05/21/01

A

JC796 U.S. PTO 05/17/01

PATENT TRANSMITTAL LETTER
(SMALL ENTITY)

Attorney Docket No.
43426.00014

JC978 U.S. PTO 09/861229 05/17/01

TO THE COMMISSIONER FOR PATENTS:

Transmitted herewith for filing under 35 U.S.C. 111 and 37 C.F.R. is the patent application of:

Yigal Edery, **Nimrod Vered and David Kroll**

FOR:

Malicious Mobile Code Runtime Monitoring System and Methods:

☒ **Certificate of Mailing with Express Mailing Label No.: EL 701 364 462 US;**

☒ **10 Informal Sheets of Drawings: FIGS 1a-1c; 2, 3, 4; 5, 6a and 6b; 7a-7b and 8; 9 10A-10B; 11; 12a-12b**

☒ **Unsigned Combined Declaration and Power of Attorney;**

☒ **General Authorization and Request to Petition for Extension of Time; and**

☒ **Return Receipt Postcard**

CLAIMS AS FILED

FOR	FILED	ALLOWED	Extra	Rate	Additional Fee
Total Claims	76	-20	56	x $ 9.00	$ 504.00
Indep. Claims	11	-3	8	x $40.00	$ 320.00
Multiple Dependent Claims (check if applicable) ☐					$
				Basic Fee	$ 355.00
				Total Filing Fee	$1,179.00

☐ No additional fee is required for amendment.
☒ Please charge Deposit Account No. 05-0150 in the amount of $ 1,179.00
☒ The Commissioner is hereby authorized to charge and credit Deposit Account No. . 05-0150 As described below. A duplicate copy of this sheet is enclosed.

☒ Charge the amount of $1,179.00 as filing fee.
☒ Credit any overpayment.
☒ Charge any additional filing fees required under 37 C.F.R. 1.16.
☒ Charge any patent application processing fees under 37 C.F.R. 1.17.
☐ Charge the issue fee set in 37 C.F.R. 1.18 at the mailing of the Notice of Allowance, pursuant to 37 C.F.R. 1.311(b).

Daryl C. Josephson

Date: 5/17/01

Daryl C. Josephson Reg. No. 37,365
Attorney for Applicants
Squire, Sanders & Dempsey L.L.P.
600 Hansen Way
Palo Alto, CA 94304-1043
Telephone: (650) 856-6500
Facsimile: (650) 856-3619

APPLICATION FOR

UNITED STATES PATENT

IN THE NAME OF

Yigal Edery, Nimrod Vered and David Kroll

OF

FINJAN SOFTWARE, LTD.

MALICIOUS MOBILE CODE RUNTIME MONITORING SYSTEM AND METHODS

DOCKET NO. 43426.00014

Please direct communications to:

Intellectual Property Department
Squire, Sanders & Dempsey L.L.P.
600 Hansen Way
Palo Alto, CA 94304-1043
(650) 856-6500

Express Mail Number **EL 701 364 624**

MALICIOUS MOBILE CODE RUNTIME MONITORING SYSTEM AND METHODS

PRIORITY REFERENCE TO RELATED APPLICATIONS

This application claims benefit of and hereby incorporates by reference provisional application serial number 60/205,591, entitled "Computer Network Malicious Code Run-time Monitoring," filed on May 17, 2000 by inventors Nimrod Itzhak Vered, et al. This application is also a Continuation-In-Part of and hereby incorporates by reference patent application serial number 09/539,667, entitled "System and Method for Protecting a Computer and a Network From Hostile Downloadables" filed on March 30, 2000 by inventor Shlomo Touboul. This application is also a Continuation-In-Part of and hereby incorporates by reference patent application serial number 09/551,302, entitled "System and Method for Protecting a Client During Runtime From Hostile Downloadables", filed on April 18, 2000 by inventor Shlomo Touboul.

BACKGROUND OF THE INVENTION

Field of the Invention

This invention relates generally to computer networks, and more particularly provides a system and methods for protecting network-connectable devices from undesirable downloadable operation.

Description of the Background Art

Advances in networking technology continue to impact an increasing number and diversity of users. The Internet, for example, already provides to expert, intermediate and even novice users the informational, product and service resources of over 100,000 interconnected networks owned by governments, universities, nonprofit groups, companies, etc. Unfortunately, particularly the Internet and other public networks have also become a major source of potentially system-fatal or otherwise damaging computer code commonly referred to as "viruses."

Efforts to forestall viruses from attacking networked computers have thus far met with only limited success at best. Typically, a virus protection program designed to identify and remove or protect against the initiating of known viruses is installed on a network firewall or individually networked computer. The program is then inevitably surmounted by some new virus that often causes damage to one or more computers. The damage is then assessed and, if isolated, the new virus is analyzed. A corresponding new virus protection program (or update thereof) is then developed and installed to combat the new virus, and the new program operates successfully until yet another new virus appears - and so on. Of course, damage has already typically been incurred.

To make matters worse, certain classes of viruses are not well recognized or understood, let alone protected against. It is observed by this inventor, for example, that Downloadable information comprising program code can include distributable components (e.g. Java™ applets and JavaScript scripts, ActiveX™ controls, Visual Basic, add-ins and/or others). It can also include, for example, application programs, Trojan horses, multiple compressed programs such as zip or meta files, among others. U.S. Patent 5,983,348 to Shuang, however, teaches a protection system for protecting

against only distributable components including "Java applets or ActiveX controls", and further does so using resource intensive and high bandwidth static Downloadable content and operational analysis, and modification of the Downloadable component; Shuang further fails to detect or protect against additional program code included within a tested Downloadable. U.S. Patent 5,974,549 to Golan teaches a protection system that further focuses only on protecting against ActiveX controls and not other distributable components, let alone other Downloadable types. U.S. patent 6,167,520 to Touboul enables more accurate protection than Shuang or Golan, but lacks the greater flexibility and efficiency taught herein, as do Shuang and Golan.

