Office de la Propriété Intellectuelle du Canada
Un organisme d'Industrie Canada

Canadian Intellectual Property Office
An agency of Industry Canada

CA 2479526 A1 2003/09/25

(21) 2 479 526

(12) DEMANDE DE BREVET CANADIEN
CANADIAN PATENT APPLICATION

(13) A1

(86) Date de dépôt PCT/PCT Filing Date: 2003/03/20
(87) Date publication PCT/PCT Publication Date: 2003/09/25
(85) Entrée phase nationale/National Entry: 2004/09/16
(86) N° demande PCT/PCT Application No.: CA 2003/000402
(87) N° publication PCT/PCT Publication No.: 2003/079196
(30) Priorité/Priority: 2002/03/20 (60/365,515) US

(51) Cl.Int.^{7}/Int.Cl.^{7} G06F 12/02, G06F 12/14

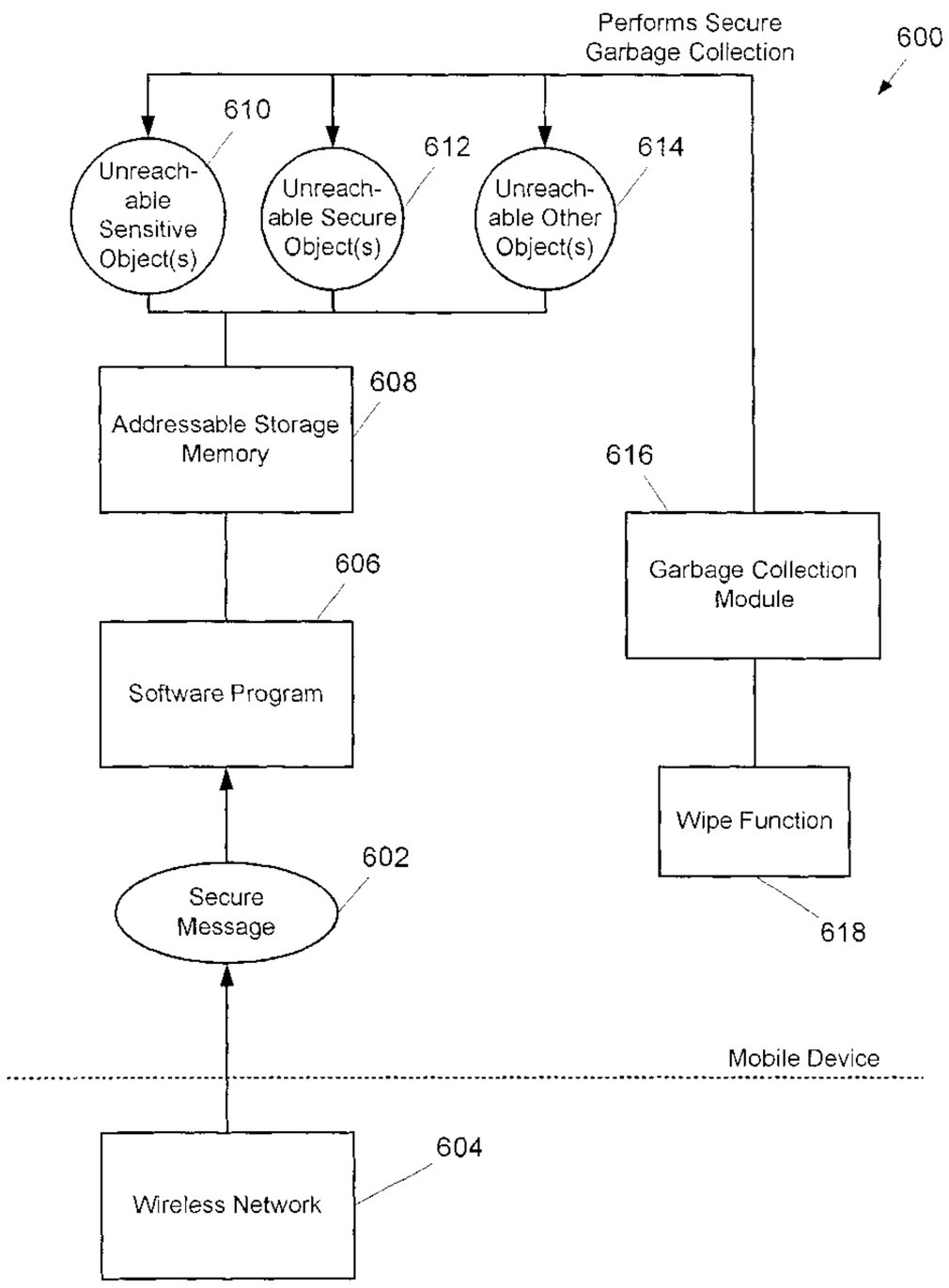
(71) Demandeur/Applicant:
RESEARCH IN MOTION LIMITED, CA

(72) Inventeurs/Inventors:
ADAMS, NEIL P., CA;
DAHMS, JOHN F. A., CA;
JANHUNEN, STEFAN E., CA;
LITTLE, HERBERT A., CA

(74) Agent: BORDEN LADNER GERVAIS LLP

(54) Titre : SYSTEME ET PROCEDE DE RECUPERATION DE L'ESPACE MEMOIRE PROTEGE DANS UN DISPOSITIF MOBILE

(54) Title: SYSTEM AND METHOD OF SECURE GARBAGE COLLECTION ON A MOBILE DEVICE



(57) Abrégé/Abstract:
A method and system for performing garbage collection involving sensitive information on a mobile device. Secure information is received at a mobile device over a wireless network. The sensitive information is extracted from the secure information. A software program operating on the mobile device uses an object to access the sensitive information. Secure garbage collection is performed upon the object after the object becomes unreachable.

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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
25 September 2003 (25.09.2003)

PCT

(10) International Publication Number
WO 2003/079196 A3

(51) International Patent Classification[7]: G06F 12/02, 12/14

(21) International Application Number: PCT/CA2003/000402

(22) International Filing Date: 20 March 2003 (20.03.2003)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/365,515 20 March 2002 (20.03.2002) US

(71) Applicant *(for all designated States except US)*: **RESEARCH IN MOTION LIMITED** [CA/CA]; 295 Phillip Street, Waterloo, Ontario N2L 3W8 (CA).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **LITTLE, Herbert, A.** [CA/CA]; 504 Old Oak Place, Waterloo, Ontario N2T 2V8 (CA). **ADAMS, Neil, P.** [CA/CA]; 550 Little Dover Cres., Waterloo, Ontario N2K 4E4 (CA). **JANHUNEN, Stefan, E.** [CA/CA]; 249 Cedarbrae Ave, Unit 10, Waterloo, Ontario N2K 4S8 (CA). **DAHMS, John, F., A.** [CA/CA]; 296 Castlefield Avenue, Waterloo, Ontario N2K 2N1 (CA).

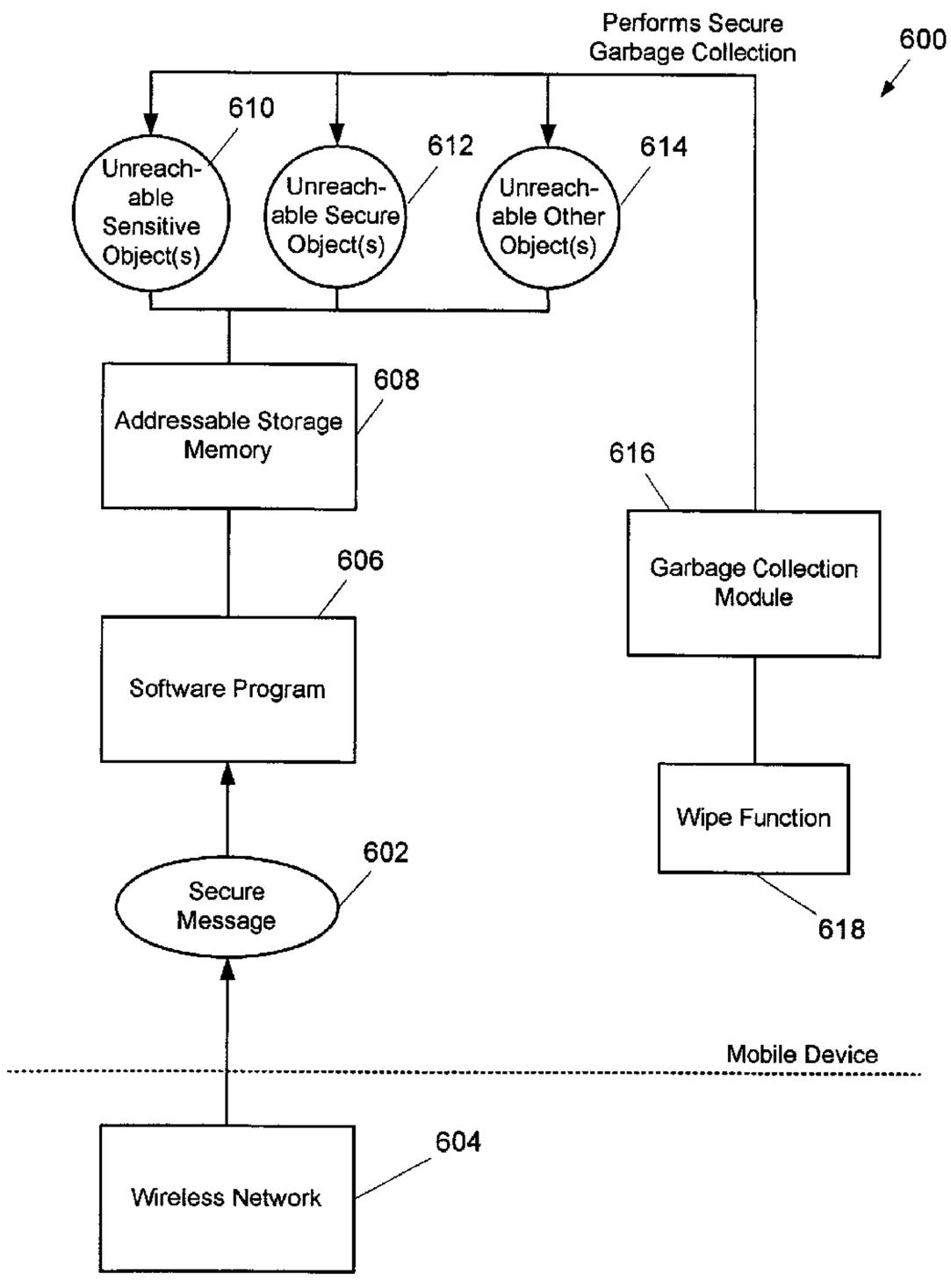
(74) Agents: **PATHIYAL, Krishna, K.** et al.; Research In Motion Limited, 295 Phillip Street, Waterloo, Ontario N2L 3W8 (CA).

