



- (51) **International Patent Classification:**
G06F 9/445 (2006.01) *H04L 29/08* (2006.01)
- (21) **International Application Number:**
PCT/US2015/017811
- (22) **International Filing Date:**
26 February 2015 (26.02.2015)
- (25) **Filing Language:** English
- (26) **Publication Language:** English
- (71) **Applicant:** HEWLETT PACKARD ENTERPRISE DEVELOPMENT LP [US/US]; 11445 Compaq Center Drive West, Houston, TX 77070 (US).
- (72) **Inventors:** CULTER, Bradley G; 11445 Compaq Center Dr W, Houston, Texas 77070 (US). CHRISTIAN, Charles E, Jr.; 13 Queenberry Way, Basking Ridge, New Jersey 07920 (US).
- (74) **Agents:** ADEKUNLE, Olaolu O et al.; Hewlett Packard Enterprise, 3404 E. Harmony Road, Mail Stop 79, Fort Collins, CO 80528 (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, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, 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, SA, 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.

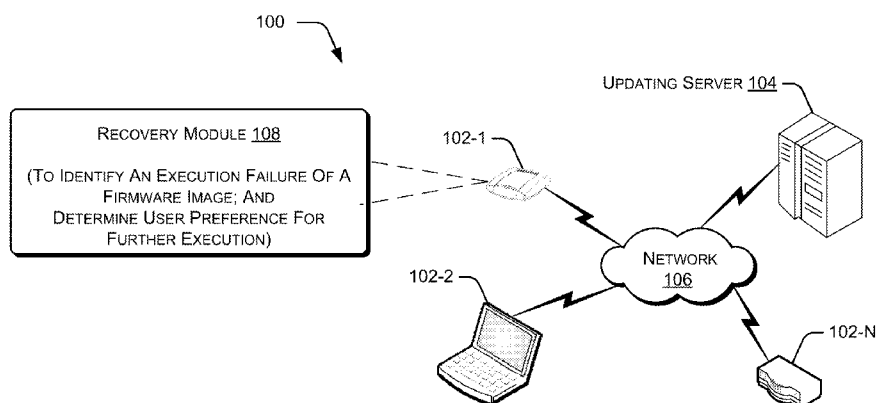
- (84) **Designated States** (*unless otherwise indicated, for every kind of regional protection available*): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, 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, KM, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

- as to the identity of the inventor (Rule 4.17(i))
- as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))

Published:

- with international search report (Art. 21(3))

(54) **Title:** UPDATING FIRMWARE IMAGES**Fig. 1(a)**

(57) **Abstract:** Example implementations relate to updating firmware images in a computing system. For example, a non-transitory computer readable medium having a set of computer readable instructions that, when executed, cause a processor resource to update firmware image of a computing system, is described. The instructions, when executed, cause the processor resource to identify occurrence of an execution failure of a firmware image from a plurality of firmware images, and determine a sentinel MAC address corresponding to the firmware image, where at least one sentinel MAC address is associated with each of the plurality of firmware images. The instructions, when executed, further cause the processor resource to request an updated firmware image from an updating server based on the sentinel MAC address.

UPDATING FIRMWARE IMAGES

BACKGROUND

[0001] Computing systems, for example, laptops, desktops, personal digital assistants (PDA's), servers, and Internet appliances generally include different components, such as input/output (I/O) cards, Basic Input/ Output System (BIOS), and device drivers. Such components generally contain a firmware image, software, or any other suitable code that is executed to perform desired functionalities. Different components of the computing systems may be updated with new firmware image or software from time to time. In an environment where servers are utilized to provide various functionalities, firmware image of BIOS, I/O cards, and drivers may be frequently updated to provide the computing systems with most recent functionalities and capabilities.

BRIEF DESCRIPTION OF DRAWINGS

[0002] The detailed description is provided with reference to the accompanying figures. In the figures, the left-most digit(s) of a reference number identifies the figure in which the reference number first appears. The same numbers are used throughout the drawings to reference like features and components.

[0003] Fig. 1(a) illustrates an example computing environment, implementing a computing system for updating firmware images, according to an example implementation of the present subject matter;

[0004] Fig. 1(b) illustrates various elements of the computing system, according to an example implementation of the present subject matter;

[0005] Fig. 1(c) illustrates various elements of the computing system, according to another example implementation of the present subject matter;

[0006] Fig. 2 is a flowchart representative of an example method of updating firmware images in a computing system;

[0007] Fig. 3 is a flowchart representative of an example method of providing an updated firmware image to a computing system;

[0008] Fig. 4(a) illustrates an example computing environment, implementing a non-transitory computer-readable medium for updating firmware images in a computing system; and

[0009] Fig. 4(b) illustrates an example computing environment, implementing a non-transitory computer-readable medium for providing an updated firmware image to a computing system.

DETAILED DESCRIPTION

[0010] Within the life time of a computing system, many new versions of firmware may be released by hardware vendors and original equipment manufacturers (OEMs) which may include important updates or patches. Different organizations may regularly install such updates to keep their computing systems up-to-date with the new versions of the firmware. Therefore, installation of computing systems in organizations may not be a one-time activity; and these computing systems may have to be updated with new firmware and software from time to time.

[0011] Generally, a computing system may include several components that may have to be updated from time to time. To manage such different types of updates in computing systems, management processors, administered by administrators, are generally implemented by organizations which provide

updating capabilities, such as providing new versions of firmware to the computing devices.

[0012] While updating a computing system with new versions of firmware images, failures may occur. Unexpected issues, such as network unavailability and power outages may interrupt the update process of the firmware images and may render the computing device unusable. In such situations, computing devices with no recovery management have to be sent back to the OEMs for reinstallation of the firmware image, and computing devices with recovery management have to download the entire firmware image again.

[0013] An attempt to reinitiate a failed firmware update may either have to be manually invoked by an administrator, or may have to be initiated by a dedicated program of the computing device. Manual initiation of firmware update is prone to human delays, thereby rendering the computing device unusable during the interim period. For automated initiation of firmware updates, the dedicated programs have to establish contact with the updating servers and request for particular firmware image which involves a lot of hardware resources and necessitates establishing secure and complex network connections. Therefore, computing systems with hardware constraints, such as restricted processing capabilities, small operating random access memory (RAM), and limited flash size are incapable of completing firmware updates automatically.

[0014] According to example implementations of the present subject matter, techniques for updating firmware images in computing systems are described. The described techniques provide for computing systems with hardware constraints to establish direct connection with updating servers and automatically update firmware images.

[0015] The described examples relating to updating firmware images may be implemented in various devices connected through various networks. Although the description herein is with reference to computing systems with constraint hardware resources, the methods and described techniques may also be implemented in other devices, albeit with a few variations. Various

implementations of the present subject matter have been described below by referring to several examples.

[0016] In an example implementation of the present subject matter, the computing device may identify execution failure of a firmware image. In such situations, based on predefined user preference, the computing system may either update the non executable firmware image or may provide an option to the user to execute an alternative firmware image.

[0017] To perform an update of the firmware image and receive an updated firmware image, the computing system may send a firmware update request to the updating server. To generate the firmware image update request, the computing system may determine a sentinel Media Access Control (MAC) address corresponding to the non executable firmware image. Based on the sentinel MAC address, the firmware update request may be generated. In one example implementation, the sentinel MAC address may be included within the firmware update request as one of unique identifier, from amongst a set of unique identifiers, included within the firmware update request. Thereafter, the computing system may send the firmware update request to the updating server and request for the updated firmware image.