Accordingly, there remains a need for efficient, accurate and flexible protection of computers and other network connectable devices from malicious Downloadables.

SUMMARY OF THE INVENTION

The present invention provides protection systems and methods capable of protecting a personal computer ("PC") or other persistently or even intermittently network accessible devices or processes from harmful, undesirable, suspicious or other "malicious" operations that might otherwise be effectuated by remotely operable code. While enabling the capabilities of prior systems, the present invention is not nearly so limited, resource intensive or inflexible, and yet enables more reliable protection. For example, remotely operable code that is protectable against can include downloadable application programs, Trojan horses and program code groupings, as well as software "components", such as Java™ applets, ActiveX™ controls, JavaScript™/Visual Basic scripts, add-ins, etc., among others. Protection can also be provided in a distributed

interactively, automatically or mixed configurable manner using protected client, server or other parameters, redirection, local/remote logging, etc., and other server/client based protection measures can also be separately and/or interoperably utilized, among other examples.

In one aspect, embodiments of the invention provide for determining, within one or more network "servers" (e.g. firewalls, resources, gateways, email relays or other devices/processes that are capable of receiving-and-transferring a Downloadable) whether received information includes executable code (and is a "Downloadable"). Embodiments also provide for delivering static, configurable and/or extensible remotely operable protection policies to a Downloadable-destination, more typically as a sandboxed package including the mobile protection code, downloadable policies and one or more received Downloadables. Further client-based or remote protection code/policies can also be utilized in a distributed manner. Embodiments also provide for causing the mobile protection code to be executed within a Downloadable-destination in a manner that enables various Downloadable operations to be detected, intercepted or further responded to via protection operations. Additional server/information-destination device security or other protection is also enabled, among still further aspects.

A protection engine according to an embodiment of the invention is operable within one or more network servers, firewalls or other network connectable information re-communicating devices (as are referred to herein summarily one or more "servers" or "re-communicators"). The protection engine includes an information monitor for monitoring information received by the server, and a code detection engine for determining whether the received information includes executable code. The protection

engine also includes a packaging engine for causing a sandboxed package, typically including mobile protection code and downloadable protection policies to be sent to a Downloadable-destination in conjunction with the received information, if the received information is determined to be a Downloadable.

A sandboxed package according to an embodiment of the invention is receivable by and operable with a remote Downloadable-destination. The sandboxed package includes mobile protection code ("MPC") for causing one or more predetermined malicious operations or operation combinations of a Downloadable to be monitored or otherwise intercepted. The sandboxed package also includes protection policies (operable alone or in conjunction with further Downloadable-destination stored or received policies/MPCs) for causing one or more predetermined operations to be performed if one or more undesirable operations of the Downloadable is/are intercepted. The sandboxed package can also include a corresponding Downloadable and can provide for initiating the Downloadable in a protective "sandbox". The MPC/policies can further include a communicator for enabling further MPC/policy information or "modules" to be utilized and/or for event logging or other purposes.

A sandbox protection system according to an embodiment of the invention comprises an installer for enabling a received MPC to be executed within a Downloadable-destination (device/process) and further causing a Downloadable application program, distributable component or other received downloadable code to be received and installed within the Downloadable-destination. The protection system also includes a diverter for monitoring one or more operation attempts of the Downloadable, an operation analyzer for determining one or more responses to the attempts, and a

security enforcer for effectuating responses to the monitored operations. The protection system can further include one or more security policies according to which one or more protection system elements are operable automatically (e.g. programmatically) or in conjunction with user intervention (e.g. as enabled by the security enforcer). The security policies can also be configurable/extensible in accordance with further downloadable and/or Downloadable-destination information.

A method according to an embodiment of the invention includes receiving downloadable information, determining whether the downloadable information includes executable code, and causing a mobile protection code and security policies to be communicated to a network client in conjunction with security policies and the downloadable information if the downloadable information is determined to include executable code. The determining can further provide multiple tests for detecting, alone or together, whether the downloadable information includes executable code.

A further method according to an embodiment of the invention includes forming a sandboxed package that includes mobile protection code ("MPC"), protection policies, and a received, detected-Downloadable, and causing the sandboxed package to be communicated to and installed by a receiving device or process ("user device") for responding to one or more malicious operation attempts by the detected-Downloadable from within the user device. The MPC/policies can further include a base "module" and a "communicator" for enabling further up/downloading of one or more further "modules" or other information (e.g. events, user/user device information, etc.).

Another method according to an embodiment of the invention includes installing, within a user device, received mobile protection code ("MPC") and protection policies in

conjunction with the user device receiving a downloadable application program, component or other Downloadable(s). The method also includes determining, by the MPC, a resource access attempt by the Downloadable, and initiating, by the MPC, one or more predetermined operations corresponding to the attempt. (Predetermined operations can, for example, comprise initiating user, administrator, client, network or protection system determinable operations, including but not limited to modifying the Downloadable operation, extricating the Downloadable, notifying a user/another, maintaining a local/remote log, causing one or more MPCs/policies to be downloaded, etc.)