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

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW),

[Continued on next page]

(54) Title: SYSTEM AND METHOD OF SECURE GARBAGE COLLECTION ON A MOBILE DEVICE



(57) Abstract: A method and system for performing garbage collection involving sensitive information on a mobile device. Secure information is received at a mobile device over a wireless network. The sensitive information is extracted from the secure information. A software program operating on the mobile device uses an object to access the sensitive information. Secure garbage collection is performed upon the object after the object becomes unreachable.

A3 WO 2003/079196

WO 2003/079196 A3

Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PT, RO, SE, SI, SK, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

— *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii)) for the following designations AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, OM, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, TJ, TM, TN, TR, TT, TZ, UA, UG, UZ, VC, VN, YU, ZA, ZM, ZW, ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PT, RO, SE, SI, SK, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG)*

— *as to the applicant's entitlement to claim the priority of the earlier application (Rule 4.17(iii)) for the following designations AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, OM, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, TJ, TM, TN, TR, TT, TZ, UA, UG, UZ, VC, VN, YU, ZA, ZM, ZW, ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PT, RO, SE, SI, SK, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG)*

— *of inventorship (Rule 4.17(iv)) for US only*

Published:

— *with international search report*

— *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments*

(88) Date of publication of the international search report:
18 March 2004

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

System and Method of Secure Garbage Collection on A Mobile Device

RELATED CASE

This application claims the benefit of and priority to United States Provisional Patent Application Serial No. 60/365,515, filed March 20, 2002, the entire disclosure of which is incorporated herein by reference.

BACKGROUND

1. TECHNICAL FIELD

This invention relates generally to mobile devices and more particularly to secure memory techniques on a mobile device.

2. DESCRIPTION OF THE RELATED ART

Many known mobile devices support objects, such as by using Java to send, receive, or at least use data, voice, and/or multi-media (audio/video). These objects may be involved in sensitive information from cellular networks and with many different services. However, garbage collection operations presently performed on mobile devices have security deficiencies.

A non-limiting example of the deficiencies includes collection of unreachable objects. For example, Fig. 1 shows a typical state of a heap between garbage collections of unreferenced objects. A typical garbage collector waits until memory becomes low before collecting unreachable objects. Thus, an object may become unreachable well before it is collected.

This creates an unpredictable window of opportunity for an attack, especially if the memory recovery itself is not secure.

SUMMARY

In accordance with the teachings disclosed herein, a secure garbage collection system is provided which includes a microprocessor, and an addressable storage, having a heap and a secure garbage collection software module capable of calling a wipe function. When the secure garbage collection software module has detected that objects in the heap are unreachable, it securely reclaims the memory they were using by calling the wipe function.

In another embodiment, the secure garbage collection may be triggered in many different ways, including but not limited to the steps of: waiting for a trigger, performing subsequent steps for all secure applications, requesting that a secure application unreference sensitive objects, perform secure garbage collecting, and determine if all secure applications have been processed.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 shows the state of a typical heap between unreferenced objects using a known garbage collector;

Fig. 2 is a block diagram showing an exemplary secure garbage collection system according to an embodiment of the invention;

Fig. 3a is a block diagram illustrating in greater detail the physical view of an example addressable storage of Fig. 2, featuring objects in RAM and Flash in an exemplary cryptographic message viewing application.

Fig. 3b is a block diagram illustrating the logical view of Fig. 3a;

Fig. 4 is a flow diagram showing an example method of triggering secure garbage collection on a mobile device according to an embodiment of the invention;

Fig. 5 is a flow diagram showing an example method of secure garbage collection whereby unreferenced objects are securely garbage collected according to an embodiment of the invention;

Fig. 6 is a block diagram illustrating software components for use in secure garbage collection on a mobile device; and

Fig. 7 is a schematic diagram of an exemplary wireless device's components.

DETAILED DESCRIPTION

With reference now to the Figures, Fig. 2 is a block diagram showing an exemplary secure garbage collection system 300. The system 300 among other things secures sensitive information, which may exist on its own, may arise from Personal Information Management (PIM), or may arise from communications such as voice and/or video calls, short messaging service (SMS) communication, e-mail messaging, web page communication, and wireless access protocol (WAP) communication. The system 300 enables

secure decryption techniques and secure persistent storage techniques. Many different types of mobile devices may utilize system 300, such as personal digital assistants, mobile communication devices, cellular phones, and wireless two-way communication devices, as well as in any device that has sensitive information.

The exemplary secure garbage collection system 300 of Fig. 2 includes a microprocessor 110, and an addressable storage 120 connected to microprocessor 110 by a data bus 130. The addressable storage 120 stores microprocessor software modules 140, heap 150 and reference table 160.

Microprocessor software 140 includes a native wipe function 170. The native wipe function 170 can obliterate the data in a portion of addressable storage 120. Suitable functions in the 'C' programming language is the function 'memset()', which could be used to write over data with all zeroes, all ones, or with random data to thwart sophisticated memory recovery techniques. Microprocessor software 140 also may include virtual machine software 200, having a secure garbage collector software module 205 capable of using native wipe function 170, as well as being able to access objects in heap 150 via reference table 160. Such software 140 may be used in many different implementation environments, such as object-oriented environments (e.g., Java).

Virtual machine software 200 is capable of interpreting virtual machine instructions found in software modules 210. A specific virtual machine software module (e.g., secure viewer application 220) is shown, and

will be used as an example application which uses secure garbage collection techniques.

Secure viewer application 220, when executed by virtual machine software 200, results in viewer object 10V being allocated in heap 150, accessible via its corresponding @V 35V entry in reference table 160. Viewer object 10V could be, for instance, a user interface object which displays sensitive information in a sensitive object, such as object 10S. Viewer object 10V preferably dynamically generates sensitive object 10S from secure object 70E by authentication in viewer application 220.

For instance, if secure object 70E is an S/MIME encrypted message, then sensitive object 10S is a clear unencrypted version of the S/MIME message, dynamically generated by secure viewer application module 220, in this case an S/MIME e-mail viewer application, preferably by obtaining and applying a private key to encrypted S/MIME message object 70E.

Heap 150 may be partitioned so that sensitive objects, such as object 10S are distinguishable from secure objects, such as object 70E, as illustrated by regions 152 and 157 which respectively bound sensitive and secure portions of heap 150. It should be understood that many different partitioning configurations (or none at all) are possible with respect to handling sensitive and secure objects in order to fit the situation at hand.

Also shown is a portion of heap 150 which is unreachable, illustrated by region 155. Region 155 contains objects 10V', 10S' and 70E' which are no longer referenced by other objects, and as such, are suitable for garbage

collection. Notice that object 10S' is both unreachable and sensitive, object 70E' is both unreachable and secure, whereas object 10V' is only unreachable.

Returning to the S/MIME viewer application 220 example, objects 10V', 10S' and 70E' are unreachable, for instance if the S/MIME viewer application stopped displaying viewer 10V' in response to a delete message user interface command. Thus, if viewer 10V' was the only object having references to sensitive object 10S' and 70E', when the reference to 10V' is lost, all three objects are candidates for garbage collection. Notice however that object 70E, although referenced by viewer 10V', is still reachable and secure (e.g. encrypted S/MIME – perhaps because it was the previous message viewed by viewer 10V' and was not deleted by the user).

Secure garbage collector module 205, once it has detected that objects 10V', 10S' and 70E' are unreachable, securely reclaims the memory they were using by calling native wipe function 170 to wipe, at least object 10S', as well as optionally objects 10V' and 70E'. Optionally, all garbage collections use the wipe native function 170 thereby treating all objects as sensitive.

With reference to Figs. 3a and 3b, Fig. 3a is a block diagram illustrating in greater detail the physical view of an example addressable storage of Fig. 2, featuring a reference table, a viewer object in RAM, a persistent encrypted object in Flash, and a transient sensitive object in RAM, in an exemplary cryptographic message viewing application.

Fig. 3b is a block diagram illustrating the logical view of Fig. 3a. Both are described next.

An object 10V that references object 10S and object 70E, are illustrated as they might appear somewhere in RAM 20 or Flash 80.

Also illustrated is a reference table 30, situated somewhere in RAM 20. The reference table 30 has several storage elements (35V, 35S, 35E) of a fixed size "w" 37 to simplify the indexed access to storage elements. Each used storage element (35V, 35S, 35E) corresponds to an object (10V, 10S, 70E) which are located in an addressable space, here consisting of RAM 20 and Flash 80. For example object V 10V finds correspondence with storage element index "v" 35V, object E 70E finds correspondence with storage element index "e" 35E, whereas object S 10S finds correspondence with storage element index "s" 35S. The addresses (40V, 40S, 90E) of corresponding objects (10V, 10S, 70E) are stored in storage elements (35V, 35S, 35E) so that knowing the index of an object in the reference table 30 it is possible to obtain the address (40V, 40S, 90E) of an object (10V, 10S, 70E) respectively. This is done by first obtaining the address @R 50 of the reference table 30. Then, given an object's reference, such as "s" 55S for the example V object 10V, the address of the storage element @(R+v*w) 60V can be obtained by multiplying the index 55V "v" by the size "w" 37 of each storage element.

Since the "v" storage element 35V holds the address of the corresponding object V 10V, resolving the contents of the storage element 35V provides the address @V 40V of object V 10V in RAM 20. Similarly, the "s" storage element 35S, when resolved provides the address @S 40S of

object 10S in RAM 20, and the "e" storage element 35E points to an address @E 90E of object 70E in Flash 80. Also shown is how each object (10V, 10S, 70E) contains within its format its "this reference" (55V, 55S, 55E) related to the reference table 30. Also shown is how, object V 10V contains within its format a reference "E" 65E to object E 70E, and a reference "S" 65S to object S 10S. This allows a runtime context within the scope of object V 10V to be able to access objects E 70E and 10S in the same way, regardless of the fact that object E 70E is situated at an address 90E in Flash 80 and objects V 70A is in RAM 20.