[0018] It would be noted that the updated firmware image requested by the computing system may include either an updated version of the non executable firmware image or another copy of the same non executable firmware image.

[0019] In an example implementation, the updating sever may receive the firmware update request from the computing system. Based on the sentinel MAC address, the updating server may determine an updated firmware image to be provided to the computing system. Therefore, the sentinel MAC address may also allow the updating server to uniquely identify the updated firmware image, from amongst a plethora of firmware images available with the server.

[0020] Therefore, the implementation of the described systems and methods may allow computing devices with limited processing capabilities and

small memory to establish a direct connection with the updating server and request an updated firmware image, without specifying the specific updated firmware image to be provided. Similarly, the updating server, upon receiving a firmware update request, may directly provide a firmware image corresponding to the sentinel MAC address included within the request, without identifying any determination of number of firmware images available with the computing system and the specific firmware image to be selected from different available firmware images.

[0021] The above mentioned example implementations are further described with reference to Fig. 1 to Fig 4. It should be noted that the description and figures merely illustrate the principles of the present subject matter along with examples described herein and, should not be construed as a limitation to the present subject matter. It is thus understood that various arrangements may be devised that, although not explicitly described or shown herein, embody the principles of the present subject matter. Moreover, all statements herein reciting principles, aspects, and embodiments of the present subject matter, as well as specific examples thereof, are intended to encompass equivalents thereof.

[0022] Fig. 1(a) schematically illustrates a computing environment 100, implementing different example computing systems, such as computing system 102-1, computing system 102-2, ..., computing system 102-N, according to an example implementation of the present subject matter. For the sake of explanation, the computing system 102-1, computing system 102-2, ..., computing system 102-N have been commonly referred to as computing systems 102, and individually referred to as a computing system 102. The computing environment 100 may either be a public distributed environment or may be a private closed computing environment. According to an implementation of the present subject matter, the computing system 102 may be implemented as, but is not limited to, a workstation, a computer, a switch, a router, a relay, a booster, a repeater, and the like. The computing system 102

may also be a machine readable instructions-based implementation or a hardware-based implementation or a combination thereof.

[0023] The computing system 102 may communicate with different entities of the computing environment 100, such as an updating server 104 through a network 106. The updating server 104 may be any server capable of providing updated firmware images to the computing system 102. The updating server 104 may include, but are not restricted to, different servers, such as virtual servers, management stations, File transfer protocol (FTP) servers, and blade servers. The updating server 104 may implemented as workstations, desktop computers, laptops, smart phones, personal digital assistants (PDAs), tablets, virtual hosts, applications, and the like.

[0024] The network 106 may be a wireless network, a wired network, or a combination thereof. The network 106 may also be an individual network or a collection of many such individual networks, interconnected with each other and functioning as a single large network, e.g., the Internet or an intranet. The network 106 may be implemented as one of the different types of networks, such as intranet, local area network (LAN), wide area network (WAN), and such. The network 106 may either be a dedicated network or a shared network, which represents an association of the different types of networks that use a variety of protocols, for example, Hypertext Transfer Protocol (HTTP), Transmission Control Protocol/Internet Protocol (TCP/IP), etc., to communicate with each other.

[0025] The network 106 may also include individual networks, such as, but are not limited to, Global System for Communication (GSM) network, Universal Telecommunications System (UMTS) network, Long Term Evolution (LTE) network, Personal Communications Service (PCS) network, Time Division Multiple Access (TDMA) network, Code Division Multiple Access (CDMA) network, Next Generation Network (NGN), Public Switched Telephone Network (PSTN), and Integrated Services Digital Network (ISDN). Depending on the implementation, the network 106 may include various network entities, such as base stations, gateways and routers; however, such details have been omitted

to maintain the brevity of the description. Further, it may be understood that the communication between the computing system 102 and the updating server 104 may take place based on the communication protocol compatible with the network 106.

[0026] In an example implementation of the present subject matter, each computing system 102 may include one or more firmware images. Such firmware images may be executed by the computing system 102 to provide different functionalities. During execution of these firmware images, failure may occur and a firmware image may be identified to be non executable, thereby causing interruption to the functioning of the computing system.

[0027] The execution failure may occur due to various reasons, such a failed update of firmware image causing it to be corrupt, a bug in the firmware image making the firmware image non executable, and failure of checksum verification.

[0028] In an implementation of the present subject matter, the computing system 102 may include a recovery module 108. The recovery module 108 may identify such execution failures of the firmware images, and also determine a user preference describing actions to be performed on occurrence of the execution failures. For example, the user preference may describe either to update the non executable firmware image, or to execute an alternate firmware image.

[0029] Based on the identified user preference, the recovery module 108 of the computing system 102 may either update the non executable firmware image or may provide available executable images to a user for selection. Therefore, the recovery module may provide automated recovery of the computing system 102 from the situations of execution failure of firmware images. The manner in which the recovery module 108 may perform different actions has been further explained in detail in reference to Fig. 1(b) and Fig. 1(c).

[0030] Fig. 1(b) schematically illustrates elements of the computing system 102, according to an example implementation of the present subject matter. The computing system 102, apart from the recovery module 108, may include a communication module 118 communicatively coupled to a processor 110.

[0031] The processor 110 may execute an operating system and other applications and services on the computing system 102. The processor 110 of the computing system 102 may be implemented as microprocessor(s), microcomputer(s), microcontroller(s), digital signal processor(s), central processing unit(s), state machine(s), logic circuit(s), and/or any device(s) that manipulates signals based on operational instructions. Among other capabilities, the processor 110 may fetch and execute computer-readable instructions stored in a memory. The functions of the processor 110 may be provided through the use of dedicated hardware as well as hardware capable of executing machine readable instructions.

[0032] In operation, the recovery module 108 of the computing system 102 may identify occurrence of an execution failure of a firmware image. As described earlier, the execution failure may occur due to various reasons, such as due to a failed update of the firmware image, or a bug in the firmware image. Upon identification of such execution failures, the recovery module 108 may also ascertain user preference related to execution failure. The user preference may indicate whether the non executable firmware image is to be updated from the updating server, or an alternate firmware image is to be executed.

[0033] If based on the user preference, it is determined by the recovery module 108 that the non executable firmware image is to be updated; the recovery module 108 may determine a sentinel MAC address corresponding to the non executable firmware image. In an example, the computing system 102 may store a separate and unique MAC address corresponding to each firmware image, and accordingly, the determined MAC address may uniquely correspond to the non-executable firmware image.

[0034] In one example implementation of the present subject matter, the communication module 118 of the computing system 102 may send a firmware update request to the updating server 104. The communication module 118 may generate the firmware update request based on the determined sentinel MAC address. In one implementation, the communication module 118 may also receive updated firmware image from the updating sever 104.

[0035] The operations of the computing system 102, to perform the updating of the firmware image, are explained in further detail in reference to Fig. 1(c).

[0036] Referring to Fig. 1(c), in one example implementation the computing system 102 comprises, apart from processor 110, interface(s) 112, module 114, and data 116 coupled to the processor 110.