Advantageously, systems and methods according to embodiments of the invention enable potentially damaging, undesirable or otherwise malicious operations by even unknown mobile code to be detected, prevented, modified and/or otherwise protected against without modifying the mobile code. Such protection is further enabled in a manner that is capable of minimizing server and client resource requirements, does not require pre-installation of security code within a Downloadable-destination, and provides for client specific or generic and readily updateable security measures to be flexibly and efficiently implemented. Embodiments further provide for thwarting efforts to bypass security measures (e.g. by "hiding" undesirable operation causing information within apparently inert or otherwise "friendly" downloadable information) and/or dividing or combining security measures for even greater flexibility and/or efficiency.

Embodiments also provide for determining protection policies that can be downloaded and/or ascertained from other security information (e.g. browser settings, administrative policies, user input, uploaded information, etc.). Different actions in response to different Downloadable operations, clients, users and/or other criteria are also

enabled, and embodiments provide for implementing other security measures, such as verifying a downloadable source, certification, authentication, etc. Appropriate action can also be accomplished automatically (e.g. programmatically) and/or in conjunction with alerting one or more users/administrators, utilizing user input, etc. Embodiments further enable desirable Downloadable operations to remain substantially unaffected, among other aspects.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1a is a block diagram illustrating a network system in accordance with an embodiment of the present invention;

FIG. 1b is a block diagram illustrating a network subsystem example in accordance with an embodiment of the invention;

FIG. 1c is a block diagram illustrating a further network subsystem example in accordance with an embodiment of the invention;

FIG. 2 is a block diagram illustrating a computer system in accordance with an embodiment of the invention;

FIG. 3 is a flow diagram broadly illustrating a protection system host according to an embodiment of the invention;

FIG. 4 is a block diagram illustrating a protection engine according to an embodiment of the invention;

FIG. 5 is a block diagram illustrating a content inspection engine according to an embodiment of the invention;

FIG. 6a is a block diagram illustrating protection engine parameters according to an embodiment of the invention;

FIG. 6b is a flow diagram illustrating a linking engine use in conjunction with ordinary, compressed and distributable sandbox package utilization, according to an embodiment of the invention;

FIG. 7a is a flow diagram illustrating a sandbox protection system operating within a destination system, according to an embodiment of the invention;

FIG. 7b is a block diagram illustrating memory allocation usable in conjunction with the protection system of FIG. 7a, according to an embodiment of the invention;

FIG. 7c is a block diagram illustrating a mobile protection code according to an embodiment of the invention;

FIG. 8 is a flowchart illustrating a method for examining a Downloadable in accordance with the present invention;

FIG. 9 is a flowchart illustrating a server based protection method according to an embodiment of the invention;

FIG. 10a is a flowchart illustrating method for determining if a potential-Downloadable includes or is likely to include executable code, according to an embodiment of the invention;

FIG. 10b is a flowchart illustrating a method for forming a protection agent, according to an embodiment of the invention;

FIG. 11 is a flowchart illustrating a method for protecting a Downloadable destination according to an embodiment of the invention;

FIG. 12a is a flowchart illustrating a method for forming a Downloadable access interceptor according to an embodiment of the invention; and

FIG. 12b is a flowchart illustrating a method for implementing mobile protection policies according to an embodiment of the invention.

DETAILED DESCRIPTION

In providing malicious mobile code runtime monitoring systems and methods, embodiments of the invention enable actually or potentially undesirable operations of even unknown malicious code to be efficiently and flexibly avoided. Embodiments provide, within one or more "servers" (e.g. firewalls, resources, gateways, email relays or other information re-communicating devices), for receiving downloadable-information and detecting whether the downloadable-information includes one or more instances of executable code (e.g. as with a Trojan horse, zip/meta file etc.). Embodiments also provide for separately or interoperably conducting additional security measures within the server, within a Downloadable-destination of a detected-Downloadable, or both.

Embodiments further provide for causing mobile protection code ("MPC") and downloadable protection policies to be communicated to, installed and executed within one or more received information destinations in conjunction with a detected-Downloadable. Embodiments also provide, within an information-destination, for detecting malicious operations of the detected-Downloadable and causing responses thereto in accordance with the protection policies (which can correspond to one or more user, Downloadable, source, destination, or other parameters), or further downloaded or downloadable-destination based policies (which can also be configurable or extensible). (Note that the term "or", as used herein, is generally intended to mean "and/or" unless otherwise indicated.)

FIGS. 1a through 1c illustrate a computer network system 100 according to an embodiment of the invention. FIG. 1a broadly illustrates system 100, while FIGS. 1b and

1c illustrate exemplary protectable subsystem implementations corresponding with system 104 or 106 of FIG. 1a.

Beginning with FIG. 1a, computer network system 100 includes an external computer network 101, such as a Wide Area Network or "WAN" (e.g. the Internet), which is coupled to one or more network resource servers (summarily depicted as resource server-1 102 and resource server-N 103). Where external network 101 includes the Internet, resource servers 1-N (102, 103) might provide one or more resources including web pages, streaming media, transaction-facilitating information, program updates or other downloadable information, summarily depicted as resources 121, 131 and 132. Such information can also include more traditionally viewed "Downloadables" or "mobile code" (i.e. distributable components), as well as downloadable application programs or other further Downloadables, such as those that are discussed herein. (It will be appreciated that interconnected networks can also provide various other resources as well.)