Object 10V could be a Secure Multipurpose Internet Mail Extensions (S/MIME) viewer, in which case object 70E could be a persisted (S/MIME) encrypted message, and object 10S could be the sensitive decrypted version of encrypted message 70E. Viewer object 10V could have generated sensitive object 10S from encrypted object 70E at the request of and after authenticating the user of viewer 10V -- that is, the intended recipient of the S/MIME message.

With reference to Fig. 4, Fig. 4 depicts a flow diagram showing an example method of triggering secure garbage collection on a mobile device. Step 410 includes waiting for a trigger. Any parameters associated with a trigger could be loaded from storage 120 via configuration 402. A trigger can result from many different events, such as but not limited:

- 405I is a timeout event, which may occur when the mobile device is left idle;

- 405H is a holstered or cradled event, which may occur when the user, or an attacker, either places or removes the device from its holster (if so equipped) or cradle (if so equipped).
- 405L is a screen lock or user lock event, which may occur due to any number of reasons, such as when a user enters a password at a lock screen, or when a user expressly locks the device or screen;
- 405A is an application event, such as when a viewer has stopped displaying a sensitive object. In the case of S/MIME, messages are preferably kept secure (encrypted) and are decrypted only if viewed. However, a configuration parameter could be used to age the decrypted message before causing a secure garbage collection trigger to give the user the opportunity to view a message, close it, and re-open it within a narrow time out period.
- 405R is a roll back trigger, which can occur whenever the system clock (if so equipped) or a time zone has been altered. A configuration parameter could be used to specify the specific cases.
- 405E is a transceiver event, which can occur if the mobile device communicates (if so equipped), for instance over a wireless network. For example, when communicating using S/MIME, or while browsing using SSL or TLS, caches may be securely garbage collected.

Step 420 includes performing subsequent steps for all secure applications. Secure applications may be selected by configuration, or may include all applications.

Step 430 includes requesting that a secure application unreference sensitive objects. Thus, this step helps ensure that the window of opportunity of an attacker is greatly limited in secure applications regardless of the trigger.

Step 440 includes securely garbage collecting. This step at least includes calling the native wipe function call, but may also include other actions, such as, but not limited to, cleaning out the system clipboard (if so equipped and configured). An exemplary method to carry out this step is discussed below with reference to Fig. 5.

Step 450 includes determining if all secure applications have been processed. If all secure applications are clean (e.g., applications have no references to sensitive objects), then steps 430 and 440 are repeated for the remaining secure applications. Alternatively, if all secure applications are clean, then step 410 ensues and the method begins anew.

It is noted that the method of Fig. 4 may be implemented as a "Daemon" application for the virtual machine.

With reference to Fig. 5, Fig. 5 is a flow diagram showing an exemplary method of secure garbage collection whereby unreferenced objects are securely garbage collected.

The method 500 of Fig. 5 may be used to carry out step 440 of Fig. 4. Step 510 includes collecting unreferenced objects. This step may receive an indication, for instance via configuration information 502, such as which trigger caused the garbage collection in the method 400 of Fig. 4. For example, if an S/MIME viewer application was the trigger, then unreferenced

sensitive objects would preferably be collected from the heap starting near the cause of the trigger.

Step 520 includes performing subsequent steps for all unreferenced objects in the heap. Step 530 includes determining if the unreferenced object is sensitive. As was described in reference to Fig. 2, in a preferred embodiment, all unreferenced objects are treated as sensitive. This may also be specified in the configuration information 502. If the unreferenced object is determined to be sensitive, then step 540 ensues, followed by step 550; if not, then step 550 ensues.

Step 540 includes calling the native wipe function to obliterate the sensitive information in the unreferenced sensitive object. As was described in reference to Fig. 2, the native wipe function could be a C "memset()" of object data to zeroes, ones, or random data. Which option to use could be specified in configuration 502. It is also envisaged that a non-native wipe function could be used.

Step 550 includes reclaiming object memory. This step could be accomplished by a traditional garbage collector. By replacing all calls to the traditional garbage collector with calls to a secure garbage collector, secure garbage collection can be enabled in many existing methods and systems.

Step 560 includes determining if all unreferenced objects have been reclaimed. If this is determined, then the method ends. If not, then step 530 ensues to continue secure garbage collection.

Fig. 6 illustrates at 600 a mobile device having a secure garbage collection system. The mobile device 600 receives information (e.g., secure message 602) over a wireless network 604. A software program 606 operating on the mobile device 600 processes the secure message 602 such that a secure object is created and is stored within addressable storage memory 608 to handle the secure message 602. In this example, sensitive information is extracted from the secure message 606, and a sensitive object is created and stored within addressable storage memory 608 in order to handle the sensitive information.

When objects (610, 612, 614) in the addressable storage memory 608 are detected as unreachable, a secure garbage collector module 616 securely reclaims the memory 608 the objects (610, 612, 614) were using by calling a wipe function 618. Optionally, all garbage collections use the wipe function 618 thereby treating all objects (610, 612, 614) as sensitive. However, it should be understood that the garbage collection module 616 may vary the type of objects the wipe function 618 may be used for. For example, the garbage collection module 616 may be configured to only use the wipe function 618 upon unreachable sensitive objects 610, or only upon unreachable secure objects 612, or combinations thereof. Moreover, the garbage collection module 616 may be configured to use the wipe function 618 upon unreachable objects of one or more software programs. Such approaches initiate secure garbage collection in order to prevent unauthorized access to sensitive information. Thus, secure garbage collection is initiated

when an object (such as a sensitive object) becomes unreachable rather than only when memory becomes scarce.

Many different types of mobile devices may utilize the systems and methods disclosed herein, such as a wireless device shown in Fig. 7. With reference to Fig. 7, wireless device 900 is preferably a two-way communication device having at least voice and data communication capabilities. The device 900 preferably has the capability to communicate with other computer systems on the Internet. Depending on the functionality provided by the device, the device 900 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).

Where the device 900 is enabled for two-way communications, the device 900 may incorporate a communication subsystem 911, including a receiver 912, a transmitter 914, and associated components such as one or more, preferably embedded or internal, antenna elements 916 and 918, local oscillators (LOs) 913, and a processing module such as a digital signal processor (DSP) 920. As will be apparent to those skilled in the field of communications, the particular design of the communication subsystem 911 will be dependent upon the communication network in which the device is intended to operate. For example, a device 900 destined for a North American market may include a communication subsystem 911 designed to operate within the Mobitex mobile communication system or DataTAC mobile

communication system, whereas a device 900 intended for use in Europe may incorporate a General Packet Radio Service (GPRS) communication subsystem 911.

In general, the device 900 may acquire or generate secure and sensitive information through its interaction with cellular networks and the services the networks provide. Examples of cellular networks and services they provide include Code Division Multiple Access (CDMA) and Global Service Mobile (GSM) networks which provide for the most part voice and some data services. Voice services are typically compatible with plain old telephony service (POTS). Short Messaging Service (SMS) and Wireless Application Protocol (WAP) are available on some cellular networks. Data networks, such as MobiTex™, Datatac™, as well as advanced networks such as General Packet Radio Service (GPRS), and Universal Mobile Telecommunications System (UMTS), may allow an appropriately configured wireless mobile device to offer data services such as e-mail, web browsing, SMS, WAP, as well as PIM. Future networks may also offer video services. Thus, sources of sensitive information abound.

Network access requirements will also vary depending upon the type of network 919. For example, in the Mobitex and DataTAC networks, mobile devices such as 900 are registered on the network using a unique personal identification number or PIN associated with each device. In GPRS networks however, network access is associated with a subscriber or user of a device 900. A GPRS device therefore requires a subscriber identity module (not

shown), commonly referred to as a SIM card, in order to operate on a GPRS network. Without a SIM card, a GPRS device will not be fully functional. Local or non-network communication functions (if any) may be operable, but the device 900 will be unable to carry out any functions involving communications over network 919. When required network registration or activation procedures have been completed, a device 900 may send and receive communication signals over the network 919. Signals received by the antenna 916 through a communication network 919 are input to the receiver 912, which may perform such common receiver functions as signal amplification, frequency down conversion, filtering, channel selection and the like, and in the example system shown in Fig. 7, analog to digital conversion. Analog to digital conversion of a received signal allows more complex communication functions such as demodulation and decoding to be performed in the DSP 920. In a similar manner, signals to be transmitted are processed, including modulation and encoding for example, by the DSP 920 and input to the transmitter 914 for digital to analog conversion, frequency up conversion, filtering, amplification and transmission over the communication network 919 via the antenna 918.

The DSP 920 not only processes communication signals, but also provides for receiver and transmitter control. For example, the gains applied to communication signals in the receiver 912 and transmitter 914 may be adaptively controlled through automatic gain control algorithms implemented in the DSP 920.

The device 900 preferably includes a microprocessor 938, which controls the overall operation of the device. Communication functions, including at least data and voice communications, are performed through the communication subsystem 911. The microprocessor 938 also interacts with further device subsystems such as the display 922, flash memory 924, random access memory (RAM) 926, auxiliary input/output (I/O) subsystems 928, serial port 930, keyboard 932, speaker 934, microphone 936, a short-range communications subsystem 940 and any other device subsystems generally designated as 942.

Some of the subsystems shown in Fig. 7 perform communication-related functions, whereas other subsystems may provide "resident" or on-device functions. Notably, some subsystems, such as keyboard 932 and display 922 for example, may be used for both communication-related functions, such as entering a text message for transmission over a communication network, and device-resident functions such as a calculator or task list.

Operating system software used by the microprocessor 938, which could be element 110 of Fig. 2 is preferably stored in a persistent store such as flash memory 924, which could be element 80 of Fig. 3a and may instead be a read only memory (ROM) or similar storage element or could be a portion of addressable storage 120 of Figs 2, 3a and 3b. Those skilled in the art will appreciate that the operating system, specific device applications, or parts thereof, may be temporarily loaded into a volatile store such as RAM 926,

which could be element 20 of Fig. 2. It is contemplated that received communication signals may also be stored to RAM 926. Flash memory 924 preferably includes data communication module 924B, and when device 900 is enabled for voice communication, voice communication module 924A. For the purposes of this invention, are also included in flash memory 924 other software modules 924N, which could be microprocessor software 140 of Fig. 2.

