



(22) Date de dépôt/Filing Date: 2006/05/24

(41) Mise à la disp. pub./Open to Public Insp.: 2006/12/29

(45) Date de délivrance/Issue Date: 2010/07/20

(30) Priorité/Priority: 2005/06/29 (EP05254068.9)

(51) Cl.Int./Int.Cl. *G06F 17/00* (2006.01),
G06F 1/24 (2006.01), *G06F 11/34* (2006.01),
G06F 9/445 (2006.01)

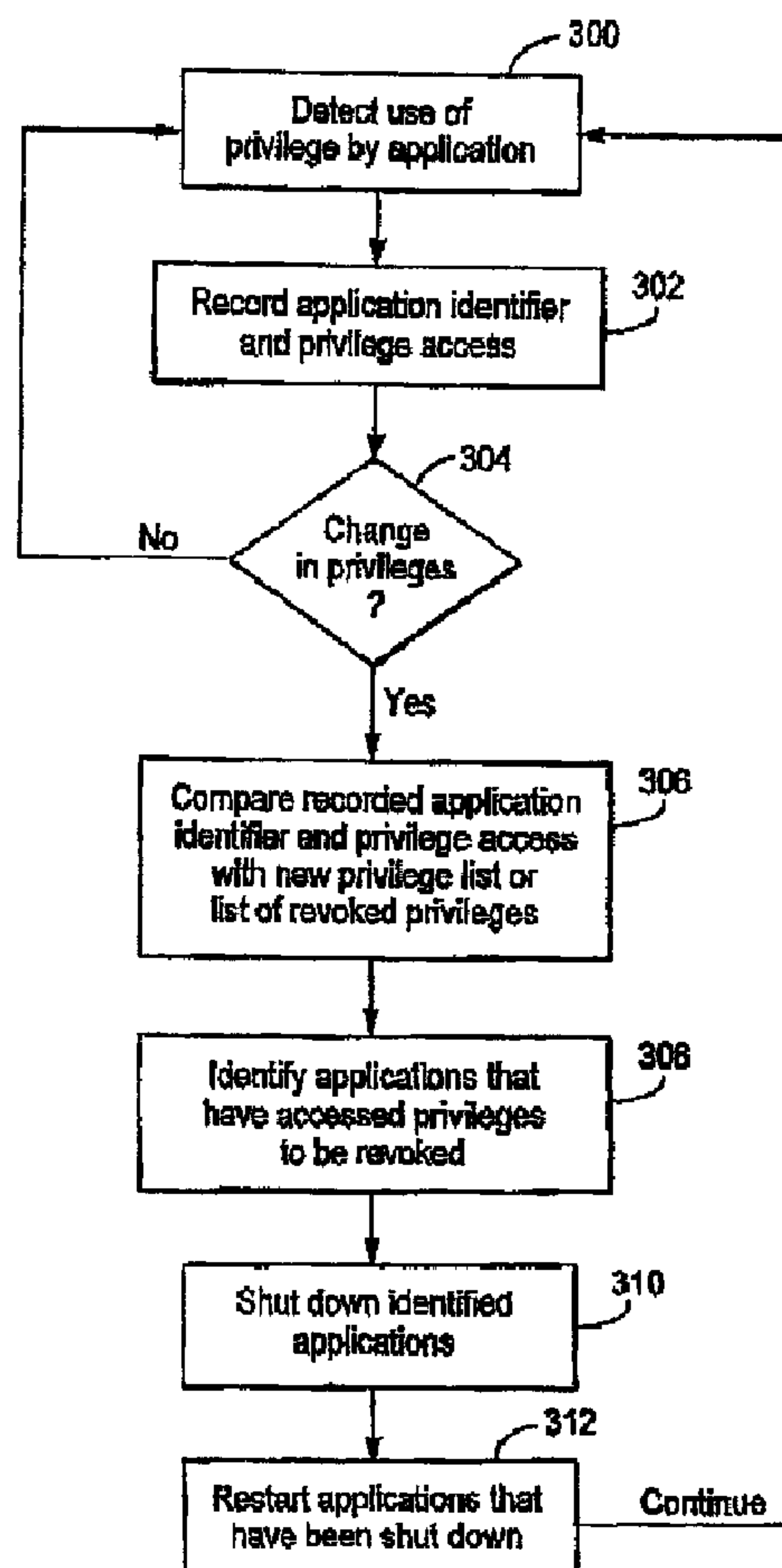
(72) Inventeurs/Inventors:
ADAMS, NEIL P., CA;
LITTLE, HERBERT A., CA

(73) Propriétaire/Owner:
RESEARCH IN MOTION LIMITED, CA

(74) Agent: BERESKIN & PARR LLP/S.E.N.C.R.L.,S.R.L.

(54) Titre : SYSTÈME ET METHODE DE GESTION ET DE REVOCATION DE PRIVILEGES

(54) Title: SYSTEM AND METHOD FOR PRIVILEGE MANAGEMENT AND REVOCATION



(57) Abrégé/Abstract:

The management of privileges associated with certain applications that are accessible by users of electronic equipment, such as networked computers, mobile wireless communications devices. In particular, systems and methods for managing privileges

(57) **Abrégé(suite)/Abstract(continued):**

associated with particular applications and for revoking these privileges in a timely and robust manner. When policies or application control changes, the system detects which privileges have been revoked for which applications, by comparing the old set of privileges with the new set of privileges. For each revoked privilege, the system determines if the application has ever accessed that privilege in the past, and, if so, the device is reset. To ensure that privileges that may be passed between applications are not overlooked, the device is arranged to perform a reset if any revoked privilege accessible by the device is one that may be passed between applications.

ABSTRACT

The management of privileges associated with certain applications that are accessible by users of electronic equipment, such as networked computers, mobile wireless communications devices. In particular, systems and methods for managing privileges associated with particular applications and for revoking these privileges in a timely and robust manner. When policies or application control changes, the system detects which privileges have been revoked for which applications, by comparing the old set of privileges with the new set of privileges. For each revoked privilege, the system determines if the application has ever accessed that privilege in the past, and, if so, the device is reset. To ensure that privileges that may be passed between applications are not overlooked, the device is arranged to perform a reset if any revoked privilege accessible by the device is one that may be passed between applications.

SYSTEM AND METHOD FOR PRIVILEGE MANAGEMENT AND REVOCATION

BACKGROUND

1. Field of the Invention

The present invention relates generally to the management of privileges associated with certain applications that are accessible by users of electronic equipment, such as, for example, networked computers, mobile wireless communications devices, and the like. In particular, the disclosure is directed to systems and methods for managing privileges associated with particular applications and for revoking these privileges in a timely and robust manner.

2. Related Art

It is well known that certain electronic equipment, such as, for example, networked computers, mobile wireless communications devices, and the like, include applications resident on such equipment that may have access to certain privileges that enable the applications to perform various functions. Typically, a system administrator may use IT policy and application control to set the privileges associated with various applications present on the equipment that is subject to the administrator's control. Examples of privileges may include, for example, allowing an application to use inter-process communication (IPC), enabling the opening of internal and external connections, enabling the injection of browser filters, enabling BluetoothTM functionality, enabling use of e-mail, enabling the use of personal information management (PIM) functionality, use of application program interface (API), etc. It is important for the system administrator be able to track which applications have access to which privileges, and to be able to revoke privileges on an as needed basis.