Also coupled via external network 101 are subsystems 104-106. Subsystems 104-106 can, for example, include one or more servers, personal computers ("PCs"), smart appliances, personal information managers or other devices/processes that are at least temporarily or otherwise intermittently directly or indirectly connectable in a wired or wireless manner to external network 101 (e.g. using a dialup, DSL, cable modem, cellular connection, IR/RF, or various other suitable current or future connection alternatives). One or more of subsystems 104-106 might further operate as user devices that are connectable to external network 101 via an internet service provider ("ISP") or

local area network ("LAN"), such as a corporate intranet, or home, portable device or smart appliance network, among other examples.

FIG. 1a also broadly illustrates how embodiments of the invention are capable of selectively, modifiably or extensibly providing protection to one or more determinable ones of networked subsystems 104-106 or elements thereof (not shown) against potentially harmful or other undesirable ("malicious") effects in conjunction with receiving downloadable information. "Protected" subsystem 104, for example, utilizes a protection in accordance with the teachings herein, while "unprotected" subsystem-N 105 employs no protection, and protected subsystem-M 106 might employ one or more protections including those according to the teachings herein, other protection, or some combination.

System 100 implementations are also capable of providing protection to redundant elements 107 of one or more of subsystems 104-106 that might be utilized, such as backups, failsafe elements, redundant networks, etc. Where included, such redundant elements are also similarly protectable in a separate, combined or coordinated manner using embodiments of the present invention either alone or in conjunction with other protection mechanisms. In such cases, protection can be similarly provided singly, as a composite of component operations or in a backup fashion. Care should, however, be exercised to avoid potential repeated protection engine execution corresponding to a single Downloadable; such "chaining" can cause a Downloadable to operate incorrectly or not at all, unless a subsequent detection engine is configured to recognize a prior packaging of the Downloadable..

FIGS. 1b and 1c further illustrate, by way of example, how protection systems according to embodiments of the invention can be utilized in conjunction with a wide variety of different system implementations. In the illustrated examples, system elements are generally configurable in a manner commonly referred to as a "client-server" configuration, as is typically utilized for accessing Internet and many other network resources. For clarity sake, a simple client-server configuration will be presumed unless otherwise indicated. It will be appreciated, however, that other configurations of interconnected elements might also be utilized (e.g. peer-peer, routers, proxy servers, networks, converters, gateways, services, network reconfiguring elements, etc.) in accordance with a particular application.

The FIG. 1b example shows how a suitable protected system 104a (which can correspond to subsystem-1 104 or subsystem-M 106 of FIG. 1) can include a protection-initiating host "server" or "re-communicator" (e.g. ISP server140a), one or more user devices or "Downloadable-destinations" 145, and zero or more redundant elements (which elements are summarily depicted as redundant client device/process 145a). In this example, ISP server 140a includes one or more email, Internet or other servers 141a, or other devices or processes capable of transferring or otherwise "re-communicating" downloadable information to user devices 145. Server 141a further includes protection engine or "PE" 142a, which is capable of supplying mobile protection code ("MPC") and protection policies for execution by client devices 145. One or more of user devices 145 can further include a respective one or more clients 146 for utilizing information received via server 140a, in accordance with which MPC and protection policies are operable to

protect user devices 145 from detrimental, undesirable or otherwise "malicious" operations of downloadable information also received by user device 145.

The FIG. 1c example shows how a further suitable protected system 104b can include, in addition to a "re-communicator", such as server 142b, a firewall 143c (e.g. as is typically the case with a corporate intranet and many existing or proposed home/smart networks.) In such cases, a server 141b or firewall 143 can operate as a suitable protection engine host. A protection engine can also be implemented in a more distributed manner among two or more protection engine host systems or host system elements, such as both of server 141b and firewall 143, or in a more integrated manner, for example, as a standalone device. Redundant system or system protection elements can also be similarly provided in a more distributed or integrated manner (see above).

System 104b also includes internal network 144 and user devices 145. User devices 145 further include a respective one or more clients 146 for utilizing information received via server 140a, in accordance with which the MPCs or protection policies are operable. (As in the previous example, one or more of user devices 145 can also include or correspond with similarly protectable redundant system elements, which are not shown.)

It will be appreciated that the configurations of FIGS 1a-1c are merely exemplary. Alternative embodiments might, for example, utilize other suitable connections, devices or processes. One or more devices can also be configurable to operate as a network server, firewall, smart router, a resource server servicing deliverable third-party/manufacturer postings, a user device operating as a firewall/server, or other information-suppliers or intermediaries (i.e. as a "re-communicator" or "server") for

servicing one or more further interconnected devices or processes or interconnected levels of devices or processes. Thus, for example, a suitable protection engine host can include one or more devices or processes capable of providing or supporting the providing of mobile protection code or other protection consistent with the teachings herein. A suitable information-destination or "user device" can further include one or more devices or processes (such as email, browser or other clients) that are capable of receiving and initiating or otherwise hosting a mobile code execution.

FIG. 2 illustrates an exemplary computing system 200, that can comprise one or more of the elements of FIGS. 1a through 1c. While other application-specific alternatives might be utilized, it will be presumed for clarity sake that system 100 elements (FIGS. 1a-c) are implemented in hardware, software or some combination by one or more processing systems consistent therewith, unless otherwise indicated.

Computer system 200 comprises elements coupled via communication channels (e.g. bus 201) including one or more general or special purpose processors 202, such as a Pentium® or Power PC®, digital signal processor ("DSP"), etc. System 200 elements also include one or more input devices 203 (such as a mouse, keyboard, microphone, pen, etc.), and one or more output devices 204, such as a suitable display, speakers, actuators, etc., in accordance with a particular application.