The microprocessor 938, in addition to its operating system functions, preferably enables execution of software applications on the device. A predetermined set of applications which control basic device operations, including at least data and voice communication applications for example, will normally be installed on the device 900 during manufacture. A preferred application that may be loaded onto the device may be a personal information manager (PIM) application having the ability to organize and manage data items relating to the device user such as, but not limited to e-mail, calendar events, voice mails, appointments, and task items. Naturally, one or more memory stores may be available on the device to facilitate storage of PIM data items on the device. Such PIM application would preferably have the ability to send and receive data items, via the wireless network. In a preferred embodiment, the PIM data items are seamlessly integrated, synchronized and updated, via the wireless network, with the device user's corresponding data items stored or associated with a host computer system. Further applications may also be loaded onto the device 900 through the network 919, an auxiliary

I/O subsystem 928, serial port 930, short-range communications subsystem 940 or any other suitable subsystem 942, and installed by a user in the RAM 926 or preferably a non-volatile store (not shown) for execution by the microprocessor 938. Such flexibility in application installation increases the functionality of the device 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 device 900.

In a data communication mode, a received signal such as a text message or web page download will be processed by the communication subsystem 911 and input to the microprocessor 938, which will preferably further process the received signal for output to the display 922, or alternatively to an auxiliary I/O device 928. A user of device 900 may also compose data items such as e-mail messages for example, using the keyboard 932, which is preferably a complete alphanumeric keyboard or telephone-type keypad, in conjunction with the display 922 and possibly an auxiliary I/O device 928. Such composed items may then be transmitted over a communication network through the communication subsystem 911.

For voice communications, overall operation of the device 900 is substantially similar, except that received signals would preferably be output to a speaker 934 and signals for transmission would be generated by a microphone 936. Alternative voice or audio I/O subsystems such as a voice message recording subsystem may also be implemented on the device 900.

Although voice or audio signal output is preferably accomplished primarily through the speaker 934, the display 922 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 serial port 930, would normally be implemented in a personal digital assistant (PDA)-type communication device for which synchronization with a user's desktop computer (not shown) may be desirable, but is an optional device component. Such a port 930 would enable a user to set preferences through an external device or software application and would extend the capabilities of the device by providing for information or software downloads to the device 900 other than through a wireless communication network. The alternate download path may for example be used to load an encryption key onto the device through a direct and thus reliable and trusted connection to thereby enable secure device communication.

A short-range communications subsystem 940 is a further optional component which may provide for communication between the device 900 and different systems or devices, which need not necessarily be similar devices. For example, the subsystem 940 may include an infrared device and associated circuits and components or a Bluetooth™ communication module to provide for communication with similarly-enabled systems and devices.

Having described in detail the preferred embodiments of the present invention, including the preferred methods of operation, it is to be understood that this operation could be carried out with different elements and steps. This

preferred embodiment is presented only by way of example and is not meant to limit the scope of the present invention. This written description may enable those skilled in the art to make and use embodiments having alternative elements that likewise correspond to the elements of the invention. The intended scope of the invention thus includes other structures, systems or methods that do not differ from the literal language of the description, and further includes other structures, systems or methods with insubstantial differences from the literal language of the description.

It is claimed:

1. A method for performing secure garbage collection for sensitive information on a mobile device, comprising the steps of:

receiving at a mobile device secure information over a wireless network;

extracting sensitive information from the secure information;

storing the sensitive information in memory of the mobile device;

wherein a software program operating on the mobile device uses a first object to access the stored sensitive information;

waiting for a trigger, wherein the trigger is to be used as an indication to perform a secure garbage collection operation

determining that a trigger has occurred;

if a trigger has occurred, requesting that a secure application unreference objects, wherein the first object becomes unreachable due to the secure application having unreferenced the first object;

determining that the first object has become unreachable;

based upon the determination that the first object has become unreachable, obliterating the unreachable first object from the memory; and

reclaiming the memory that the first object was using.

2. A method of triggering secure garbage collection on a mobile device, comprising the steps of:

waiting for a trigger, wherein the trigger is to be used as an indication to perform a secure garbage collection operation

determining that a trigger has occurred;

if a trigger has occurred, requesting that an application operating on the mobile device unreference sensitive objects; and

performing secure garbage collection upon the unreferenced sensitive objects, wherein the secure garbage collection renders sensitive data associated with an unreferenced sensitive object unreadable.

3. A garbage collection system for operation on a mobile device, wherein the mobile device includes memory for storing at least one object used by a software program to access sensitive information stored on the mobile device, comprising:

a configuration data structure to store information about at least one triggering event, wherein the triggering event is used as an indication to perform a secure garbage collection operation;

a garbage collection module having a data access pathway to the configuration data structure and to the memory;

wherein the garbage collection module is configurable to perform a secure garbage collection operation based upon a detection of the

triggering event, wherein the detection of the triggering event is performed based upon the information stored in the configuration data structure;

wherein the secure garbage operation includes a determination that the object is unreachable and a call to a wipe function with respect to the unreachable object.

4. A secure garbage collection system for handling sensitive information on a mobile device, comprising:

means for receiving at a mobile device secure information over a wireless network;

means for extracting sensitive information from the secure information;

means for storing the sensitive information in memory of the mobile device;

wherein a software program operating on the mobile device uses an object to access the stored sensitive information;

means for determining that a triggering event has occurred;

means for determining that the object has become unreachable;

means for obliterating the unreachable object from the memory after the object has been determined as unreachable;

wherein the obliterated object is rendered unreadable; and

means for reclaiming the memory that the unreachable object was using.

5. A mobile device configurable to perform secure garbage collection, said mobile device comprising:

a microprocessor configurable to execute a software program which handles sensitive information;

heap memory for storing at least one object used by the software program to access the sensitive information;

a garbage collection module operable on the microprocessor and having a data access pathway to the heap memory;

wherein the garbage collection module is configurable to call a wipe function with respect to the object stored in heap memory upon a detection that the object in the heap memory is unreachable.

6. A method for performing secure garbage collection involving sensitive information on a mobile device, comprising the steps of:

receiving at a mobile device secure information over a wireless network;

extracting sensitive information from the secure information;

storing the sensitive information in memory of the mobile device;

wherein a software program operating on the mobile device uses an object to access the stored sensitive information;

determining that the object has become unreachable;

based upon the determination that the object has become unreachable, obliterating the unreachable object from the memory; and

reclaiming the memory that the object was using.

7. The method of claim 6, wherein the mobile device is a personal digital assistant.

8. The method of claim 6, wherein the mobile device is a mobile communication device.

9. The method of claim 6, wherein the mobile device is a cellular telephone with data messaging capabilities.

10. The method of claim 6, wherein the mobile device is a wireless two-way communication device.

11. The method of claim 6, wherein the mobile device is a data messaging device.

12. The method of claim 6, wherein the mobile device is a two-way pager.

13. The method of claim 6, wherein the mobile device is a wireless Internet appliance.

14. The method of claim 6, wherein the wireless network includes means for providing the wireless network.

15. The method of claim 6, wherein the secure information is an S/MIME encrypted message.

16. The method of claim 15, wherein the sensitive information is an unencrypted version of the S/MIME message.

17. The method of claim 16, wherein the secure message was unencrypted by obtaining and applying a private key to the encrypted S/MIME message.

18. The method of claim 6, wherein the sensitive information arises from use of a Personal Information Management (PIM) application.

19. The method of claim 6, wherein the sensitive information arises from use of the mobile device for voice communications.

20. The method of claim 6, wherein the sensitive information arises from use of the mobile device for video communications.

21. The method of claim 6, wherein the sensitive information arises from use of the mobile device for short messaging service (SMS) communications.

22. The method of claim 6, wherein the sensitive information arises from use of the mobile device for e-mail messaging communications.

23. The method of claim 6, wherein the sensitive information arises from use of the mobile device for web page communications.

24. The method of claim 6, wherein the sensitive information arises from use of the mobile device for wireless access protocol (WAP) communications.

25. The method of claim 6, wherein an object is unreachable when it is not referenced by another object.

26. The method of claim 6, wherein an object is unreachable when it is unreferenced by the software program.

27. The method of claim 6, wherein an object is unreachable when it is not referenced by any object.

28. The method of claim 6, wherein when the object in the memory is detected as being unreachable, a wipe function is used in order to securely reclaim the memory which the object was using.

29. The method of claim 28, wherein a system clipboard is cleaned in addition to the obliteration of the object.

30. The method of claim 28, wherein at least substantially all garbage collections performed on the mobile device use the wipe function.

31. The method of claim 28, wherein all garbage collections performed on the mobile device use the wipe function thereby treating all objects as sensitive.

32. The method of claim 6, wherein the memory stores software programs for operation upon the mobile device's microprocessor, wherein a non-native wipe function is used to obliterate the object from the memory.

33. The method of claim 6, wherein the memory stores software programs for operation upon the mobile device's microprocessor, wherein a native wipe function is used to obliterate the object from the memory.

34. The method of claim 33, wherein the object is obliterated from the memory by writing over the object's data in the memory with all zeroes.

35. The method of claim 33, wherein the object is obliterated from the memory by writing over the object's data in the memory with all ones.

36. The method of claim 33, wherein the object is obliterated from the memory by writing over the object's data in the memory with random data.

37. The method of claim 33, wherein the object is obliterated from the memory in order to thwart a sophisticated memory recovery technique by an attacker.

38. The method of claim 6, wherein the memory stores software programs for operation upon the mobile device's microprocessor, wherein a wipe function is used to obliterate the object from the memory.

39. The method of claim 38, wherein the software programs include virtual machine software having a secure garbage collector software module capable of using the wipe function.

40. The method of claim 39, wherein the virtual machine software is configured to interpret virtual machine instructions found in the software programs.

41. The method of claim 39, wherein the memory includes dynamically allocable memory for use in storing the object.

42. The method of claim 39, wherein the memory includes heap memory, wherein the virtual machine software is configured to access objects stored in heap via a reference table.

43. The method of claim 42, wherein the object handling the sensitive object is a sensitive object, wherein an object handling the secure information is a secure object, wherein the heap contains partitions so that sensitive objects are distinguishable from secure objects.

44. The method of claim 43, wherein the secure objects are stored in non-volatile memory of the mobile device, wherein the sensitive objects are stored in volatile memory of the mobile device.

45. The method of claim 44, wherein memory addresses of objects are stored in the reference table so that given an index of an object in the reference table, the memory address of the object may be obtained.

46. The method of claim 45, wherein an object contains within its format the object's references to any other objects.

47. The method of claim 46, wherein the reference table allows the object to access other objects irrespective of whether they are stored in volatile or non-volatile memory.

48. The method of claim 47, wherein the software program that creates the object is a secure viewer application operating on the mobile device.