For example, if an application has access to a privilege, and the system administrator revokes that privilege, the application should immediately, or within a small window of time, be denied access to that privilege. In other words, the privilege should be revoked as soon as possible. Events that might trigger a revocation of privileges may

include, for example, an application being loaded before the IT administrator/application control data is present on the device, an application is discovered to be a rogue application, or company policy changes, resulting in limiting the use or availability of certain applications and/or privileges associated therewith.

Regardless of the reason for privilege revocation, such revocation must be accomplished in a secure manner and in a manner that prevents possible work arounds by malicious applications or individuals. In general, according to current privilege revocation schemes, privilege checking is typically performed on the first access to a privilege. For example, applications communicate with IPC using the application registry. Once an application has a reference (e.g., pointer) to the application registry, it is difficult to take this reference away from the application. In another example, if an application has passed some of its privileges to another application using IPC, conventional systems can detect that the first application has access to IPC, but there is no way to detect that the other application has been passed the privilege(s).

Therefore, there remains a need for a system and method for effectively managing privileges associated with applications, and in particular, when privilege revocation is required, to revoke these privileges in a timely and robust manner.

SUMMARY

In view of the foregoing, we have now identified an efficient, accurate, robust and easy to implement system and method for managing privileges associated with particular applications and for revoking these privileges in a timely and reliable manner

According to an exemplary embodiment an application having privileges that are to be revoked may be shut down. When this application is restarted, access to revoked privileges will be denied. In order to keep track of which applications need to be shut down and reset, the system must keep track of which applications have accessed which privileges. To enable monitoring of the privileges accessed by applications, each time an application uses a privilege, the system records the application identifier and which

privilege has been accessed. This may be done in any of a number of conventional methods, such as, for example, a data table listing the application identifier and corresponding accessed privilege(s) associated with the application identifier. When the system administrator, or any other authority, institutes a change in privileges, the system accesses the record of which applications have accessed which privileges. A comparison of accessed privileges with the privileges to be revoked is performed. Each application that has accessed a privilege that is to be revoked is identified by the system and then shut down. When the application is restarted, the application will not have access to any of the revoked privileges.

In another embodiment, a device reset may be performed. By resetting a device, the system is brought to a known state. According to this embodiment, a system administrator, or any other authority, specifies that a device must be reset when a new policy that revokes (or changes) privileges is instituted. A device reset may be instituted at any time the administrator or authority deems it necessary to do so. For example, a device reset may be done whenever a new policy that includes privilege revocation is instituted. However, this may result in numerous unnecessary device reset events that are potentially inconvenient to the user and may interfere with use of the device. Alternatively, the system may keep track of which devices include which applications, and use this information to determine a less intrusive device reset schedule. For example, resetting only those devices having applications that may potentially be affected by the policy change.

In yet another advantageous embodiment, aspects of previously described embodiments are combined to provide effective privilege management and revocation. According to this exemplary embodiment, the system administrator or other authority has no actual control over when a device is reset. The administrator merely manages the privileges of the system and particular applications. The device itself is responsible for resetting when necessary. In this embodiment, the device keeps track of which applications get access to which privileges. When policies or application control changes,

the system detects which privileges have been revoked for which applications. This can be accomplished by simply comparing the old set of privileges with the new set of privileges. For each revoked privilege for a given application, the system determines if the application has ever accessed that privilege in the past. As noted above, the system has been keeping track of these since the system was first started up. If an application has accessed a privilege that is now revoked at any time in the past, the device is reset. For some privileges (e.g., IPC), it still cannot be determined whether a privilege has been used as a result of being passed from another application. To overcome the potential for missing a revocation of privileges when applications pass privileges between themselves, if, for example, IPC (or any other privilege that is able to be passed between applications) is revoked from any application, regardless of if the system has detected that the application has accessed the privilege, the device must be reset. This reset will bring the device back to a known state. Device reset will only be done when necessary, thus limiting the number of resets and solving the problem associated with privileges that have been passed between applications.

Each of these embodiments is useful in a variety of privilege management environments ranging from a low-level low-priority where a modest level of privilege revocation security is needed, resulting in a less robust system, to a highly important ultra-robust environment where user inconvenience is secondary to the need for absolute certainty of privilege revocation.

BRIEF DESCRIPTION OF THE DRAWINGS

These and other embodiments along with their attendant advantages are described herein with reference to the following drawings in which like reference numerals refer to like elements, and wherein:

FIG. 1 is a block diagram showing a computer network and communication system in which electronic devices running applications having access to associated privileges are used;

FIG. 2 is a block diagram of a wireless mobile communication device as an example of an electronic device running applications having access to associated privileges;

FIG. 3 is a flow diagram illustrating a method of revoking privileges according to an exemplary embodiment;

FIG. 4 is a flow diagram illustrating a method of revoking privileges according to another exemplary embodiment; and

FIG. 5 is a flow diagram illustrating yet another method of revoking privileges according to another exemplary embodiment.

DETAILED DESCRIPTION OF EXEMPLARY EMBODIMENTS

FIG. 1 is a block diagram showing a computer network and communication system in which electronic devices running applications having access to associated privileges are used. The computer network 18 includes, for example, various networked computers 28 and, optionally, a message server 26, all linked via a Local Area Network (LAN) 30. The communication system includes a Wide Area Network (WAN) 12 coupled to a computer system 14, a wireless network gateway 16 and the LAN 30 of the computer network 18. The wireless network gateway 16 is also connected to a wireless communication network 20 in which a wireless mobile communication device 22 (hereinafter “mobile device”), is configured to operate. The entire system 10 is typically managed by, among others, a system administrator or like authority 32.

The computer system 14 may be a desktop or laptop personal computer that is configured to communicate with the WAN 12 or any other suitable network, such as, for example, the Internet. Personal computers, such as the computer system 14, typically access the Internet via an Internet Service Provider (ISP), Application Service Provider (ASP), or the like.

The LAN 30 is an example of a typical working computer network environment, in which multiple computers 28 are connected in a network. The computer network 18 is

typically located behind a security firewall 24. Within the LAN 30, a message server 26, operating on a computer behind the firewall 24 may act as the primary interface for the owner of the computer network 18 to exchange messages both within the LAN 30 and with other external messaging clients via the WAN 12. Known message servers include, for example, Microsoft OutlookTM, Lotus NotesTM, Yahoo!TM Messenger, AOL Instant Messenger, or any other client-server or peer-to-peer, or similar messaging clients with various architectures. Messages received by the message server 26 are distributed to mailboxes for user accounts addressed in the received messages, and are then accessed by a user through a messaging client operating on a computer system 28. The foregoing is merely an exemplary description illustrating a client-server architecture, and in no way implies that such architecture is necessary, as other suitable architectures known to those skilled in the art may be used.