[0037] The interface(s) 112 may include a variety of machine readable instructions-based interfaces and hardware interfaces that allow the computing system 102 to interact with different entities, such as the updating server 104 and the network 106. Further, the interface(s) 112 may enable the components of the computing system 102 to communicate with other components, such as the processors 110, the modules 114 and the data 116.

[0038] In one example implementation of the present subject matter, the modules 114 of the computing system 102 may include different module(s). The modules 114 may include other module(s) 120 in addition to the recovery module 108 and the communication module 118. The modules 114 may be communicatively coupled to the processor 110 of the computing system 102. The module(s), amongst other things, include routines, services, programs, objects, components, data structures, and the like, which perform particular tasks or implement particular abstract data types. The modules 114 may further include sub modules that supplement applications on the computing system 102, for example, services of the firmware. The other modules 120 may support different functions of the computing system 102 and may provide different functionalities for the purpose of functioning of the computing system 102.

[0039] The data 116 serves, amongst other things, as a repository for storing data that may be fetched, processed, received, or generated by the module 114 of the computing system 102. The data 116 may include different firmware images 122, sentinel MAC addresses 124, and other data 132. Although the data 116 is shown internal to the computing system 102, it may be understood that the data 116 may reside in an external repository (not shown), which may be communicatively coupled to the computing system 102. The computing system 102 may communicate with the external repository through the interface(s) 112 to obtain information from the data 116.

[0040] In an example implementation of the present subject matter, the data 116 may be stored in a memory (not shown). The memory may include any computer-readable medium including, for example, volatile memory (e.g., RAM), and/or non-volatile memory (e.g., EPROM, flash memory, Memristor, etc.). The memory may be communicatively coupled to the processor 110 and different components of computing system 102.

[0041] In an example implementation of the present subject matter, the computing system 102 may include multiple firmware images 122 for execution. The different firmware images may be executed by the computing system from time to time to provide different and varied functionalities. Further, the computing system 102 may include a predefined sentinel MAC address 124 corresponding to each firmware image. Each sentinel MAC address may uniquely correspond to a firmware image within the computing system 102. Therefore, each computing system 102 may include a general MAC address for the purpose of communication, along with several different sentinel MAC addresses 124, each corresponding to a firmware image. The sentinel MAC addresses may either be hardcoded into the computing system, or may be configurable, as per the configuration of the OEM. As explained earlier, the sentinel MAC address 124 may be stored in the memory of the computing system 102 as sentinel MAC addresses 124.

[0042] In operation, the computing system may identify occurrence of an execution failure event corresponding to a firmware image. The execution failure

of the firmware image may occur due to various reasons. For example due to an incomplete firmware update, the firmware image may get corrupted and the execution may fail. Similarly, checksum of the firmware image may fail due to which the firmware execution may not be completed. Therefore, the computing system 102 may encounter an occurrence of the execution failure situation.

[0043] In one example implementation of the present subject matter, the recovery module 108 of the computing system 102 may identify the occurrence of the execution failure. The recovery module 108 may implement a watchdog timer to analyze the execution any firmware image. In situations where the execution is not completed, the watchdog timer may expire and the recovery module 108 may determine the occurrence of the execution failure in the computing system 102.

[0044] The recovery module 108 may thereafter ascertain the user preference of execution, in situations of execution failures. The user preference can be understood as a choice of performing an action upon occurrence of the execution failure. In an example implementation, the user preference may be predefined and available with the computing system 102. In another example implementation, the user preference may be received by prompting a user of the computing system 102. The user preference may indicate either to perform an update of the non executable firmware image or to execute an alternative firmware image.

[0045] In one example implementation, to allow automatic recovery of the computing system 102 after the situation of the execution failure, the user preference may be predefined. For example, a user preference of performing an update of the non-executable firmware image may be defined. In such instances, the recovery module 108 may determine the predefined user preference, and may automatically initiate updating of the non executable firmware image.

[0046] In case the user preference is predefined to execute an alternate firmware image, the recovery module 108 may identify such alternate firmware

image and may execute it on the computing system 102. In another example, the recovery module 108 may also prompt the user to select the alternative firmware image from different available firmware images 122 for the purpose of execution.

[0047] In situations where the user preference may be ascertained to be performing an update of the non executable firmware image, the recovery module 108 may determine a sentinel MAC address corresponding to the firmware image to be updated. As described earlier, the computing system 102 may include different sentinel MAC addresses 124 corresponding to the different firmware images 122. The sentinel MAC addresses 124 may be different from the general MAC address of the computing system 102, used for the purpose of communicating with the network 106.

[0048] The communication module 118 of the computing system 102 may utilize the sentinel MAC address determined by the recovery module 108 to generate a firmware update request. In one example implementation, the firmware update request may include a set of unique identifiers, uniquely corresponding to the non executable firmware image and the updated firmware image to be requested from the updating server 104. The set of unique identifiers may include different identifiers, like the sentinel MAC address corresponding to the non-executable firmware image, Internet Protocol (IP) address associated with the computing system 102, an address representing the updated firmware image like a trivial file transfer protocol (tftp) address, etc.

[0049] In an implementation, the communication module 118 may send the firmware update request to the updating server 104. In an example, the firmware update request may be a new request including the set of unique identifiers, communicated by the communication module 118 through the network 106.

[0050] In another example, the communication module 118 may send the firmware update request as a modified 'bootp' request. The modified bootp request may include the sentinel MAC address corresponding to the non

executable firmware image, instead of the MAC address of the computing system 102, in the Ethernet frame. Modification of the bootp request may allow the communication module 118 to establish secure connection with the updating server 104 without sharing any different or excessive handshake information. Accordingly, constraint computing systems 102 may perform firmware updates with ease and based on standard communication protocols.

[0051] It would be noted that the updated firmware image requested by the computing system 102 may include either an updated version of the non executable firmware image, or another copy of the same non executable firmware image.

[0052] The updating server 104 may receive the firmware update request along with the set of unique identifiers. Based on the set of unique identifiers, the updating server 104 may determine an updated firmware image, to be provided to the computing system 102. As discussed earlier, the set of unique identifiers may either include the sentinel MAC address, or may be based on the sentinel MAC address to uniquely indicate the firmware image to be updated.

[0053] In one example implementation of the present subject matter, the updating server 104, capable of providing updated firmware images to the computing system 102 may include mapping, such as a lookup table with details of available firmware images and their corresponding associated sentinel MAC addresses 124. It would be understood that since the updating sever 104 may provide updated firmware images to multiple computing systems 102, the sentinel MAC addresses in the mapping may be unique. However, two or more different sentinel MAC addresses may correspond to a same firmware image depending on the computing systems and their firmware implementation.

[0054] For example, the updating server 104 may provide updated firmware images to 20 different computing systems 102. In said example, each computing system 102 may include 3 firmware images. Accordingly, each computing system 102 may include 3 different sentinel MAC addresses. Whereas, the updating server 104 may include a mapping between 60 sentinel

addresses, corresponding to the 20 different computing systems 102. It may occur that 2 computing systems 102, say computing system 'A' and computing system 'B', may be similar and may implement same firmware images. In such situation, the sentinel address of the computing system 'A' may correspond to similar firmware images, as would the sentinel MAC addresses of the computing system 'B' would correspond to.