49. The method of claim 48, wherein the secure viewer application, when executed by the virtual machine software, results in the object being allocated in heap and accessible via its corresponding entry in the reference table.

50. The method of claim 49, wherein a viewer object is used to display to a user of the mobile device the sensitive information.

51. The method of claim 50, wherein the viewer object dynamically generates a sensitive object by authentication in the secure viewer application.

52. The method of claim 6, wherein decision to obliterate the object and to reclaim the memory that the object was using is based upon a security consideration.

53. The method of claim 6, wherein decision to obliterate the object and to reclaim the memory that the object was using is based upon a security consideration and not based upon amount of memory remaining.

54. The method of claim 6, wherein decision to obliterate the object and to reclaim the memory that the object was using is based upon said step which determines that the object has become unreachable, wherein the decision is independent of the level of available memory.

55. The method of claim 6, wherein decision to obliterate the object and to reclaim the memory that the object was using is based upon said step which determines that the object has become unreachable, wherein the decision is not performed based upon the amount of memory being lower than a preselected threshold.

56. The method of claim 6, further comprising the steps of:

determining that a trigger has occurred, wherein the trigger is used as an indication to perform a secure garbage collection operation;

based upon receiving the trigger, requesting that a secure application unreference the application's sensitive objects;

determining that a sensitive object has been unreferenced by the secure application;

based upon the determination that the sensitive object has been unreferenced, obliterating the unreferenced object from memory.

57. The method of claim 56, wherein a plurality of objects are obliterated based upon a determination that a plurality of sensitive objects have been unreferenced by the secure applications, wherein the type of trigger is used to determine how garbage collection is to be performed.

58. The method of claim 56, wherein a plurality of secure applications are requested to unreference their respective sensitive objects.

59. The method of claim 56, wherein a parameter associated with the trigger is loaded from a configuration file, wherein the parameter is used in the determining that a trigger has occurred.

60. The method of claim 56, wherein the configuration file specifies that all unreferenced objects are to be treated as sensitive objects.

61. The method of claim 56, wherein the trigger includes trigger means.

62. The method of claim 56, wherein the trigger occurs based upon a preselected event occurring.

63. The method of claim 62, wherein the event is timeout event which occurs when the mobile device is idle for a preselected time.

64. The method of claim 62, wherein the event is a cradled event.

65. The method of claim 64, wherein the cradled event occurs when the mobile device is placed in or removed from the mobile device's cradle.

66. The method of claim 62, wherein the event is a holstered event.

67. The method of claim 66, wherein the holstered event occurs when the mobile device is placed in or removed from the mobile device's holster.

68. The method of claim 62, wherein the event is a screen lock event.

69. The method of claim 62, wherein the event is a user lock event.

70. The method of claim 62, wherein the event is an application event.

71. The method of claim 70, wherein the application event includes when a viewer has stopped displaying a sensitive object.

72. The method of claim 71, wherein a configuration parameter is used to age a decrypted message before causing a trigger to be generated.

73. The method of claim 72, wherein the aging of a decrypted message is used to give the mobile device's user an opportunity to view a message, close it, and re-open it within a preselected time out period.

74. The method of claim 62, wherein the event is a roll back trigger event.

75. The method of claim 74, wherein the roll back trigger event occurs when the mobile device's system clock or a time zone has been altered.

76. The method of claim 62, wherein the event is a transceiver-based event.

77. The method of claim 76, wherein the transceiver-based event occurs if the mobile device communicates over the wireless network.

78. The method of claim 56, wherein based upon the determining of the trigger, requesting that a secure application unreference the application's non-sensitive objects.

79. The method of claim 56, wherein the secure garbage collection operation includes obliterating the unreferenced sensitive objects.

80. The method of claim 79, wherein the secure garbage collection operation includes reclaiming memory that was used by the unreferenced sensitive objects.

81. The method of claim 6, further comprising the steps of:

determining that a trigger has occurred, wherein the trigger is used as an indication to perform a secure garbage collection operation;

based upon receiving the trigger, requesting that a secure application unreference the application's sensitive objects; and

performing a secure garbage collection operation upon the unreferenced sensitive objects.

82. The method of claim 6, further comprising the steps of:

waiting for a trigger before performing a secure garbage collection operation;

determining that a trigger has occurred, wherein the trigger is used as an indication to perform a secure garbage collection operation;

based upon receiving the trigger, requesting that a secure application unreference the application's sensitive objects; and

performing a secure garbage collection operation upon the sensitive objects unreferenced by the application.

83. The method of claim 6, further comprising the steps of:

waiting for a trigger before performing a secure garbage collection operation;

determining that a trigger has occurred, wherein the trigger is used as an indication to perform a secure garbage collection operation;

based upon receiving the trigger, requesting that secure applications unreference the applications' sensitive objects; and

performing a secure garbage collection operation upon the sensitive objects unreferenced by the applications.