Although only a message server 26 is shown in the LAN 30, those skilled in the art will appreciate that a LAN may include other types of servers supporting resources that are shared between the networked computer systems 28, and that the message server 26 may also provide additional functionality, such as dynamic database storage for data such as, but not limited to, calendars, to-do lists, task lists, e-mail and documentation. The message server 26 and electronic messaging are described for illustrative purposes only. Systems and methods for managing and revoking privileges are applicable to a wide range of electronic devices, and are in no way limited to electronic devices with messaging capabilities.

The wireless gateway 16 provides an interface to a wireless network 20, through which messages may be exchanged with a mobile device 22. Such functions as addressing of the mobile device 22, encoding or otherwise transforming messages for wireless transmission, and any other interface functions are performed by the wireless gateway 16. The wireless gateway 16 may be configured to operate with more than one wireless network 20, in which case the wireless gateway 16 also determines a most likely network

for locating a given mobile device 22 and possibly track mobile devices as users roam between countries or networks.

The mobile device 22 is, for example, a data communication device, a voice communication device, a dual-mode communication device such as many modern cellular telephones having both data and voice communications functionality, a multiple-mode device capable of voice, data and other types of communications, a personal digital assistant (PDA) enabled for wireless communications, or a laptop or desktop computer system with a wireless modem.

Any computer system with access to the WAN 12 may exchange messages with the mobile device 22 through the wireless network gateway 16. Alternatively, private wireless network gateways such as wireless Virtual Private Network (VPN) routers could be implemented to provide a private interface to a wireless network. A wireless VPN router implemented in the LAN 30 provides a private interface from the LAN 30 to one or more mobile devices such as 22 through the wireless network 20. A private interface to a mobile device 22 may also effectively be extended to entities outside the LAN 30 by providing a message forwarding or redirection system that operates with the message server 26. Such a message redirection system is disclosed in United States Patent No. 6,219,694, which is hereby incorporated into this application by reference. In this type of system, incoming messages received by the message server 26 and addressed to a user of a mobile device 22 are sent through the wireless network interface, either a wireless VPN router, the wireless gateway 16, or another interface, for example, to the wireless network 20 and to the user's mobile device 22. Another alternate interface to a user's mailbox on a message server 26 may be a Wireless Application Protocol (WAP) gateway. Through a WAP gateway, a list of messages in a user's mailbox on the message server 26, and possibly each message or a portion of each message, may be sent to the mobile device 22.

A wireless network 20 normally delivers messages to and from communication devices such as the mobile device 22 via RF transmissions between base stations and devices. The wireless network 20 may, for example, be a data-centric wireless network, a

voice-centric wireless network, or a dual-mode network that can support both voice and data communications over the same infrastructure. Recently developed networks include Code Division Multiple Access (CDMA) networks and General Packet Radio Service (GPRS) networks. So-called third-generation (3G) networks like Enhanced Data rates for Global Evolution (EDGE) and Universal Mobile Telecommunications Systems (UMTS) are currently under development. Older data-centric networks include, but are not limited to, the MobitexTM Radio Network ("Mobitex"), and the DataTACTM Radio Network ("DataTAC"). Voice-centric data networks such as Personal Communication System (PCS) networks, including Global System for Mobile Communications (GSM) and Time Division Multiple Access (TDMA) systems, have been available in North America and world-wide for several years.

FIG. 2 is a block diagram of an exemplary wireless mobile communication device as an example of an electronic device. However, it should be understood that the systems and methods disclosed herein may be used with many different types of devices, such as personal digital assistants (PDAs), desktop computers, or the like.

The mobile device 500 is preferably a two-way communication device having at least voice and data communication capabilities. The mobile device 500 preferably has the capability to communicate with other computer systems on the Internet. Depending on the functionality provided by the mobile device, the mobile device may be referred to as a data messaging device, a two-way pager, a cellular telephone with data messaging capabilities, a wireless Internet appliance, or a data communication device (with or without telephony capabilities). As mentioned above, such devices are referred to generally herein as mobile devices.

The mobile device 500 includes a transceiver 511, a microprocessor 538, a display 522, non-volatile memory 524, random access memory (RAM) 526, auxiliary input/output (I/O) devices 528, a serial port 530, a keyboard 532, a speaker 534, a microphone 536, a short-range wireless communications sub-system 540, and may also include other device sub-systems 542. The transceiver 511 preferably includes transmit and receive antennas

516, 518, a receiver (Rx) 512, a transmitter (Tx) 514, one or more local oscillators (LOs) 513, and a digital signal processor (DSP) 520. Within the non-volatile memory 524, the mobile device 500 includes a plurality of software modules 524A-524N that can be executed by the microprocessor 538 (and/or the DSP 520), including a voice communication module 524A, a data communication module 524B, and a plurality of other operational modules 524N for carrying out a plurality of other functions.

The mobile device 500 is preferably a two-way communication device having voice and data communication capabilities. Thus, for example, the mobile device 500 may communicate over a voice network, such as any of the analog or digital cellular networks, and may also communicate over a data network. The voice and data networks are depicted in FIG. 2 by the communication tower 519. These voice and data networks may be separate communication networks using separate infrastructure, such as base stations, network controllers, etc., or they may be integrated into a single wireless network. References to the network 519 should therefore be interpreted as encompassing both a single voice and data network and separate networks.

The communication subsystem 511 is used to communicate with the network 519. The DSP 520 is used to send and receive communication signals to and from the transmitter 514 and receiver 512, and also exchange control information with the transmitter 514 and receiver 512. If the voice and data communications occur at a single frequency, or closely-spaced set of frequencies, then a single LO 513 may be used in conjunction with the transmitter 514 and receiver 512. Alternatively, if different frequencies are utilized for voice communications versus data communications or the mobile device 500 is enabled for communications on more than one network 519, then a plurality of LOs 513 can be used to generate frequencies corresponding to those used in the network 519. Although two antennas 516, 518 are depicted in FIG. 2, the mobile device 500 could be used with a single antenna structure. Information, which includes both voice and data information, is communicated to and from the communication module 511 via a link between the DSP 520 and the microprocessor 538.

The detailed design of the communication subsystem 511, such as frequency band, component selection, power level, etc., is dependent upon the communication network 519 in which the mobile device 500 is intended to operate. For example, a mobile device 500 intended to operate in a North American market may include a communication subsystem 511 designed to operate with the Mobitex or DataTAC mobile data communication networks and also designed to operate with any of a variety of voice communication networks, such as AMPS, TDMA, CDMA, PCS, etc., whereas a mobile device 500 intended for use in Europe may be configured to operate with the GPRS data communication network and the GSM voice communication network. Other types of data and voice networks, both separate and integrated, may also be utilized with the mobile device 500.

Communication network access requirements for the mobile device 500 also vary depending upon the type of network 519. For example, in the Mobitex and DataTAC data networks, mobile devices are registered on the network using a unique identification number associated with each device. In GPRS data networks, however, network access is associated with a subscriber or user of the mobile device 500. A GPRS device typically requires a subscriber identity module ("SIM"), which is required in order to operate the mobile device 500 on a GPRS network. Local or non-network communication functions (if any) may be operable, without the SIM, but the mobile device 500 is unable to carry out functions involving communications over the network 519, other than any legally required operations, such as '911' emergency calling.