[0055] Therefore, the sentinel MAC addresses included in the mapping of the updating server 104 may be unique; however, two or more different sentinel MAC addresses may correspond to a same firmware image depending on the computing systems and their firmware implementation.

[0056] Upon identification of the updated firmware image corresponding to the set of unique identifiers which may include the sentinel MAC address, the updating server 104 may send the updated firmware image to the computing system 102. In an example implementation of the present subject matter, the communication module 118 of the computing system 102 may receive the updated firmware image and may provide it to the recovery module 108 for processing.

[0057] The recovery module 108 may reprogram a flash memory of the computing system 102 based on the received updated firmware image and may then execute the firmware image. The described implementations of updating the firmware image in a computing system may allow computing devices to directly communicate with the updating servers 104 and request firmware images based on established communication standards. Further, the use of set of unique identifiers based on the sentinel MAC address may allow the updating servers 104 to efficiently and quickly identify the updated firmware image to be provided to the computing system 102.

[0058] Fig. 2 and 3 illustrate method 200 and 300 for performing a firmware image update and for providing an updated firmware image, respectively, according to an example implementation of the present subject matter. The order in which the methods 200 and 300 are described is not

intended to be construed as a limitation, and any number of the described method blocks may be combined in any order to implement the methods 200 and 300, or any other alternative methods. Furthermore, the methods 200 and 300 may be implemented by processor(s) or computing system(s) through any suitable hardware, non-transitory machine readable instructions, or combination thereof.

[0059] It may be understood that steps of the methods 200 and 300 may be performed by programmed computing systems. The steps of the methods 200 and 300 may be executed based on instructions stored in a non-transitory computer readable medium, as will be readily understood. The non-transitory computer readable medium may include, for example, digital memories, magnetic storage media, such as one or more magnetic disks and magnetic tapes, hard drives, or optically readable digital data storage media.

[0060] Further, although the methods 200 and 300 may be implemented in a variety of communication networks utilizing different computing systems and updating servers; in an implementation described in Fig. 2 and 3, the methods 200 and 300 are explained in context of the aforementioned computing system 102 and updating server 104 for ease of explanation. Although, for the sake of explanation, the forgoing explanation has been made considering that one computing system 102 is communicatively coupled to the updating server 104, it would be noted that the updating server 104 may be coupled to more than one computing system 102.

[0061] Referring to Fig. 2, in an implementation of the present subject matter, at block 202, occurrence of an execution failure of a firmware image may be identified. In an example implementation, the computing system 102 may include multiple firmware images and execution of any one of these multiple firmware images may fail.

[0062] At block 204, a sentinel media access controller (MAC) address corresponding to the firmware image may be determined. In an example implementation, the computing system 102 may include different sentinel MAC

addresses, each uniquely corresponding to one firmware image of the computing system.

[0063] At block 206, an updated firmware image is requested from an updating server, based on the sentinel mac address. In an example implementation, the computing system 102 may send a firmware update request to the updating server 104 to receive the updated firmware image. Further, the firmware update request may include a set of unique identifiers, such as the sentinel MAC address to uniquely correspond to the updated firmware image available with the updating server.

[0064] Referring to Fig. 3, in an implementation of the present subject matter, at block 302 a firmware update request is received from a computing system, by the updating server. The firmware update request may include a sentinel MAC address, uniquely corresponding to an updated firmware image. In one example implementation, the sentinel MAC address may form part of a set of unique identifiers included within the firmware update request.

[0065] At block 304, the updated firmware image may be identified from amongst a plurality of firmware images, corresponding to the sentinel MAC address included in the received firmware update request.

[0066] At block 306, the updating server 104 may transmit the updated firmware image to the computing system.

[0067] Fig. 4(a) and 4(b) illustrates computing environments 400 and 450 implementing non-transitory computer-readable mediums 402 and 452, respectively, according to an implementation of the present subject matter. In one implementation, the non-transitory computer readable medium 402 may be utilized by different computing systems 102 and the non-transitory computer readable medium 452 may be utilized by different updating servers 104. The computing system 102 and the updating server 104 may either be implemented in different computing environments, such as 400 and 450, or may be implemented in a common computing environment.

[0068] In one implementation, the computing environments 400 and 450 may include processing resource 404 and processing resource 454, communicatively coupled to the non-transitory computer readable medium 402 and computer readable medium 452, through communication link 406 and communication link 456, connecting to network 408 and network 458, respectively.

[0069] For example, the processing resource 404 may be implemented in a computing system, such as the computing system 102 described earlier. Similarly, the processing resource 454 may be implemented in an updating server, such as the updating server 104 described earlier. The computer readable medium 402 and the computer readable medium 452 may be, for example, an internal memory device or an external memory device. In one implementation, the communication link 406 and the communication link 456 may be a direct communication link, such as any memory read/write interface. In another implementation, the communication link 406 and the communication link 456 may be an indirect communication link, such as a network interface. In such a case, the processing resource 404 may access the computer readable medium 402 through the network 408, and the processing resource 454 may access the computer readable medium 452 through the network 458. The network 408 and the network 458 may either be a single network or a combination of multiple networks and may use a variety of different communication protocols.

[0070] Referring to Fig. 4(a), in one implementation, the computer readable medium 402 may include a set of computer readable instructions, including the communication module 118, and the recovery module 108. The set of computer readable instructions may be accessed by the processing resource 404 through the communication link 406 and subsequently executed to communicate with other entities such as the updating server 104, through the network 408.

[0071] In an example implementation of the present subject matter, the recovery module 108 may identify occurrence of an execution failure of a firmware image from amongst a plurality of firmware images. The recovery

module 108 may also determine a sentinel MAC address corresponding to the firmware image, where at least one sentinel MAC address is associated with each of the plurality of firmware images. The communication module 118 of the computer readable medium 402 may generate a firmware update request based on the sentinel MAC address and request an updated firmware image from the updating server 104.

[0072] Referring to Fig. 4(b), in one implementation, the computer readable medium 452 may include a set of computer readable instructions, including an updating module 460. The set of computer readable instructions may be accessed by the processing resource 454 through the communication link 456 and subsequently executed to communicate with other entities, such as the computing system 102, through the network 458.

[0073] In an example implementation of the present subject matter, the updating module 460 may receive a firmware update request from the computing system 102. The firmware update request may include a set of unique identifiers. The set of unique identifiers may be based on a sentinel MAC address. The updating module 460 may identify an updated firmware image from amongst a plurality of firmware images, corresponding to the set of unique identifiers. Further, the updating module 460 may also provide the updated firmware image to the computing system 102.

[0074] Although implementations of present subject matter have been described in language specific to structural features and/or methods, it is to be understood that the present subject matter is not limited to the specific features or methods described. Rather, the specific features and methods are disclosed and explained in the context of a few implementations for the present subject matter.

What is claimed is:

1. A computing system for updating a firmware image, the computing system comprising:

a processor;

a recovery module coupled to the processor to:

identify failure of execution of the firmware image from a plurality of firmware images; and

ascertain a user preference in response to the failure, wherein the user preference is to update the firmware image or execute an alternate firmware image from the plurality of firmware images,

wherein when the user preference is to update the firmware image, the recovery module is to:

determine a sentinel MAC address corresponding to the firmware image, wherein at least one MAC address is associated with each of the plurality of firmware images; and

a communication module coupled to the processor to send a firmware update request to an updating server and request an updated firmware image based on the sentinel MAC address.