System 200 also includes a computer readable storage media reader 205 coupled to a computer readable storage medium 206, such as a storage/memory device or hard or removable storage/memory media; such devices or media are further indicated separately as storage device 208 and memory 209, which can include hard disk variants, floppy/compact disk variants, digital versatile disk ("DVD") variants, smart cards, read

only memory, random access memory, cache memory, etc., in accordance with a particular application. One or more suitable communication devices 207 can also be included, such as a modem, DSL, infrared or other suitable transceiver, etc. for providing inter-device communication directly or via one or more suitable private or public networks that can include but are not limited to those already discussed.

Working memory further includes operating system ("OS") elements and other programs, such as application programs, mobile code, data, etc. for implementing system 100 elements that might be stored or loaded therein during use. The particular OS can vary in accordance with a particular device, features or other aspects in accordance with a particular application (e.g. Windows, Mac, Linux, Unix or Palm OS variants, a proprietary OS, etc.). Various programming languages or other tools can also be utilized, such as C++, Java, Visual Basic, etc. As will be discussed, embodiments can also include a network client such as a browser or email client, e.g. as produced by Netscape, Microsoft or others, a mobile code executor such as an OS task manager, Java Virtual Machine ("JVM"), etc., and an application program interface ("API"), such as a Microsoft Windows or other suitable element in accordance with the teachings herein. (It will also become apparent that embodiments might also be implemented in conjunction with a resident application or combination of mobile code and resident application components.)

One or more system 200 elements can also be implemented in hardware, software or a suitable combination. When implemented in software (e.g. as an application program, object, downloadable, servlet, etc. in whole or part), a system 200 element can be communicated transitionally or more persistently from local or remote storage to

memory (or cache memory, etc.) for execution, or another suitable mechanism can be utilized, and elements can be implemented in compiled or interpretive form. Input, intermediate or resulting data or functional elements can further reside more transitionally or more persistently in a storage media, cache or more persistent volatile or non-volatile memory, (e.g. storage device 207 or memory 208) in accordance with a particular application.

FIG. 3 illustrates an interconnected re-communicator 300 generally consistent with system 140b of FIG. 1, according to an embodiment of the invention. As with system 140b, system 300 includes a server 301, and can also include a firewall 302. In this implementation, however, either server 301 or firewall 302 (if a firewall is used) can further include a protection engine (310 or 320 respectively). Thus, for example, an included firewall can process received information in a conventional manner, the results of which can be further processed by protection engine 310 of server 301, or information processed by protection engine 320 of an included firewall 302 can be processed in a conventional manner by server 301. (For clarity sake, a server including a singular protection engine will be presumed, with or without a firewall, for the remainder of the discussion unless otherwise indicated. Note, however, that other embodiments consistent with the teachings herein might also be utilized.)

FIG. 3 also shows how information received by server 301 (or firewall 302) can include non-executable information, executable information or a combination of non-executable and one or more executable code portions (e.g. so-called Trojan horses that include a hostile Downloadable within a friendly one, combined, compressed or otherwise encoded files, etc.). Particularly such combinations will likely remain

undetected by a firewall or other more conventional protection systems. Thus, for convenience, received information will also be referred to as a "potential-Downloadable", and received information found to include executable code will be referred to as a "Downloadable" or equivalently as a "detected-Downloadable" (regardless of whether the executable code includes one or more application programs, distributable "components" such as Java, ActiveX, add-in, etc.).

Protection engine 310 provides for detecting whether received potential-Downloadables include executable code, and upon such detection, for causing mobile protection code ("MPC") to be transferred to a device that is a destination of the potential-Downloadable (or "Downloadable-destination"). Protection engine 310 can also provide protection policies in conjunction with the MPC (or thereafter as well), which MPC/policies can be automatically (e.g. programmatically) or interactively configurable in accordance user, administrator, downloadable source, destination, operation, type or various other parameters alone or in combination (see below). Protection engine 310 can also provide or operate separately or interoperably in conjunction with one or more of certification, authentication, downloadable tagging, source checking, verification, logging, diverting or other protection services via the MPC, policies, other local/remote server or destination processing, etc. (e.g. which can also include protection mechanisms taught by the above-noted prior applications; see FIG. 4).

Operationally, protection engine 310 of server 301 monitors information received by server 301 and determines whether the received information is deliverable to a protected destination, e.g. using a suitable monitor/data transfer mechanism and comparing a destination-address of the received information to a protected destination set,

such as a protected destinations list, array, database, etc. (All deliverable information or one or more subsets thereof might also be monitored.) Protection engine 310 further analyzes the potential-Downloadable and determines whether the potential-Downloadable includes executable code. If not, protection engine 310 enables the not executable potential-Downloadable 331 to be delivered to its destination in an unaffected manner.

In conjunction with determining that the potential-Downloadable is a detected-Downloadable, protection engine 310 also causes mobile protection code or "MPC" 341 to be communicated to the Downloadable-destination of the Downloadable, more suitably in conjunction with the detected-Downloadable 343 (see below). Protection engine 310 further causes downloadable protection policies 342 to be delivered to the Downloadable-destination, again more suitably in conjunction with the detected-Downloadable. Protection policies 342 provide parameters (or can additionally or alternatively provide additional mobile code) according to which the MPC is capable of determining or providing applicable protection to a Downloadable-destination against malicious Downloadable operations.

(One or more "checked", tag, source, destination, type, detection or other security result indicators, which are not shown, can also be provided as corresponding to determined non-Downloadables or Downloadables, e.g. for testing, logging, further processing, further identification tagging or other purposes in accordance with a particular application.)