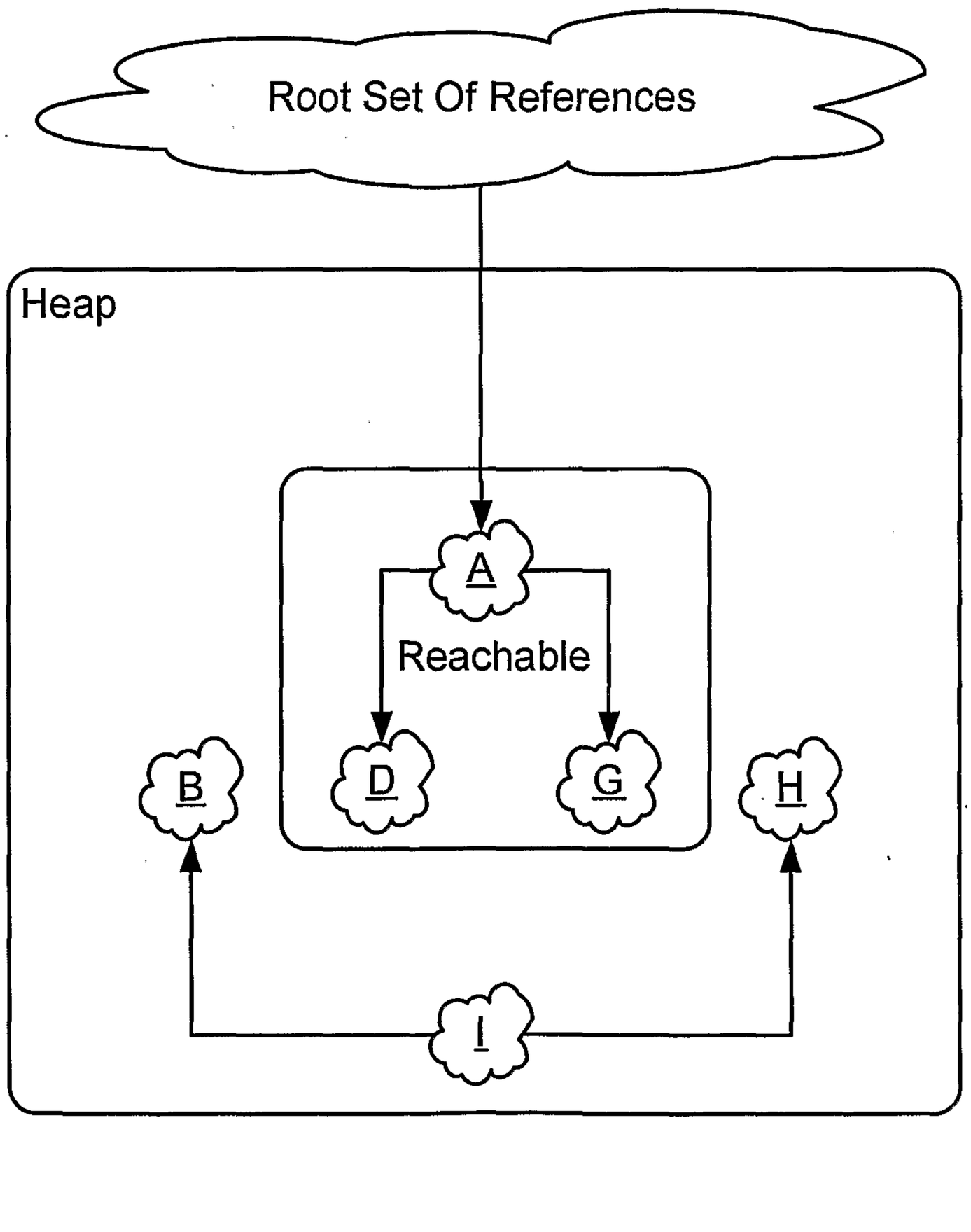


Fig. 1

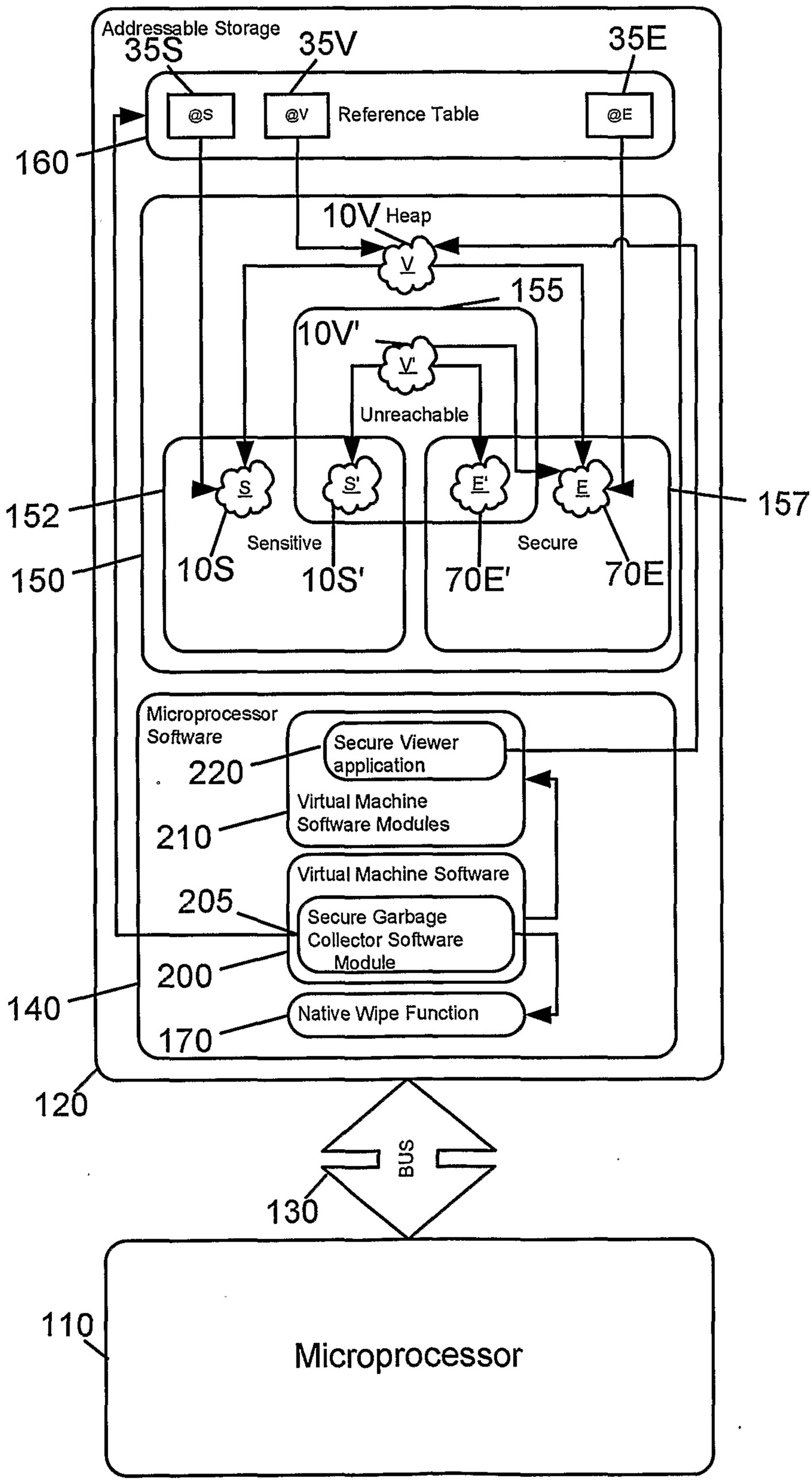
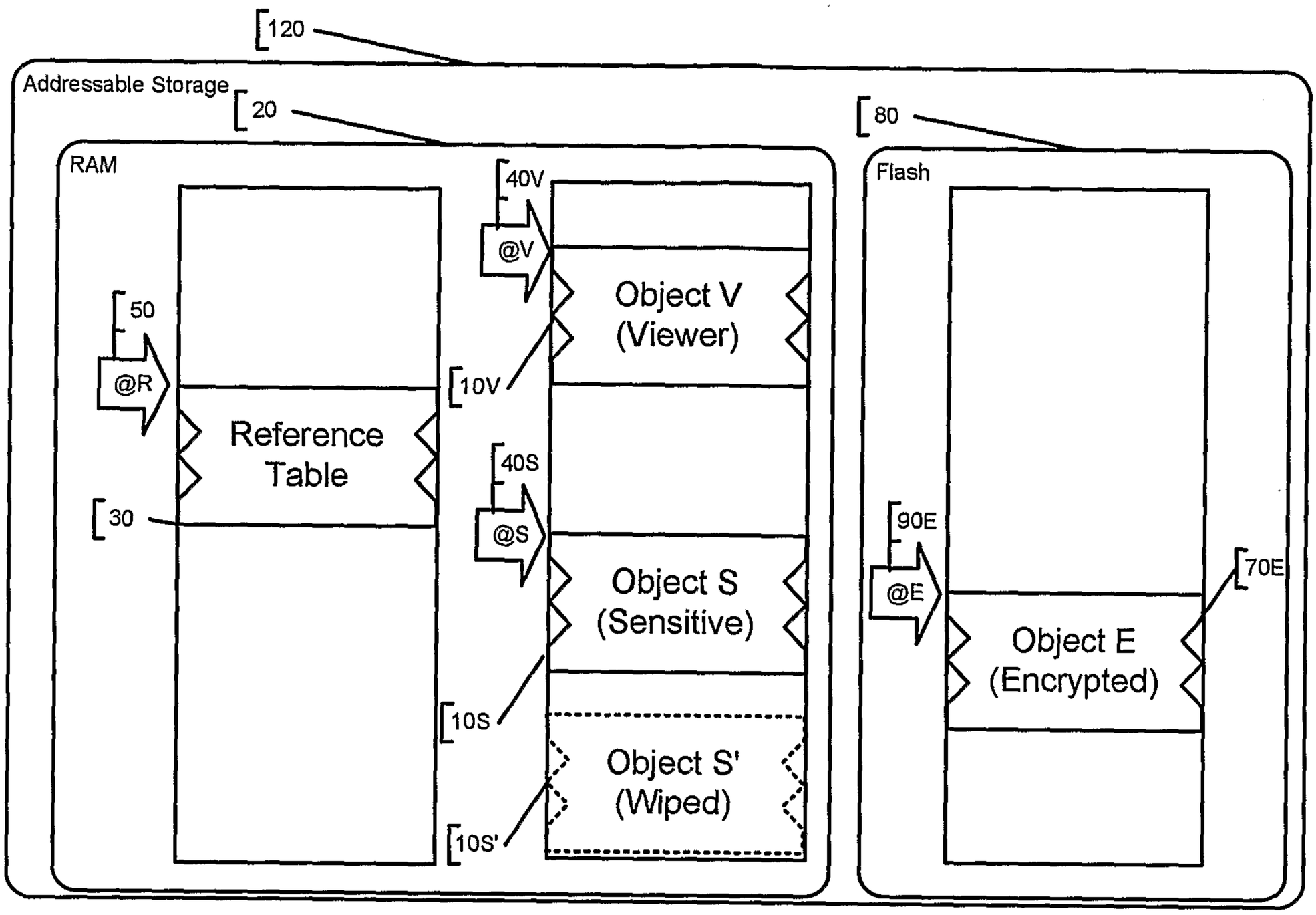
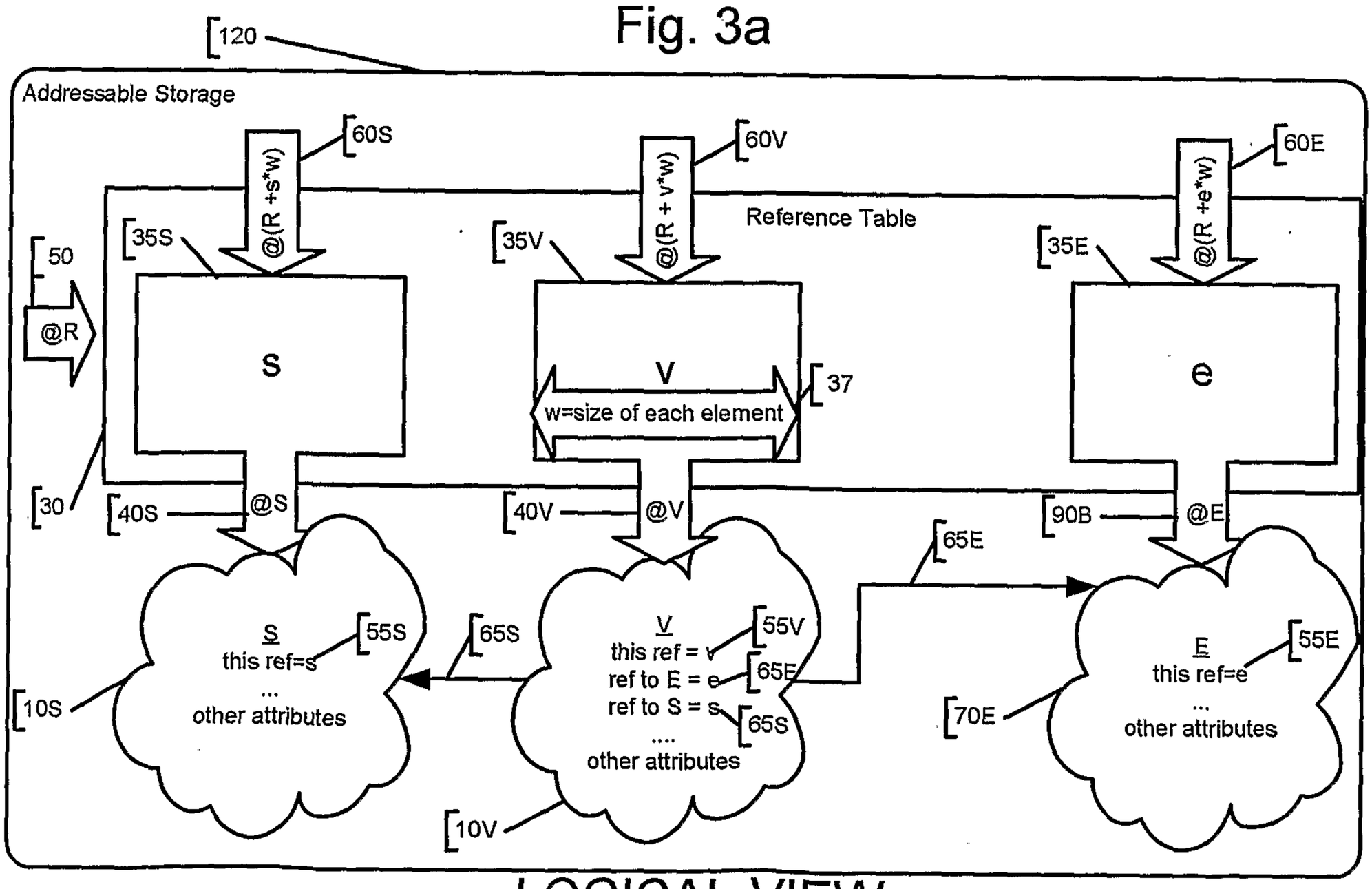

Addressable Storage
35S
35V
35E
@S
@V
Reference Table
@E
160
10V Heap
V
155
10V'
V'
Unreachable
S
S'
E'
E
152
157
Sensitive
Secure
150
10S
10S'
70E'
70E
Microprocessor Software
220
Secure Viewer application
Virtual Machine Software Modules
210
Virtual Machine Software
205
Secure Garbage Collector Software Module
200
140
Native Wipe Function
170
120
BUS
130
110
Microprocessor
300