2. The computing system as claimed in claim 1, wherein the recovery module is to identify the failure of execution of the firmware image due to one of a failed update of the firmware image and a bug in the firmware image.

3. The computing system as claimed in claim 1, wherein the communication module includes a set of unique identifiers in the firmware update request, and wherein the set of unique identifiers are based on the sentinel MAC address.

4. The computing system as claimed in claim 1, wherein the communication module sends a modified bootp request as the firmware update request, and wherein the modified bootp request includes the sentinel MAC address.

5. The computing system as claimed in claim 1, wherein when the user preference is to execute the alternate firmware image, the recovery module is to further:

provide, at least one alternate firmware image from the plurality of firmware images to a user, for selection; and

execute a selected firmware image from the at least one alternate firmware image.

6. The computing system as claimed in claim 1, wherein the recovery module is to further:

receive, from the updating server, the updated firmware image;

reprogram a flash memory associated with the updated firmware image; and

execute the updated firmware image.

7. A method for providing an updated firmware image to a computing system, the method comprising:

receiving a firmware update request from the computing system, wherein the firmware update request includes a sentinel MAC address;

identifying the updated firmware image from a plurality of firmware images, corresponding to the sentinel MAC address; and

transmitting the updated firmware image to the computing system.

8. The method as claimed in claim 7, wherein the firmware update request includes a set of unique identifiers corresponding to the updated firmware image.

9. The method as claimed in claim 7, wherein identifying the updated firmware image is based on querying a lookup table, and wherein the lookup table includes a mapping between a plurality of sentinel MAC addresses and different firmware images.

10. The method as claimed in claim 7, wherein the firmware update request is a modified bootp request, and wherein the modified bootp request includes the sentinel MAC address.

11. A non-transitory computer readable medium having a set of computer readable instructions that, when executed, cause a processor resource to:

identify occurrence of an execution failure of a firmware image from a plurality of firmware images;

determine a sentinel MAC address corresponding to the firmware image, wherein at least one sentinel MAC address is associated with each of the plurality of firmware images; and

request an updated firmware image from an updating server based on the sentinel MAC address.

12. The non-transitory computer readable medium as claimed in claim 11, wherein the computer readable instructions, when executed, further cause the processor resource to:

receive, from the updating server, the updated firmware image;

reprogram a flash memory associated with the updated firmware image; and

execute the updated firmware image.

13. The non-transitory computer readable medium as claimed in claim 11, wherein the computer readable instructions, when executed, cause the processor resource to determine the sentinel MAC address based on assessment of a user preference, wherein the user preference is to update the firmware image.

14. The non-transitory computer readable medium as claimed in claim 13, wherein the computer readable instructions, when executed, further cause the processor resource to:

determine, upon occurrence of the execution failure, the user preference to be indicative of executing an alternate firmware image from the plurality of images;

provide, at least one alternate firmware image from the plurality of firmware images to a user for selection; and

execute a selected firmware image from the at least one firmware image.

15. The non-transitory computer readable medium as claimed in claim 11, wherein the computer readable instructions, when executed, cause the processor resource to request the updated firmware image by sending a firmware update request to the updating server, and wherein the firmware update request includes a set of unique identifiers based on the determined sentinel MAC address.

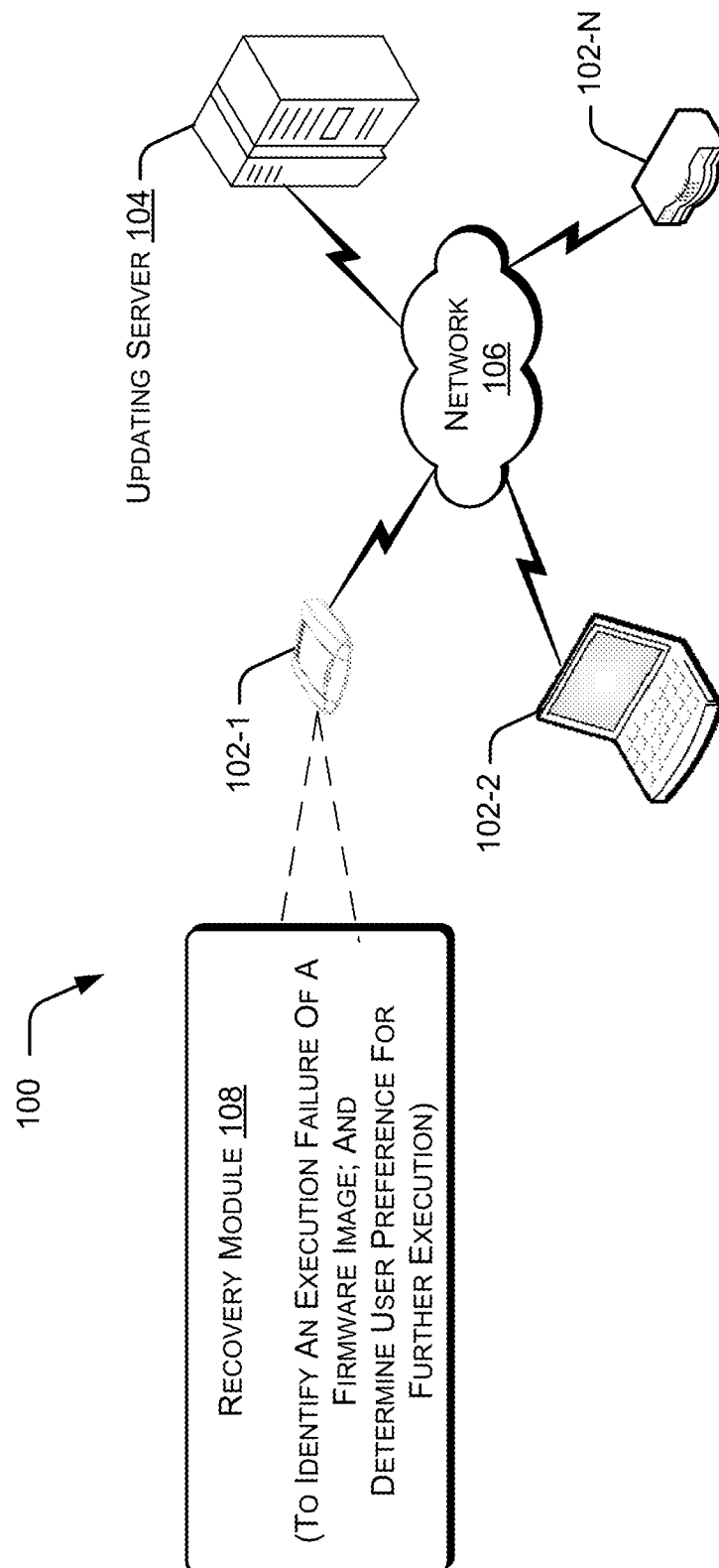


Fig. 1(a)