After any required network registration or activation procedures have been completed, the mobile device 500 is able to send and receive communication signals, preferably including both voice and data signals, over the network 519. Signals received by the antenna 516 from the communication network 519 are routed to the receiver 512, which provides for signal amplification, frequency down conversion, filtering, channel selection, etc., and may also provide analog to digital conversion. Analog to digital conversion of the received signal allows more complex communication functions, such as

digital demodulation and decoding, to be performed using the DSP 520. In a similar manner, signals to be transmitted to the network 519 are processed, including modulation and encoding, for example, by the DSP 520 and are then provided to the transmitter 514 for digital to analog conversion, frequency up conversion, filtering, amplification and transmission to the communication network 519 via the antenna 518. Although a single transceiver 511 is shown for both voice and data communications, in alternative embodiments, the mobile device 500 may include multiple distinct transceivers, such as a first transceiver for transmitting and receiving voice signals, and a second transceiver for transmitting and receiving data signals, or a first transceiver configured to operate within a first frequency band, and a second transceiver configured to operate within a second frequency band.

In addition to processing the communication signals, the DSP 520 also provides for receiver and transmitter control. For example, the gain levels applied to communication signals in the receiver 512 and transmitter 514 may be adaptively controlled through automatic gain control algorithms implemented in the DSP 520. Other transceiver control algorithms could also be implemented in the DSP 520 in order to provide more sophisticated control of the transceiver 511.

The microprocessor 538 preferably manages and controls the overall operation of the mobile device 500. Many types of microprocessors or microcontrollers could be used here, or, alternatively, a single DSP 520 could be used to carry out the functions of the microprocessor 538. Low-level communication functions, including at least data and voice communications, are performed through the DSP 520 in the transceiver 511. High-level communication applications, including the voice communication application 524A, and the data communication application 524B are stored in the non-volatile memory 524 for execution by the microprocessor 538. For example, the voice communication module 524A may provide a high-level user interface operable to transmit and receive voice calls between the mobile device 500 and a plurality of other voice devices via the network 519. Similarly, the data communication module 524B may provide a high-level user interface

operable for sending and receiving data, such as e-mail messages, files, organizer information, short text messages, etc., between the mobile device 500 and a plurality of other data devices via the network 519.

The microprocessor 538 also interacts with other device subsystems, such as the display 522, RAM 526, auxiliary I/O devices 528, serial port 530, keyboard 532, speaker 534, microphone 536, a short-range communications subsystem 540 and any other device subsystems generally designated as 542. For example, the modules 524A-N are executed by the microprocessor 538 and may provide a high-level interface between a user of the mobile device and the mobile device. This interface typically includes a graphical component provided through the display 522, and an input/output component provided through the auxiliary I/O devices 528, keyboard 532, speaker 534, or microphone 536. Additionally, the microprocessor 538 is capable of running a variety of applications that may be present in the device non-volatile memory 524, including applications that have access to various privileges, as will be described in more detail herein.

Some of the subsystems shown in FIG. 2 perform communication-related functions, whereas other subsystems may provide "resident" or on-device functions. Notably, some subsystems, such as keyboard 532 and display 522 may be used for both communication-related functions, such as entering a text message for transmission over a data communication network, and device-resident functions such as a calculator or task list or other PDA type functions.

Operating system software used by the microprocessor 538 is preferably stored in a persistent store such as the non-volatile memory 524. In addition to the operating system and communication modules 524A-N, the non-volatile memory 524 may include a file system for storing data. The non-volatile memory 524 may also include data stores for owner information and owner control information. The operating system, specific device applications or modules, or parts thereof, may be temporarily loaded into a volatile store, such as RAM 526 for faster operation. Moreover, received communication signals may also be temporarily stored to RAM 526, before permanently writing them to a file system

located in the non-volatile memory 524. The non-volatile memory 524 may be implemented, for example, with Flash memory, non-volatile RAM, or battery backed-up RAM.

An exemplary application module 524N that may be loaded onto the mobile device 500 is a PIM application providing PDA functionality, such as calendar events, appointments, and task items. This module 524N may also interact with the voice communication module 524A for managing phone calls, voice mails, etc., and may also interact with the data communication module 524B for managing e-mail communications and other data transmissions. Alternatively, all of the functionality of the voice communication module 524A and the data communication module 524B may be integrated into the PIM module.

The non-volatile memory 524 preferably provides a file system to facilitate storage of PIM data items on the device. The PIM application preferably includes the ability to send and receive data items, either by itself, or in conjunction with the voice and data communication modules 524A, 524B, via the wireless network 519. The PIM data items are preferably seamlessly integrated, synchronized and updated, via the wireless network 519, with a corresponding set of data items stored or associated with a host computer system, thereby creating a mirrored system for data items associated with a particular user.

The mobile device 500 is manually synchronized with a host system by placing the mobile device 500 in an interface cradle, which couples the serial port 530 of the mobile device 500 to a serial port of the host system. The serial port 530 may also be used to insert owner information and owner control information onto the mobile device 500 and to download other application modules 524N for installation on the mobile device 500. This wired download path may further be used to load an encryption key onto the mobile device 500 for use in secure communications, which is a more secure method than exchanging encryption information via the wireless network 519.

Owner information, owner control information and additional application modules 524N may be loaded onto the mobile device 500 through the network 519, through an

auxiliary I/O subsystem 528, through the short-range communications subsystem 540, or through any other suitable subsystem 542, and installed by a user in the non-volatile memory 524 or RAM 526. Such flexibility in application installation increases the functionality of the mobile device 500 and may provide enhanced on-device functions, communication-related functions, or both. For example, secure communication applications may enable electronic commerce functions and other such financial transactions to be performed using the mobile device 500.

When the mobile device 500 is operating in a data communication mode, a received signal, such as a text message or a web page download, will be processed by the transceiver 511 and provided to the microprocessor 538, which preferably further processes the received signal for output to the display 522, or, alternatively, to an auxiliary I/O device 528. Owner information, owner control information, commands or requests related to owner information or owner control information, and software applications received by the transceiver 511 are processed as described above. A user of mobile device 500 may also compose data items, such as email messages, using the keyboard 532, which is preferably a complete alphanumeric keyboard laid out in the QWERTY style, although other styles of complete alphanumeric keyboards such as the known DVORAK style may also be used. User input to the mobile device 500 is further enhanced with the plurality of auxiliary I/O devices 528, which may include a thumbwheel input device, a touchpad, a variety of switches, a rocker input switch, etc. The composed data items input by the user are then transmitted over the communication network 519 via the transceiver 511.

When the mobile device 500 is operating in a voice communication mode, the overall operation of the mobile device 500 is substantially similar to the data mode, except that received signals are output to the speaker 534 and voice signals for transmission are generated by a microphone 536. In addition, the secure messaging techniques described above might not necessarily be applied to voice communications. Alternative voice or audio I/O devices, such as a voice message recording subsystem, may also be implemented on the mobile device 500. Although voice or audio signal output is

accomplished through the speaker 534, the display 522 may also be used to provide an indication of the identity of a calling party, the duration of a voice call, or other voice call related information. For example, the microprocessor 538, in conjunction with the voice communication module 524A and the operating system software, may detect the caller identification information of an incoming voice call and display it on the display 522.

