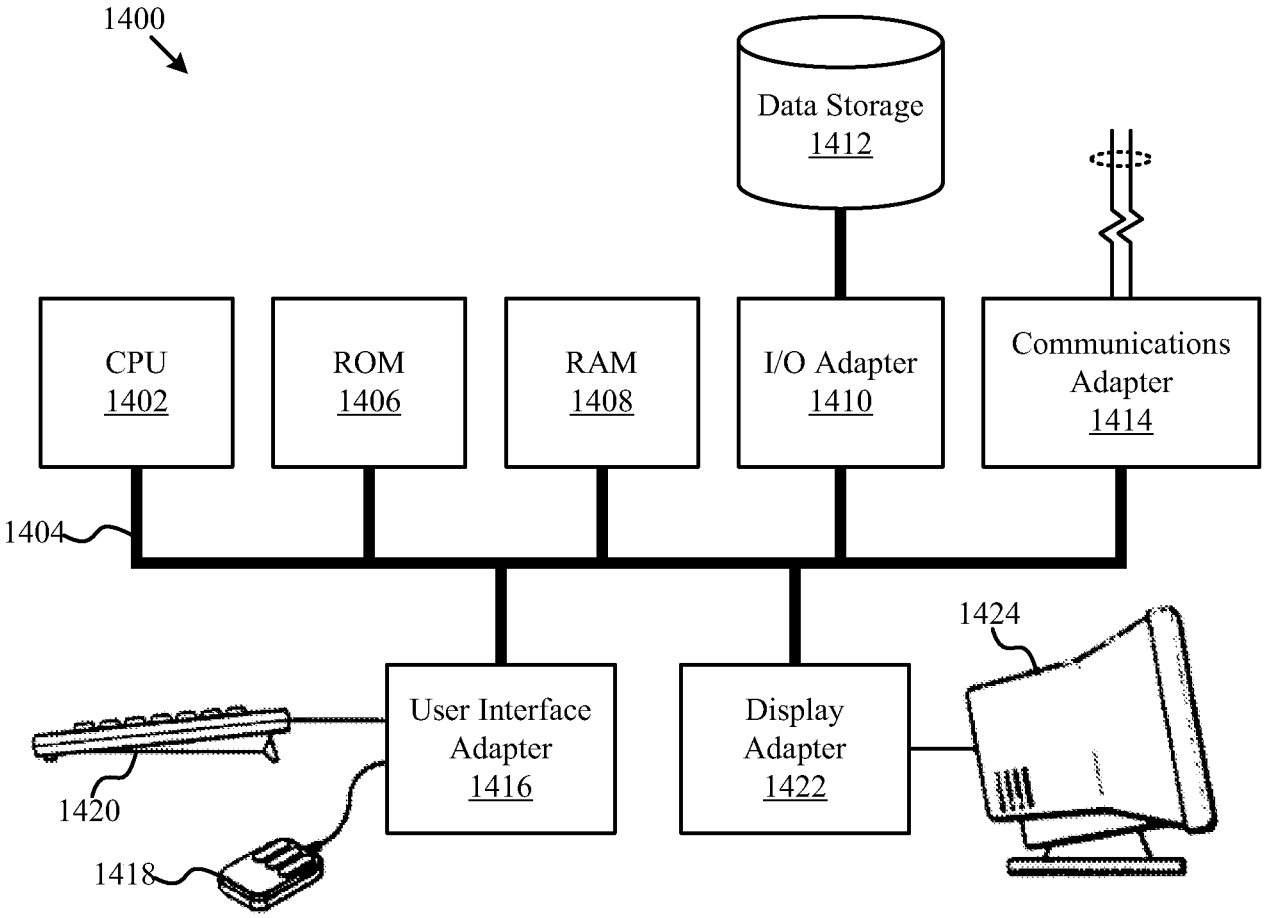

1400
Data Storage
1412
CPU
1402
ROM
1406
RAM
1408
I/O Adapter
1410
Communications
Adapter
1414
1404
1424
User Interface
Adapter
1416
Display
Adapter
1422
1420
1418

FIG. 14

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2012/071423

A. CLASSIFICATION OF SUBJECT MATTER
INV. H04L29/06 G06F21/60 G06F3/06 G06F21/57
ADD.

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

B. FIELDS 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 practicable, search terms used)
EPO-Internal , WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WO 01/01241 A2 (EMC CORP [US]) 4 January 2001 (2001-01-04)	1-3,5-9, 11,12, 14,15, 19,20
Y	page 6, line 28 - page 19, line 13 figures 6,7,8a	4,10,13, 16-18

Y	US 2004/103205 A1 (LARSON VICTOR [US] ET AL) 27 May 2004 (2004-05-27) paragraph [0009] - paragraph [0016] paragraph [0078] - paragraph [0093]	4,10,16

Y	US 2009/222653 A1 (FINDEISEN RALF [DE] ET AL) 3 September 2009 (2009-09-03) paragraph [0029] - paragraph [0030] paragraph [0036] - paragraph [0038]	13,17

	-/-	

[X] Further documents are listed in the continuation of Box C. [X] See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) orwhich is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
22 March 2013	02/04/2013

Name and mailing address of the ISA/	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Strobeck, Anders

2

Form PCT/ISA/210 (second sheet) (April 2005)

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2012/071423

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 2009/319782 A1 (LEE RICHARD M [US] LEE RICHARD M [US] ET AL) 24 December 2009 (2009-12-24) paragraph [0051] - paragraph [0071] ---	13,18
A	EP 0 488 782 A1 (FUJITSU LTD [JP]) 3 June 1992 (1992-06-03) column 1, line 19 - line 31 ---	1-20

2

Form PCT/ISA/210 (continuation of second sheet) (April 2005)

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/US2012/071423

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO 0101241 A2	04-01-2001	JP 4191927 B2	03-12-2008
		JP 2003503790 A	28-01-2003
		us 6845395 Bl	18-01-2005
		us 2004221049 Al	04-11-2004
		Wo 0101241 A2	04-01-2001
US 2004103205 Al	27-05-2004	US 2004103205 Al	27--05-2004
		us 2004107285 Al	03-06--2004
		us 2004107286 Al	03-06-2004
		us 2008005792 Al	03--01--2008
		us 2008216168 Al	04--09--2008
		us 2011270998 Al	03--11--2011
		us 2012042086 Al	16--02--2012
		us 2013013795 Al	10--01--2013
		us 2013019280 Al	17--01--2013
US 2009222553 Al	03-09-2009	CN 101965570 A	02--02-2011
		DE 102008011925 Al	03--09--2009
		EP 2250599 Al	17--11--2010
		JP 2011527777 A	04--11--2011
		KR 20100125371 A	30-11-2010
		TW 200943123 A	16-10--2009
		US 2009222653 Al	03--09--2009
		WO 2009108371 Al	03--09--2009
US 2009319782 Al	24-12-·2009	EP 2291759 Al	09-03-2011
		US 2009319782 Al	24-12-2009
		WO 2009154705 Al	23-12-2009
EP 0488782 Al	03-06-·1992	EP 0488782 Al	03-06-1992
		JP 2603757 B2	23-04-1997
		JP H04205519 A	27-07-1992
		US 5357509 A	18-10-1994