2/7

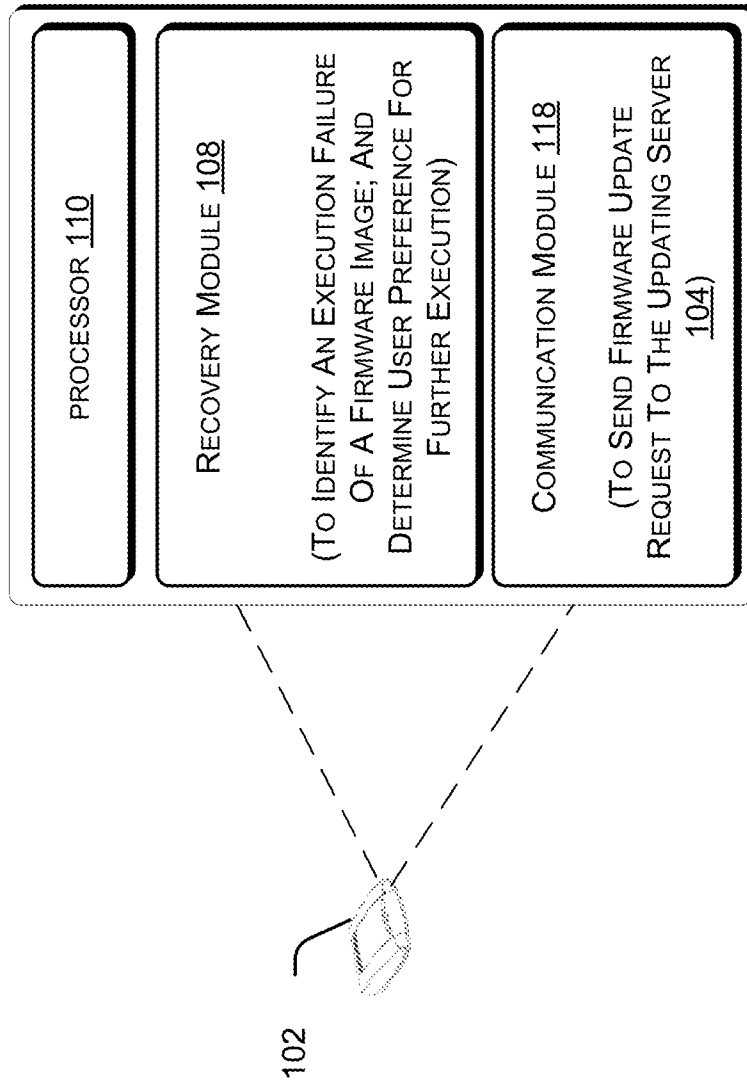


Fig. 1(b)

3/7

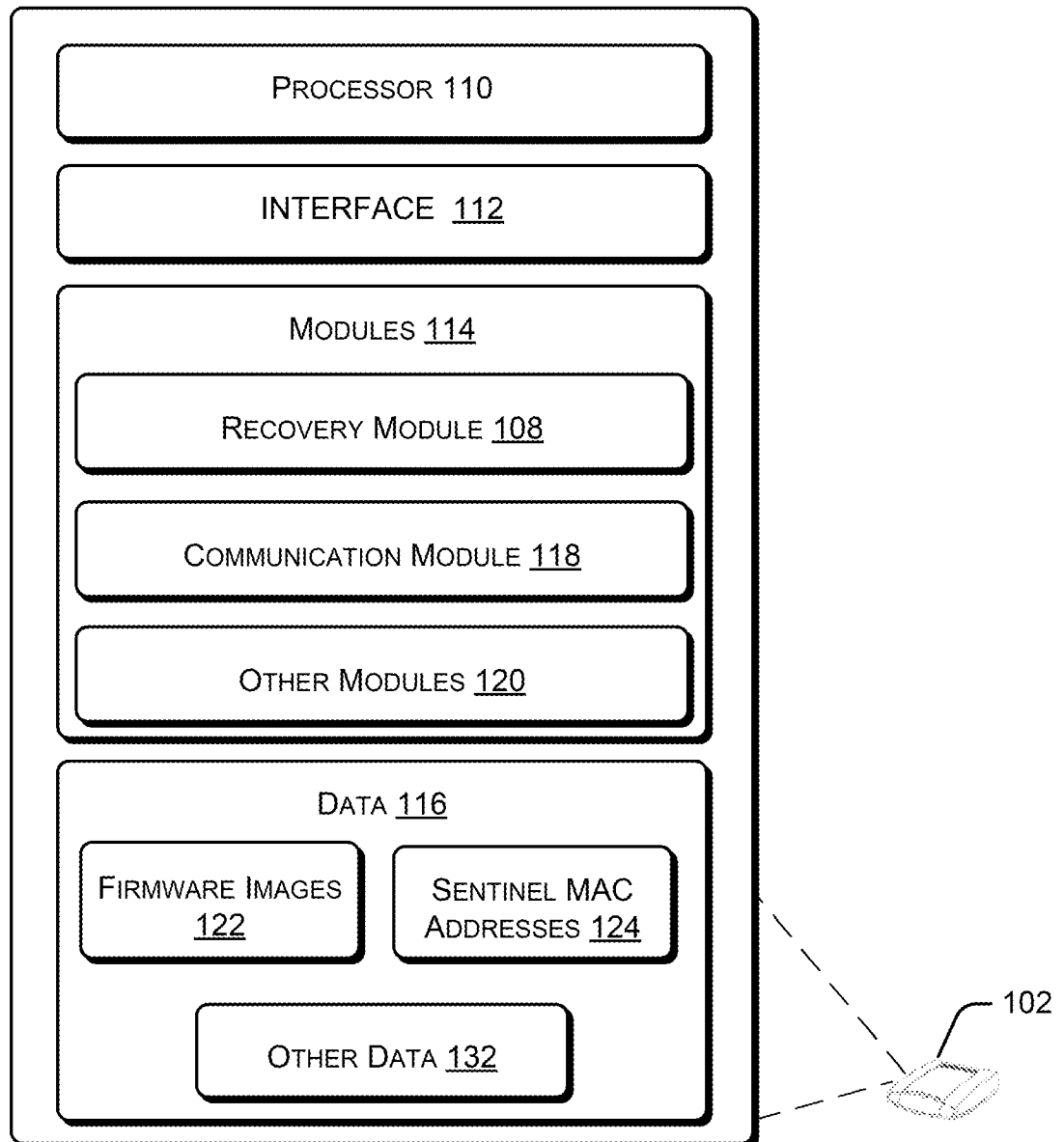


Fig. 1(c)