A short-range communications subsystem 540 is also be included in the mobile device 500. For example, the subsystem 540 may include an infrared device and associated circuits and components, or a Bluetooth or 802.11 short-range wireless communication module to provide for communication with similarly-enabled systems and devices. Thus, owner information insertion, owner control information insertion, and application loading operations as described above may be enabled on the mobile device 500 via the serial port 530 or other short-range communications subsystem 540.

FIG. 2 represents a specific example of an electronic device in which owner control systems and methods described herein may be implemented. Implementation of such systems and methods in other electronic devices having further, fewer, or different components than those shown in FIG. 2 would occur to one skilled in the art to which this application pertains and are therefore considered to be within the scope of the present application.

FIG. 3 is a flow diagram illustrating a method of revoking privileges according to an exemplary embodiment. In this example, an application having privileges that are to be revoked may be shut down. When the application is restarted, access to revoked privileges will be denied. In order to keep track of which applications need to be shut down and reset, the system must keep track of which applications have access to which privileges. To accomplish this, for example, the system monitors and detects use of privileges by applications 300. The system may record an application identifier associated with a particular application and which privilege has been accessed by the application 302. This may be accomplished in any number of conventional methods that are readily apparent to those skilled in the art. For example, a data table listing the application

identifier and having pointers directed to privilege(s) accessed by the application or associated with the application identifier. The system then continues to monitor the system for any change, for example, a change in IT policy, that might result in a revocation of privileges 304. So long as no such change is detected in decision block 304, the system continues to monitor and keep track of applications and associated privileges.

Upon a detection of a change that would result in revocation of privileges being accessed by applications in the system by the decision block 304, such as, for example, a change in IT policy, a comparison of the recorded data relating to the accessed privileges with a listing of the new privileges is performed 306. For example, the application identifiers and associated accessed privileges recorded in step 302 are compared to the new privilege list, or to a list of revoked privileges 306. The changes discussed herein are typically instituted and administered by a system administrator or other authority who has responsibility for operation and management of the system.

As a result of the comparison 306, each application that has accessed a privilege(s) to be revoked is identified 308 by the system. Upon identification of these applications, the system implements a shut down of these identified applications 310. When these applications are restarted 312, the applications will not have access to any of the revoked privileges. The system will continue to monitor and detect the accessing of privileges and associated applications as described above.

In another embodiment, as illustrated in the flow diagram of FIG. 4, a device reset may be performed. According to this example, all devices in the system are monitored 400. A system administrator or other authority specifies, for example, that whenever there is a change in system policy that requires revocation or changes in privileges 402, all devices in the system must be reset 404. Resetting the devices brings the system to a known state, i.e., a state in which the system knows which applications have access to which privileges throughout the system. Upon restarting these applications 406 after the device reset 404, the applications will no longer have access to any of the revoked privileges. According to this example, a device reset may be instituted any time the

system administrator or authority deems it necessary to do so. For example, a device reset may be done whenever a new policy that includes privilege revocation is instituted. As a result, this solution may invoke numerous (potentially unnecessary) device reset events that may be intrusive and inconvenient for the users. However, this embodiment provides very robust and timely privilege revocation, and is thus suitable to highly secure systems where privilege management is more important than user convenience.

Turning now to FIG. 5, another advantageous embodiment implementing features of both embodiments described above with respect to FIGS. 3 and 4 is illustrated. According to this example, aspects of the previously described embodiments are combined to provide highly effective and timely privilege management and revocation. In this example, the system administrator or authority has no actual control over when a device is reset. The administrator or authority merely manages the privileges of the system and of particular applications. The device itself is responsible for resetting as needed.

In this example, the device monitors which applications of the device have access to which privileges 600, and a log of privileges for the device is kept 602. The device monitors whether policies or application control changes are made in the system 604. If there is no change detected 604, the device continues to monitor applications and keep a log of privileges for the device 600, 602. If a change in policy or application control is detected in step 604, the system determines which privileges have been revoked for which applications by comparing the old set of privileges in the log with the new set of privileges received from the system administrator 606. The device then determines if any revoked privileges are present on the device 608. If revoked privileges are detected, e.g., if an application has accessed a privilege at any time in the past that has now been revoked, the device will reset 610. As described above, resetting the device brings the system to a known state in which all applications and privileges are known. After the device is reset 610, it is restarted 612. Upon restart 612, the device applications will have access to the correct privileges. Advantageously, if no revoked privileges are detected in step 608, the device performs another check to ensure that no privileges that have been passed between

application have been missed. As explained above, for some privileges, e.g., IPC, it cannot be determined whether a privilege has been used as a result of being passed from another application. To overcome the potential for missing revocation of privileges when applications pass privileges between themselves, the system checks for privileges that are able to be passed between applications (e.g., IPC) 614. If a privilege that is able to be passed between applications is revoked from any application, regardless of if the system has detected that the application has accessed the privilege 614, the device must be reset 610 to bring the system to a known state. After resetting, the device is restarted 612, and will now have the only have access to the correct privileges. In this manner, device reset will only be performed when necessary, thus limiting the number of resets and solving the problem associated with privileges that go undetected due to their ability to be passed between applications.

While this disclosure describes specific exemplary embodiments, it is evident that many alternatives, modifications and variations will be apparent to those skilled in the art. Accordingly, the exemplary embodiments described herein, are intended to be illustrative, not limiting. Various changes may be made without departing from the true spirit and full scope of the invention, as defined in the following claims.

Claims:

1. A method for managing privileges associated with applications in an electronic system, comprising:

5 monitoring a plurality of devices in the electronic system to detect use of privileges by applications being run by any of the plurality of devices in the electronic system;

recording which applications have accessed which privileges;

detecting a change in privileges of the system;

10 comparing recorded privileges previously accessed by applications with a set of privilege changes;

identifying any privileges to be revoked based on a result of said comparing step;

when a privilege to be revoked is identified, shutting down an application that has previously accessed the revoked privilege; and

15 restarting any application that was shut down for having accessed a revoked privilege.

2. A method according to claim 1, wherein said step of recording, comprises:

storing an application identifier associated with an application that has accessed a privilege; and

20 associating said identifier with the privilege accessed by the application.

3. A method according to claim 2, wherein said recording is stored in a data table listing the application identifier and privileges associated therewith.

4. A method according to any one of claims 1 to 3, wherein said set of
25 privilege changes comprises privileges to be revoked.

5. A method according to any one of claims 1 to 3, wherein said set of privilege changes comprises a listing of all privileges available for use, and does not include privileges to be revoked.

6. A method according to any one of claims 1 to 5, wherein said devices comprise at least one wireless mobile electronic communication device.

7. A method according to any one of claims 1 to 6, further comprising continuously repeating the steps of the method.

5 8. A method according to any one of claims 1 to 7, wherein said change in privileges is a result of a change in a system IT policy.