Fig. 2



PHYSICAL VIEW
Fig. 3a



LOGICAL VIEW
Fig. 3b

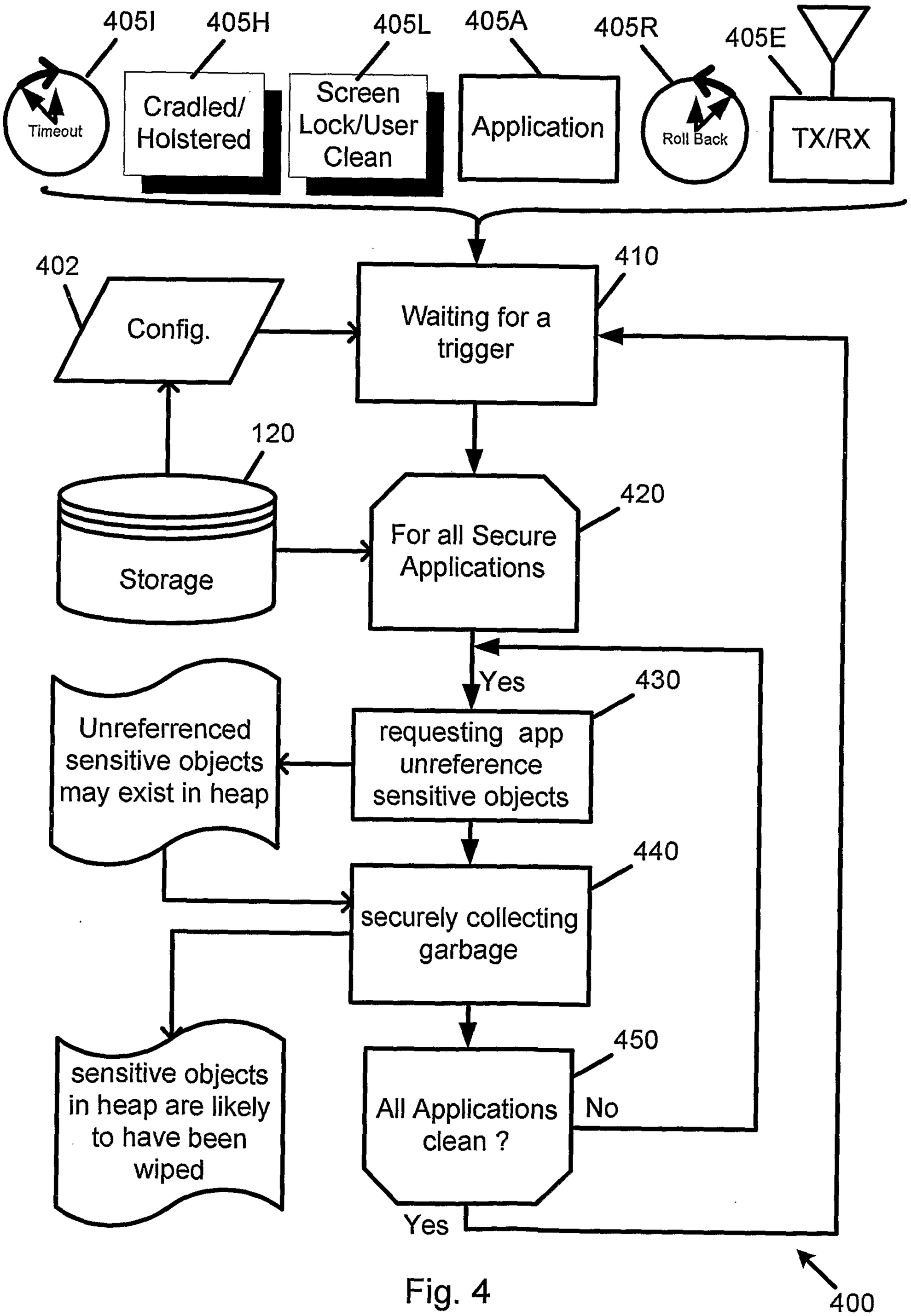

405I
405H
405L
405A
405R
405E
Timeout
Cradled/ Holstered
Screen Lock/User Clean
Application
Roll Back
TX/RX
402
Config.
410
Waiting for a trigger
120
Storage
420
For all Secure Applications
Yes
430
requesting app unreference sensitive objects
Unreferrenced sensitive objects may exist in heap
440
securely collecting garbage
sensitive objects in heap are likely to have been wiped
450
All Applications clean ?
No
Yes
400

Fig. 4

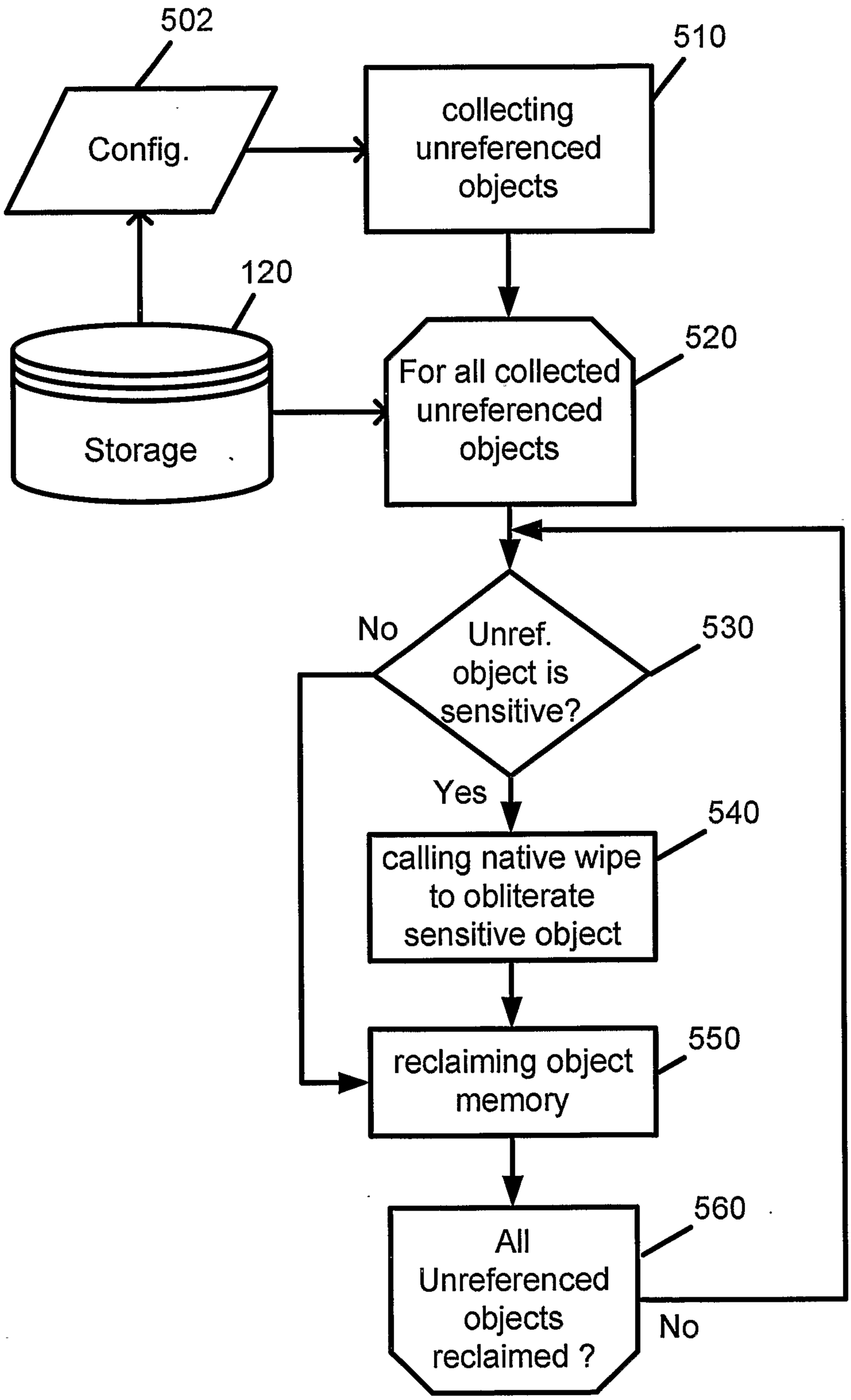

502
Config.
510
collecting unreferenced objects
120
Storage
520
For all collected unreferenced objects
No
Unref. object is sensitive?
530
Yes
540
calling native wipe to obliterate sensitive object
550
reclaiming object memory
560
All Unreferenced objects reclaimed ?
No
500