4/7

200

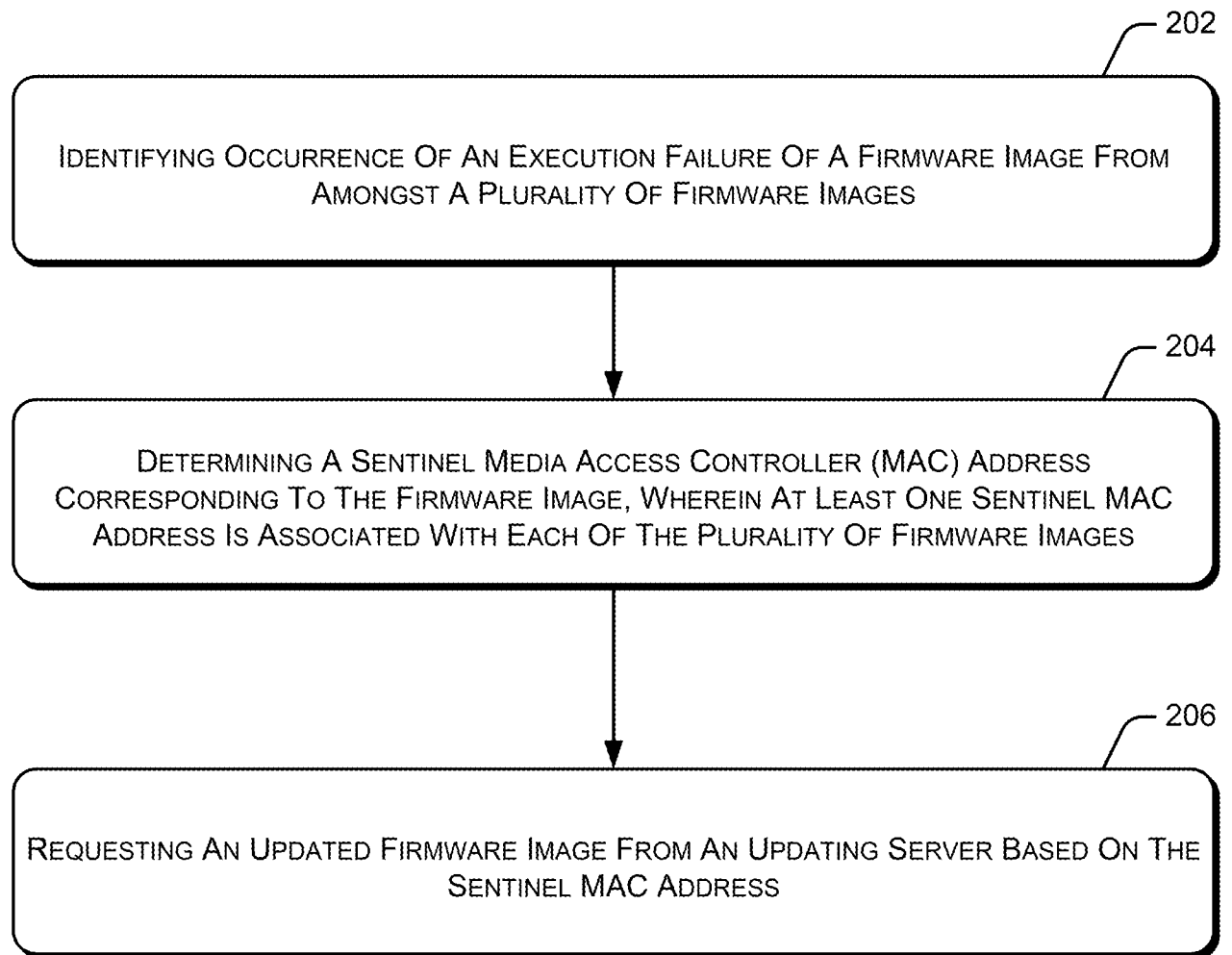


Fig. 2

5/7

300

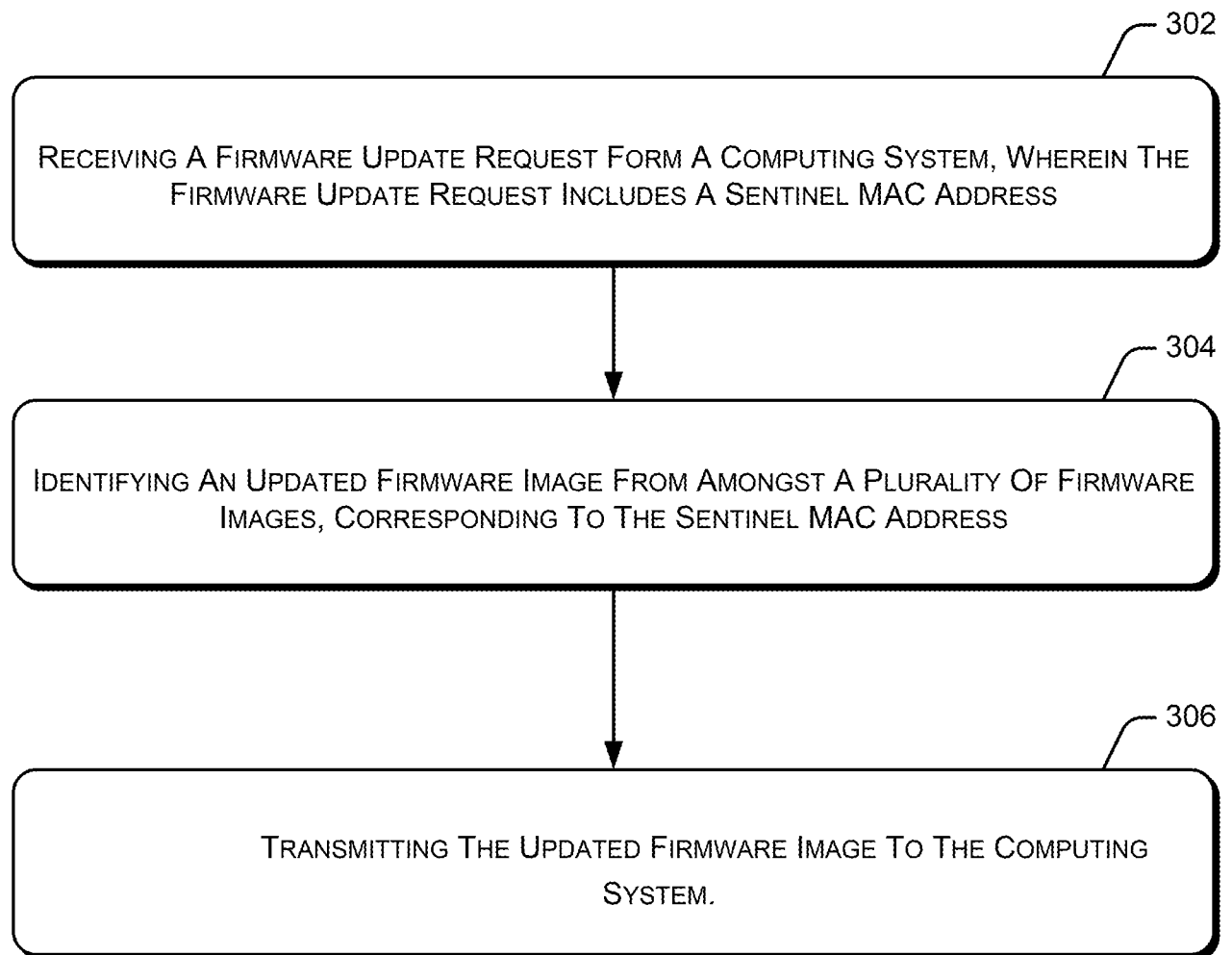


Fig. 3

400 →

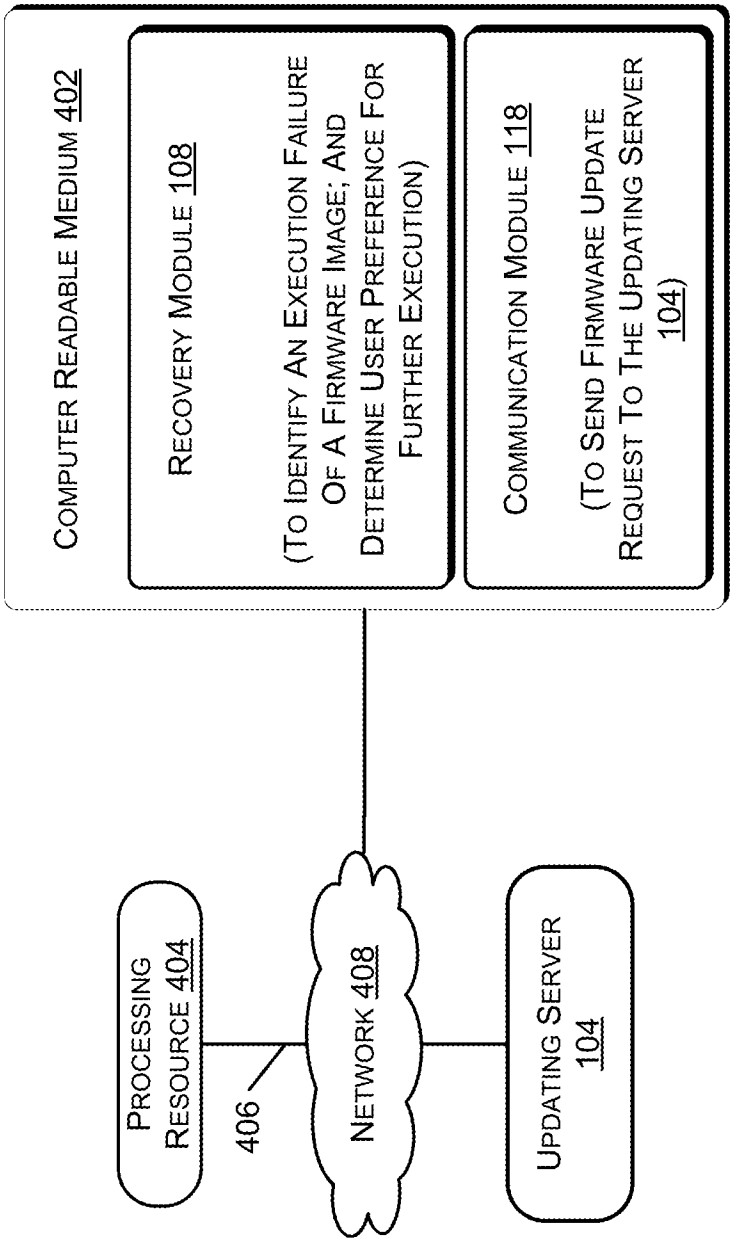


Fig. 4(a)

450 →

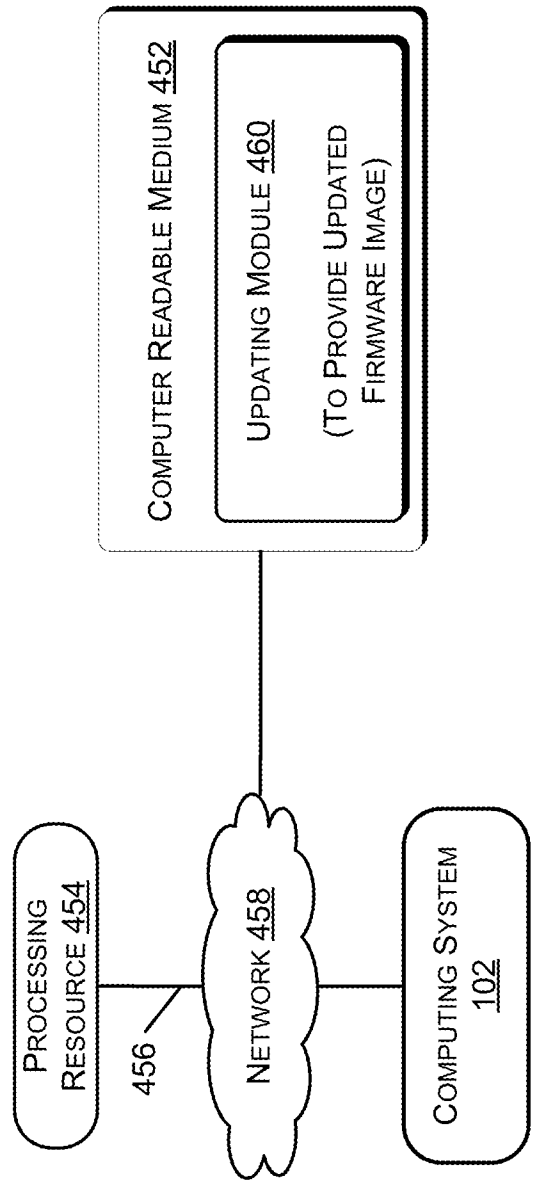


Fig. 4(b)

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2015/017811**A. CLASSIFICATION OF SUBJECT MATTER****G06F 9/445(2006.01)i, H04L 29/08(2006.01)i**

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)

G06F 9/445; G06F 15/16; G06F 9/44; G06F 11/14; G06F 15/177; H04L 29/08

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

Korean utility models and applications for utility models

Japanese utility models and applications for utility models

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

eKOMPASS(KIPO internal) & Keywords:firmware, update, failure, recovery

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 7702896 B1 (POLYUDOV, F.) 20 April 2010 See abstract; col. 7, line 47 - col. 9, line 2; fig. 7; and claims 1 and 15.	1-15
Y	US 2009-0249120 A1 (YAO, J. et al.) 01 October 2009 See abstract; paragraphs [0025]-[0026]; fig. 1; and claims 1 and 16.	1-15
A	US 7673301 B1 (RIGHI, S. et al.) 02 March 2010 See abstract; col. 4, line 11 - col. 5, line 62; and fig. 1.	1-15