9. A method for managing privileges in a system comprising a plurality of electronic devices, said method comprising:

10 monitoring, by each device of the system, which applications on each device have access to which privileges;
recording privileges accessible via applications resident on each device of the system;
detecting a change in privileges available to said devices in said system;
15 comparing an old set of privileges with a new set of privileges, said new set of privileges corresponding to said change in privileges;
determining if any privileges are to be revoked based on said comparison;
20 resetting each device that has access to a privilege that is to be revoked;
determining whether any privileges to be revoked are able to be passed between applications, and
when a privilege to be revoked is able to be passed between applications, resetting each device that has access to said privilege.

25 10. A method according to claim 9, wherein said change in privileges is a result of change in a system IT policy.

11. A method according to claim 9 or claim 10, wherein said change in privileges includes an indication that privileges are to be revoked.

12. A method according to any one of claims 9 to 11, wherein said step of resetting said electronic devices brings said system to a known state.

13. A method according to claim 12, further comprising restarting said electronic devices after said devices have been reset.

5 14. A method according to any one of claims 9 to 13, wherein said electronic devices comprise at least one mobile wireless communication device.

15. A system for managing privileges in a network comprising a plurality of electronic devices, said system comprising:

10 a processor resident in each device of the system, for monitoring which applications on each device have access to which privileges;
a memory for recording privileges accessible via applications resident on each device of the system; and
wherein said processor is operable to:

15 detect a change in privileges available to said devices in said system;

compare an old set of privileges with a new set of privileges, said new set of privileges corresponding to said change in privileges;

determine if any privileges are to be revoked based on said comparison;

20 reset each device that has access to a privilege that is to be revoked; and

25 determine whether any privileges to be revoked are able to be passed between applications; and when a privilege to be revoked is able to be passed between applications, resetting each device that has access to said privilege.