Further MPCs, protection policies or other information are also deliverable to a the same or another destination, for example, in accordance with communication by an MPC/protection policies already delivered to a downloadable-destination. Initial or

subsequent MPCs/policies can further be selected or configured in accordance with a Downloadable-destination indicated by the detected-Downloadable, destination-user or administrative information, or other information providable to protection engine 310 by a user, administrator, user system, user system examination by a communicated MPC, etc. (Thus, for example, an initial MPC/policies can also be initially provided that are operable with or optimized for more efficient operation with different Downloadable-destinations or destination capabilities.)

While integrated protection constraints within the MPC might also be utilized, providing separate protection policies has been found to be more efficient, for example, by enabling more specific protection constraints to be more easily updated in conjunction with detected-Downloadable specifics, post-download improvements, testing, etc. Separate policies can further be more efficiently provided (e.g. selected, modified, instantiated, etc.) with or separately from an MPC, or in accordance with the requirements of a particular user, device, system, administration, later improvement, etc., as might also be provided to protection engine 310 (e.g. via user/MPC uploading, querying, parsing a Downloadable, or other suitable mechanism implemented by one or more servers or Downloadable-destinations).

(It will also become apparent that performing executable code detection and communicating to a downloadable-Destination an MPC and any applicable policies as separate from a detected-Downloadable is more accurate and far less resource intensive than, for example, performing content and operation scanning, modifying a Downloadable, or providing completely Downloadable-destination based security.)

System 300 enables a single or extensible base-MPC to be provided, in anticipation or upon receipt of a first Downloadable, that is utilized thereafter to provide protection of one or more Downloadable-destinations. It is found, however, that providing an MPC upon each detection of a Downloadable (which is also enabled) can provide a desirable combination of configurability of the MPC/policies and lessened need for management (e.g. given potentially changing user/destination needs, enabling testing, etc.).

Providing an MPC upon each detection of a Downloadable also facilitates a lessened demand on destination resources, e.g. since information-destination resources used in executing the MPC/policies can be re-allocated following such use. Such alternatives can also be selectively, modifiably or extensibly provided (or further in accordance with other application-specific factors that might also apply.) Thus, for example, a base-MPC or base-policies might be provided to a user device that is/are extensible via additionally downloadable "modules" upon server 301 detection of a Downloadable deliverable to the same user device, among other alternatives.

In accordance with a further aspect of the invention, it is found that improved efficiency can also be achieved by causing the MPC to be executed within a Downloadable-destination in conjunction with, and further, prior to initiation of the detected Downloadable. One mechanism that provides for greater compatibility and efficiency in conjunction with conventional client-based Downloadable execution is for a protection engine to form a sandboxed package 340 including MPC 341, the detected-Downloadable 343 and any policies 342. For example, where the Downloadable is a binary executable to be executed by an operating system, protection engine 310 forms a

protected package by concatenating, within sandboxed package 340, MPC 341 for delivery to a Downloadable-destination first, followed by protection policies 342 and Downloadable 343. (Concatenation or techniques consistent therewith can also be utilized for providing a protecting package corresponding to a Java applet for execution by a JVM of a Downloadable-destination, or with regard to ActiveX controls, add-ins or other distributable components, etc.)

The above concatenation or other suitable processing will result in the following. Upon receipt of sandboxed package 340 by a compatible browser, email or other destination-client and activating of the package by a user or the destination-client, the operating system (or a suitable responsively initiated distributed component host) will attempt to initiate sandboxed package 340 as a single Downloadable. Such processing will, however, result in initiating the MPC 341 and -in accordance with further aspects of the invention- the MPC will initiate the Downloadable in a protected manner, further in accordance with any applicable included or further downloaded protection policies 342. (While system 300 is also capable of ascertaining protection policies stored at a Downloadable-destination, e.g. by poll, query, etc. of available destination information, including at least initial policies within a suitable protecting package is found to avoid associated security concerns or inefficiencies.)

Turning to FIG. 4, a protection engine 400 generally consistent with protection engine 310 (or 320) of FIG. 3 is illustrated in accordance with an embodiment of the invention. Protection engine 400 comprises information monitor 401, detection engine 402, and protected packaging engine 403, which further includes agent generator 431, storage 404, linking engine 405, and transfer engine 406. Protection engine 400 can also

include a buffer 407, for temporarily storing a received potential-Downloadable, or one or more systems for conducting additional authentication, certification, verification or other security processing (e.g. summarily depicted as security system 408) Protection engine 400 can further provide for selectively re-directing, further directing, logging, etc. of a potential/detected Downloadable or information corresponding thereto in conjunction with detection, other security, etc., in accordance with a particular application.

(Note that FIG. 4, as with other figures included herein, also depicts exemplary signal flow arrows; such arrows are provided to facilitate discussion, and should not be construed as exclusive or otherwise limiting.)

Information monitor 401 monitors potential-Downloadables received by a host server and provides the information via buffer 407 to detection engine 402 or to other system 400 elements. Information monitor 401 can be configured to monitor host server download operations in conjunction with a user or a user-device that has logged-on to the server, or to receive information via a server operation hook, servlet, communication channel or other suitable mechanism.

Information monitor 401 can also provide for transferring, to storage 404 or other protection engine elements, configuration information including, for example, user, MPC, protection policy, interfacing or other configuration information (e.g. see FIG. 6). Such configuration information monitoring can be conducted in accordance with a user/device logging onto or otherwise accessing a host server, via one or more of configuration operations, using an applet to acquire such information from or for a particular user, device or devices, via MPC/policy polling of a user device, or via other suitable mechanisms.