Fig. 5

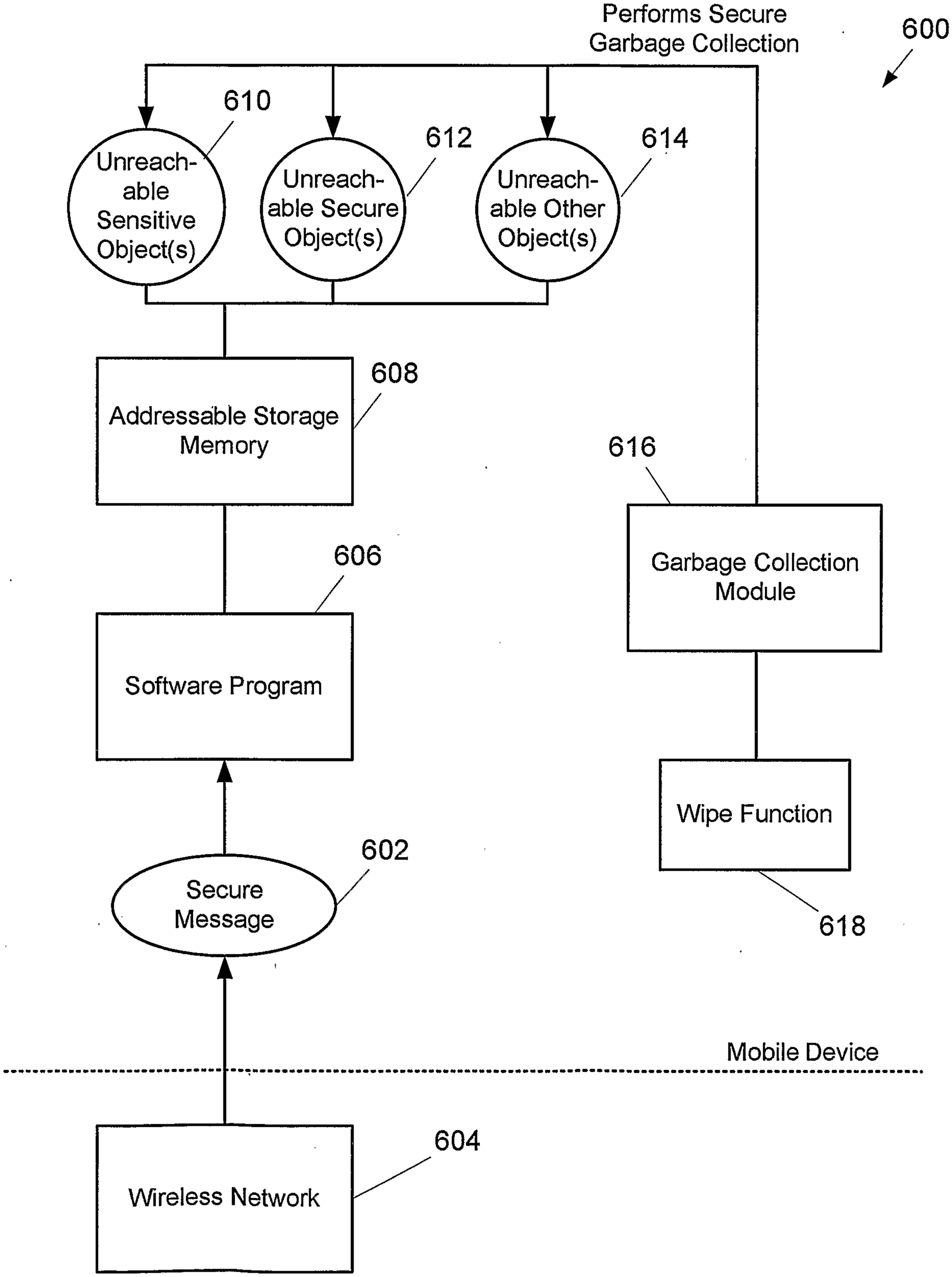

Performs Secure Garbage Collection
600
610
612
614
Unreach-able Sensitive Object(s)
Unreach-able Secure Object(s)
Unreach-able Other Object(s)
608
Addressable Storage Memory
616
606
Software Program
Garbage Collection Module
Wipe Function
602
Secure Message
618
Mobile Device
604
Wireless Network

Fig. 6

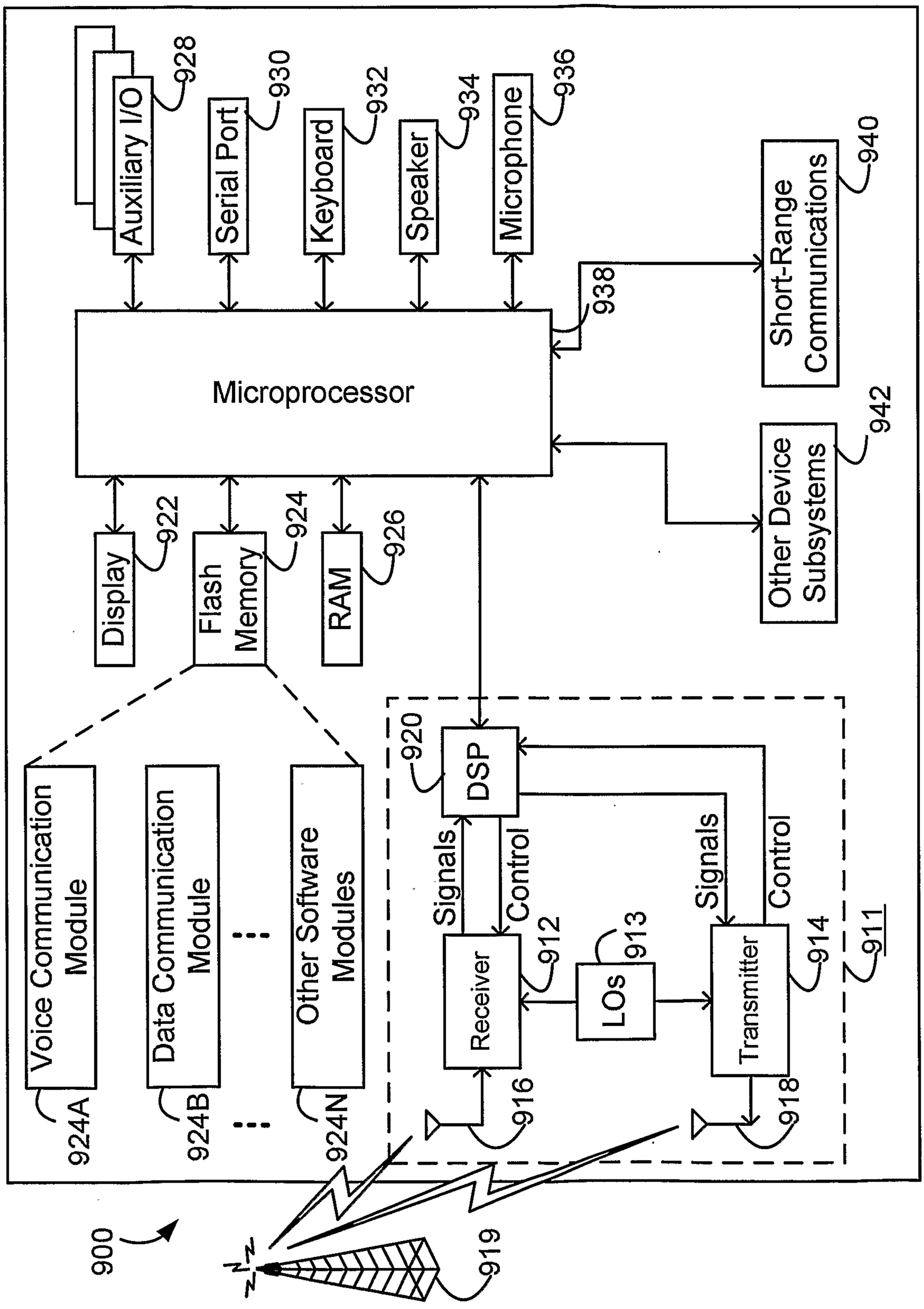
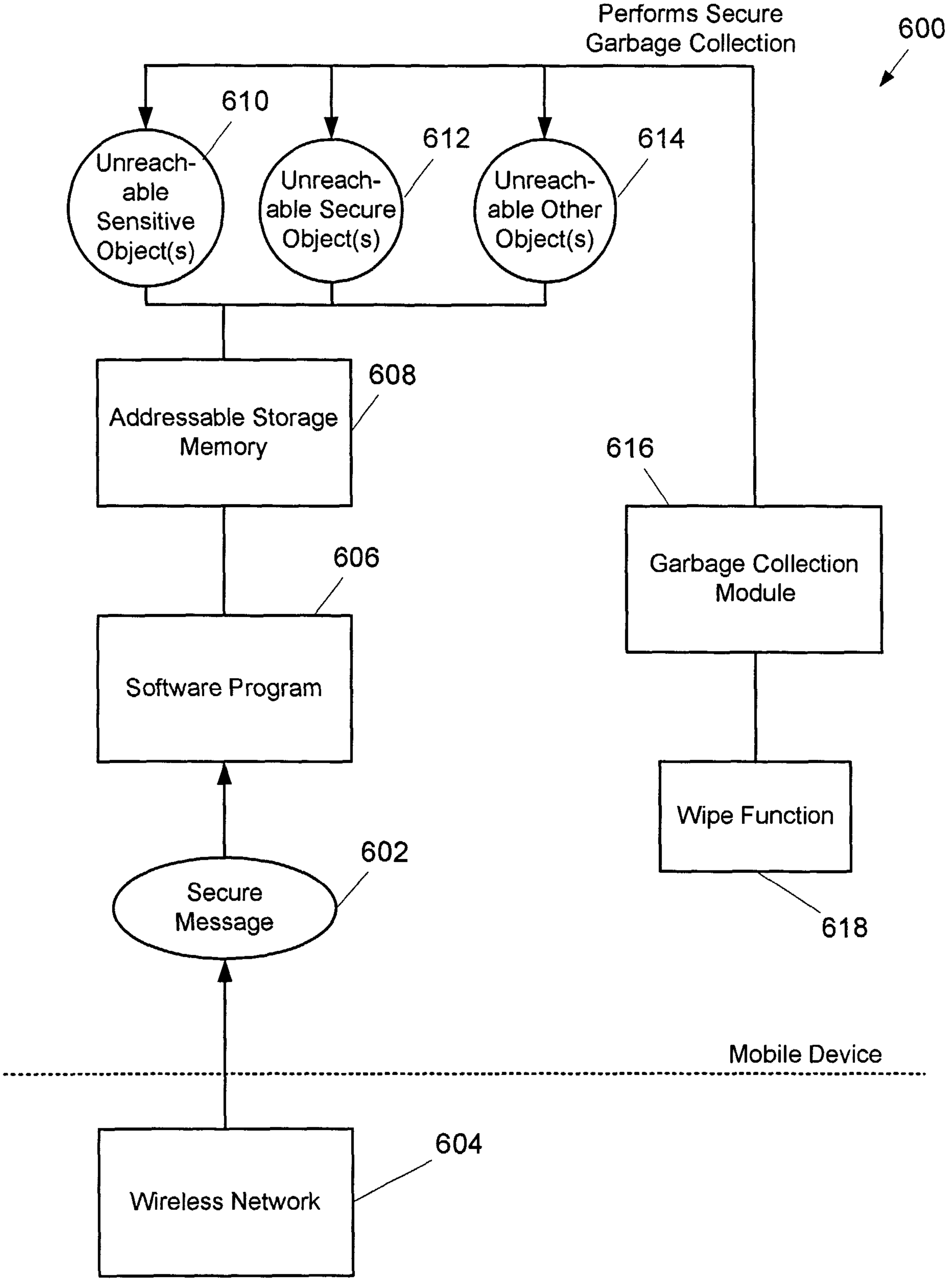

928
Auxiliary I/O
930
Serial Port
932
Keyboard
934
Speaker
936
Microphone
938
Microprocessor
940
Short-Range Communications
942
Other Device Subsystems
922
Display
924
Flash Memory
926
RAM
924A
Voice Communication Module
924B
Data Communication Module
924N
Other Software Modules
920
DSP
Signals
Control
912
Receiver
913
LOs
914
Transmitter
911
916
918
900
919

Fig. 7


Performs Secure Garbage Collection
600
610
612
614
Unreach-able Sensitive Object(s)
Unreach-able Secure Object(s)
Unreach-able Other Object(s)
Addressable Storage Memory
608
616
606
Garbage Collection Module
Software Program
Wipe Function
602
618
Secure Message
Mobile Device
604
Wireless Network