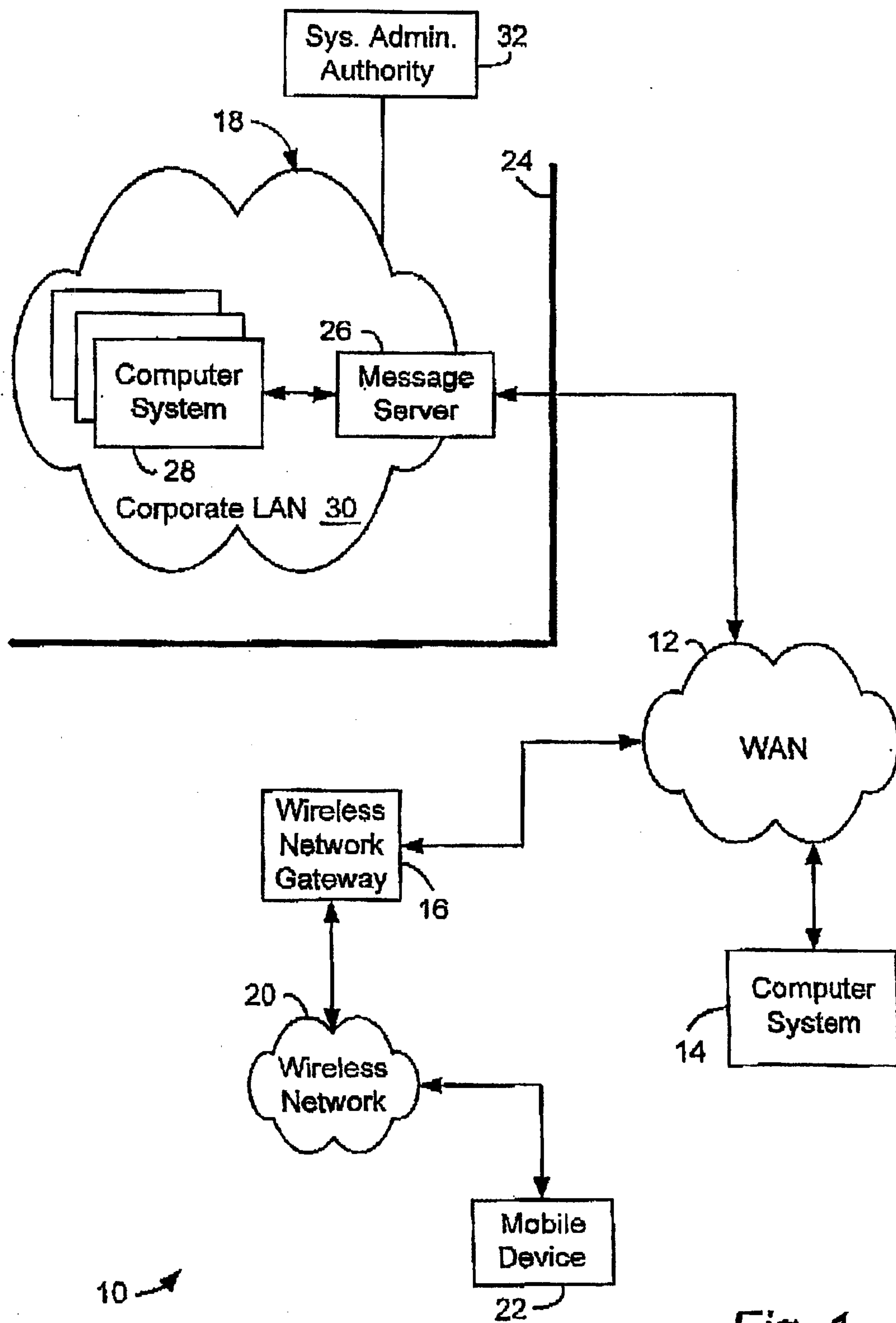


Fig. 1

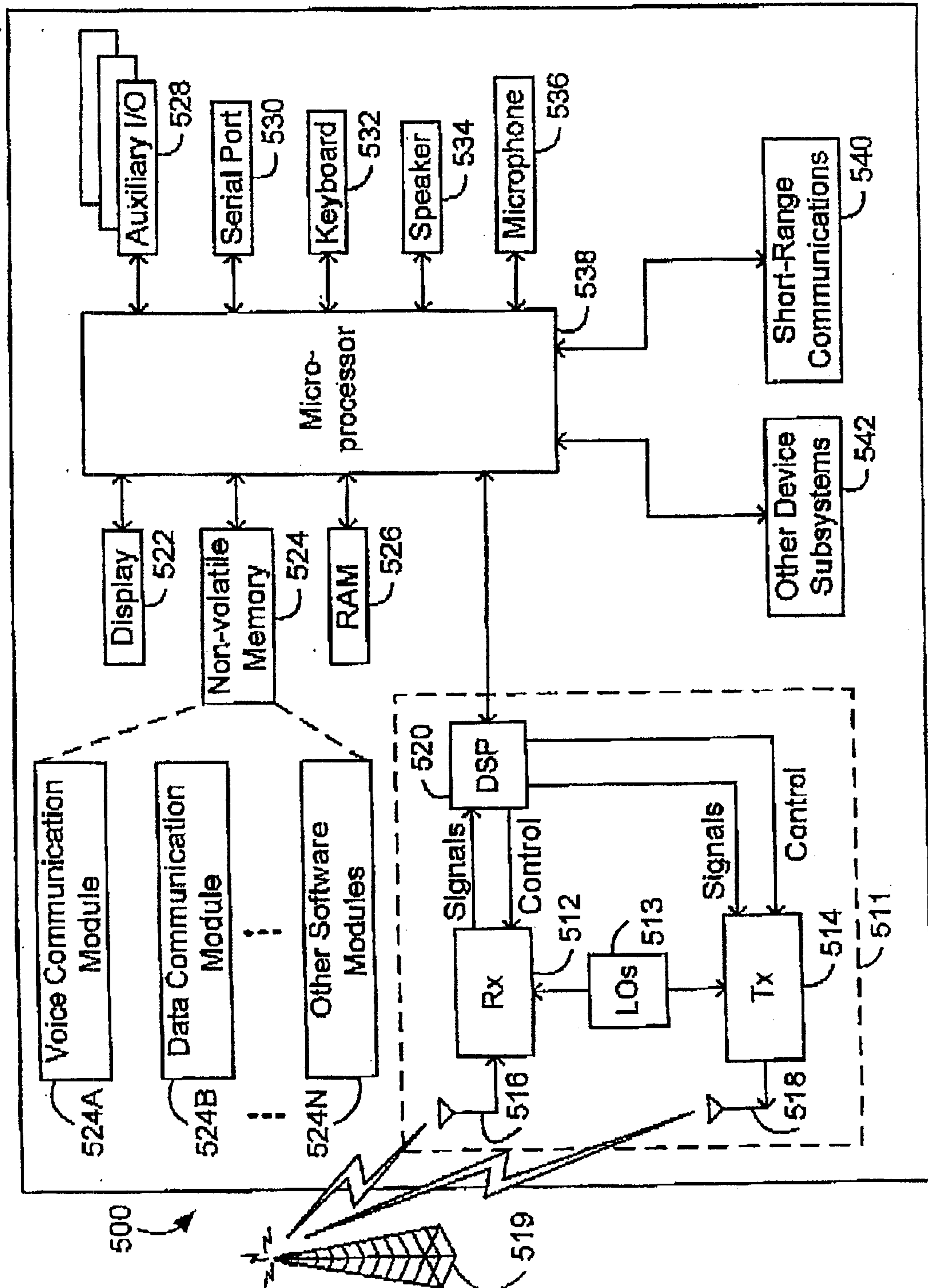
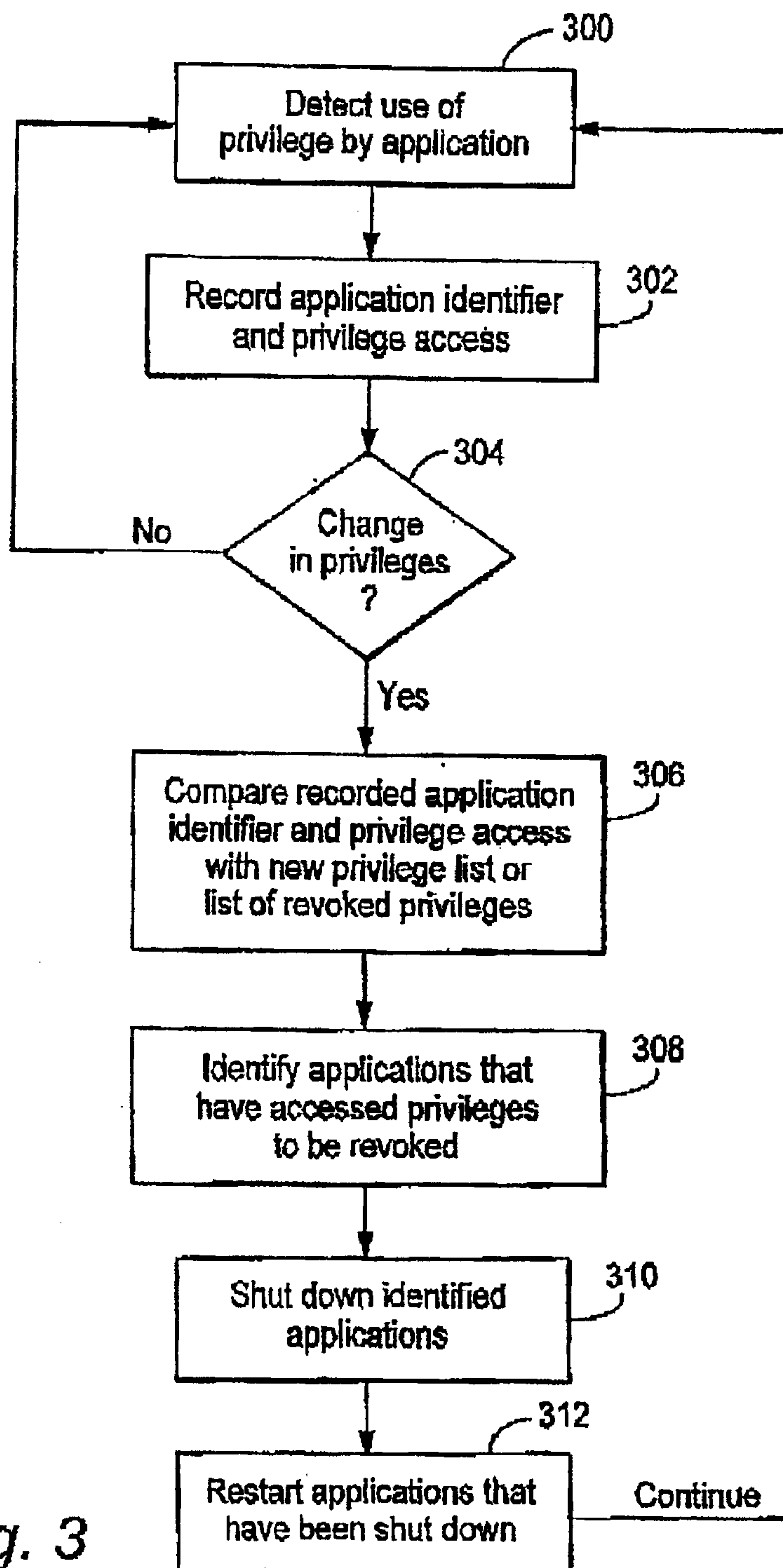
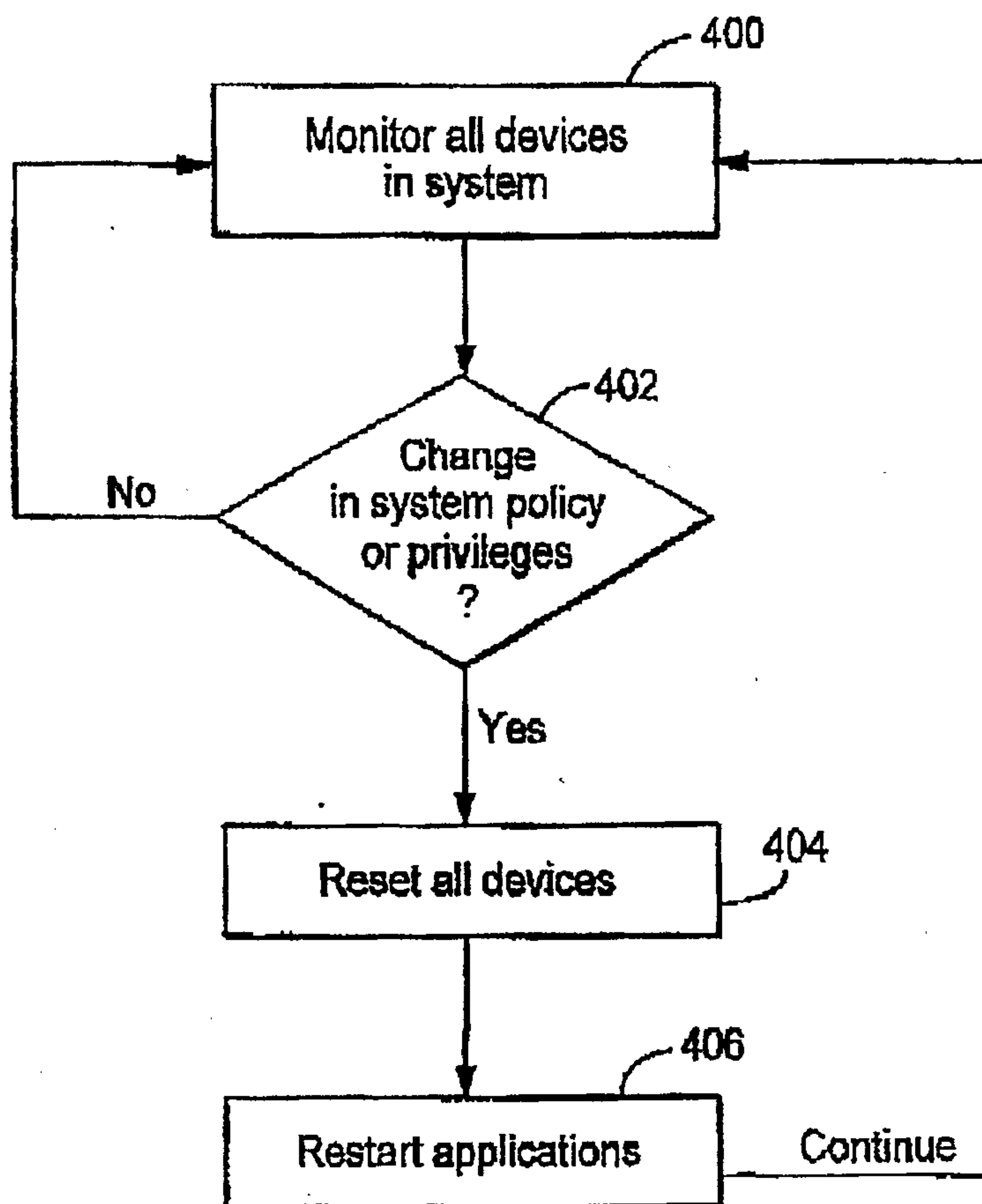