Detection engine 402 includes code detector 421, which receives a potential-Downloadable and determines, more suitably in conjunction with inspection parameters 422, whether the potential-Downloadable includes executable code and is thus a "detected-Downloadable". (Code detector 421 can also include detection processors for performing file decompression or other "decoding", or such detection-facilitating processing as decryption, utilization/support of security system 408, etc. in accordance with a particular application.)

Detection engine 402 further transfers a detected-downloadable ("XEQ") to protected packaging engine 403 along with indicators of such detection, or a determined non-executable ("NXEQ") to transfer engine 406. (Inspection parameters 422 enable analysis criteria to be readily updated or varied, for example, in accordance with particular source, destination or other potential Downloadable impacting parameters, and are discussed in greater detail with reference to FIG. 5). Detection engine 402 can also provide indicators for delivery of initial and further MPCs/policies, for example, prior to or in conjunction with detecting a Downloadable and further upon receipt of an indicator from an already downloaded MPC/policy. A downloaded MPC/policy can further remain resident at a user device with further modules downloaded upon or even after delivery of a sandboxed package. Such distribution can also be provided in a configurable manner, such that delivery of a complete package or partial packages are automatically or interactively determinable in accordance with user/administrative preferences/policies, among other examples.

Packaging engine 403 provides for generating mobile protection code and protection policies, and for causing delivery thereof (typically with a detected-

Downloadable) to a Downloadable-destination for protecting the Downloadable-destination against malicious operation attempts by the detected Downloadable. In this example, packaging engine 403 includes agent generator 431, storage 404 and linking engine 405.

Agent generator 431 includes an MPC generator 432 and a protection policy generator 433 for "generating" an MPC and a protection policy (or set of policies) respectively upon receiving one or more "generate MPC/policy" indicators from detection engine 402, indicating that a potential-Downloadable is a detected-Downloadable. MPC generator 432 and protection policy generator 433 provide for generating MPCs and protection policies respectively in accordance with parameters retrieved from storage 404. Agent generator 431 is further capable of providing multiple MPCs/policies, for example, the same or different MPCs/policies in accordance with protecting ones of multiple executables within a zip file, or for providing initial MPCs/policies and then further MPCs/policies or MPC/policy "modules" as initiated by further indicators such as given above, via an indicator of an already downloaded MPC/policy or via other suitable mechanisms. (It will be appreciated that pre-constructed MPCs/policies or other processing can also be utilized, e.g. via retrieval from storage 404, but with a potential decrease in flexibility.)

MPC generator 432 and protection policy generator 433 are further configurable. Thus, for example, more generic MPCs/policies can be provided to all or a grouping of serviced destination-devices (e.g. in accordance with a similarly configured/administered intranet), or different MPCs/policies that can be configured in accordance with one or more of user, network administration, Downloadable-destination or other parameters (e.g.

see FIG. 6). As will become apparent, a resulting MPC provides an operational interface to a destination device/process. Thus, a high degree of flexibility and efficiency is enabled in providing such an operational interface within different or differently configurable user devices/processes or other constraints.

Such configurability further enables particular policies to be utilized in accordance with a particular application (e.g. particular system uses, access limitations, user interaction, treating application programs or Java components from a particular known source one way and unknown source ActiveX components, or other considerations). Agent generator 431 further transfers a resulting MPC and protection policy pair to linking engine 405.

Linking engine 405 provides for forming from received component elements (see above) a sandboxed package that can include one or more initial or complete MPCs and applicable protection policies, and a Downloadable, such that the sandboxed package will protect a receiving Downloadable-destination from malicious operation by the Downloadable. Linking engine 405 is implementable in a static or configurable manner in accordance, for example, with characteristics of a particular user device/process stored intermittently or more persistently in storage 404. Linking engine 405 can also provide for restoring a Downloadable, such as a compressed, encrypted or otherwise encoded file that has been decompressed, decrypted or otherwise decoded via detection processing (e.g. see FIG. 6b).

It is discovered, for example, that the manner in which the Windows OS initiates a binary executable or an ActiveX control can be utilized to enable protected initiation of a detected-Downloadable. Linking engine 405 is, for example, configurable to form, for

an ordinary single-executable Downloadable (e.g. an application program, applet, etc.) a sandboxed package 340 as a concatenation of ordered elements including an MPC 341, applicable policies 342 and the Downloadable or "XEQ" 343 (e.g. see FIG. 4).

Linking engine 405 is also configurable to form, for a Downloadable received by a server as a compressed single or multiple-executable Downloadable such as a zipped or meta file, a protecting package 340 including one or more MPCs, applicable policies and the one or more included executables of the Downloadable. For example, a sandboxed package can be formed in which a single MPC and policies precede and thus will affect all such executables as a result of inflating and installation. An MPC and applicable policies can also, for example, precede each executable, such that each executable will be separately sandboxed in the same or a different manner according to MPC/policy configuration (see above) upon inflation and installation. (See also FIGS. 5 and 6)

Linking engine is also configurable to form an initial MPC, MPC-policy or sandboxed package (e.g. prior to upon receipt of a downloadable) or an additional MPC, MPC-policy or sandboxed package (e.g. upon or following receipt of a downloadable), such that suitable MPCs/policies can be provided to a Downloadable-destination or other destination in a more distributed manner. In this way, requisite bandwidth or destination resources can be minimized (via two or more smaller packages) in compromise with latency or other considerations raised by the additional required communication.