A	US 2004-0054995 A1 (LEE, K.) 18 March 2004 See abstract; paragraphs [0019]-[0028]; and fig. 3.	1-15
A	US 2013-0125107 A1 (BANDAKKA, J. et al.) 16 May 2013 See abstract; paragraphs [0027]-[0038]; and fig. 1.	1-15



Further documents are listed in the continuation of Box C.



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) or which 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

17 November 2015 (17.11.2015)

Date of mailing of the international search report

17 November 2015 (17.11.2015)

Name and mailing address of the ISA/KR

International Application Division

Korean Intellectual Property Office

189 Cheongsa-ro, Seo-gu, Daejeon, 35208, Republic of Korea

Facsimile No. +82-42-472-7140

Authorized officer

YU, Jintae

Telephone No. +82-42-481-8530



INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2015/017811

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 7702896 B1	20/04/2010	US 8230210 B1	24/07/2012
US 2009-0249120 A1	01/10/2009	US 7865775 B2	04/01/2011
US 7673301 B1	02/03/2010	US 7421688 B1	02/09/2008
US 2004-0054995 A1	18/03/2004	CN 1215409 C	17/08/2005
		CN 1482538 A	17/03/2004
		KR 10-0442884 B1	02/08/2004
		KR 10-2004-0024412 A	20/03/2004
		US 7536685 B2	19/05/2009
US 2013-0125107 A1	16/05/2013	US 2015-020060 A1	15/01/2015
		US 8869138 B2	21/10/2014
		WO 2013-070295 A1	16/05/2013