Fig. 2

*Fig. 3*

*Fig. 4*

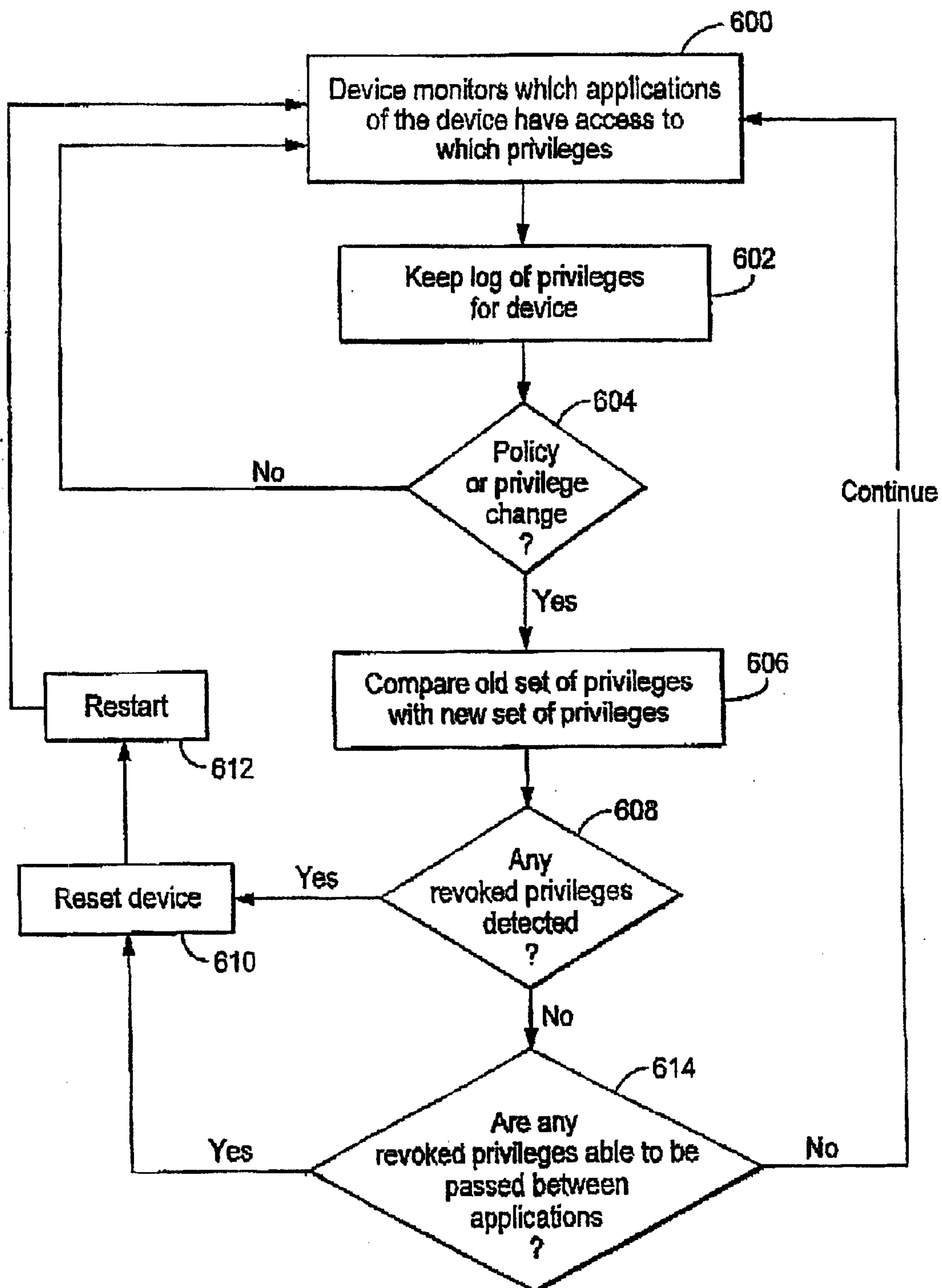


Fig. 5