A configurable linking engine can also be utilized in accordance with other requirements of particular devices/processes, further or different elements or other permutations in accordance with the teachings herein. (It might, for example be desirable to modify the ordering of elements, to provide one or more elements separately, to

provide additional information, such as a header, etc., or perform other processing in accordance with a particular device, protocol or other application considerations.)

Policy/authentication reader-analyzer 481 summarily depicts other protection mechanisms that might be utilized in conjunction with Downloadable detection, such as already discussed, and that can further be configurable to operate in accordance with policies or parameters (summarily depicted by security/authentication policies 482). Integration of such further protection in the depicted configuration, for example, enables a potential-Downloadable from a known unfriendly source, a source failing authentication or a provided-source that is confirmed to be fictitious to be summarily discarded, otherwise blocked, flagged, etc. (with or without further processing). Conversely, a potential-Downloadable from a known friendly source (or one confirmed as such) can be transferred with or without further processing in accordance with particular application considerations. (Other configurations including pre or post Downloadable detection mechanisms might also be utilized.)

Finally, transfer engine 406 of protection agent engine 303 provides for receiving and causing linking engine 405 (or other protection) results to be transferred to a destination user device/process. As depicted, transfer engine 406 is configured to receive and transfer a Downloadable, a determined non-executable or a sandboxed package. However, transfer engine 406 can also be provided in a more configurable manner, such as was already discussed for other system 400 elements. (Any one or more of system 400 elements might be configurably implemented in accordance with a particular application.) Transfer engine 406 can perform such transfer, for example, by adding the information to a server transfer queue (not shown) or utilizing another suitable method.

Turning to FIG. 5 with reference to FIG. 4, a code detector 421 example is illustrated in accordance with an embodiment of the invention. As shown, code detector 421 includes data fetcher 501, parser 502, file-type detector 503, inflator 504 and control 506; other depicted elements. While implementable and potentially useful in certain instances, are found to require substantial overhead, to be less accurate in certain instances (see above) and are not utilized in a present implementation; these will be discussed separately below. Code detector elements are further configurable in accordance with stored parameters retrievable by data fetcher 501. (A coupling between data fetcher 501 and control 506 has been removed for clarity sake.)

Data fetcher 501 provides for retrieving a potential-Downloadable or portions thereof stored in buffer 407 or parameters from storage 404, and communicates such information or parameters to parser 502. Parser 502 receives a potential-Downloadable or portions thereof from data fetcher 501 and isolates potential-Downloadable elements, such as file headers, source, destination, certificates, etc. for use by further processing elements.

File type detector 502 receives and determines whether the potential-Downloadable (likely) is or includes an executable file type. File-reader 502 can, for example, be configured to analyze a received potential-Downloadable for a file header, which is typically included in accordance with conventional data transfer protocols, such as a portable executable or standard ".exe" file format for Windows OS application programs, a Java class header for Java applets, and so on for other applications, distributed components, etc. "Zipped", meta or other compressed files, which might include one or more executables, also typically provide standard single or multi-level

headers that can be read and used to identify included executable code (or other included information types). File type detector 502 is also configurable for analyzing potential-Downloadables for all potential file type delimiters or a more limited subset of potential file type delimiters (e.g. ".exe" or ".com" in conjunction with a DOS or Microsoft Windows OS Downloadable-destination).

Known file type delimiters can, for example, be stored in a more temporary or more persistent storage (e.g. storage 404 of FIG. 4) which file type detector 502 can compare to a received potential-Downloadable. (Such delimiters can thus also be updated in storage 404 as a new file type delimiter is provided, or a more limited subset of delimiters can also be utilized in accordance with a particular Downloadable-destination or other considerations of a particular application.) File type detector 502 further transfers to controller 506 a detected file type indicator indicating that the potential-Downloadable includes or does not include (i.e. or likely include) an executable file type.

In this example, the aforementioned detection processor is also included as pre-detection processor or, more particularly, a configurable file inflator 504. File inflator 504 provides for opening or "inflating" compressed files in accordance with a compressed file type received from file type detector 503 and corresponding file opening parameters received from data fetcher 501. Where a compressed file (e.g. a meta file) includes nested file type information not otherwise reliably provided in an overall file header or other information, inflator 504 returns such information to parser 502. File inflator 504 also provides any now-accessible included executables to control 506 where one or more included files are to be separately packaged with an MPC or policies.

Control 506, in this example, operates in accordance with stored parameters and provides for routing detected non-Downloadables or Downloadables and control information, and for conducting the aforementioned distributed downloading of packages to Downloadable-destinations. In the case of a non-Downloadable, for example, control 506 sends the non-Downloadable to transfer engine 406 (FIG. 4) along with any indicators that might apply. For an ordinary single-executable Downloadable, control 506 sends control information to agent generator 431 and the Downloadable to linking engine 405 along with any other applicable indicators (see 641 of FIG. 6b). Control 506 similarly handles a compressed single-executable Downloadable or a multiple downloadable to be protected using a single sandboxed package. For a multiple-executable Downloadable, control 506 sends control information for each corresponding executable to agent generator agent generator 431, and sends the executable to linking engine 405 along with controls and any applicable indicators, as in 643b of FIG. 6b. (The above assumes, however, that distributed downloading is not utilized; when used – according to applicable parameters- control 506 also operates in accordance with the following.)

Control 506 conducts distributed protection (e.g. distributed packaging) by providing control signals to agent generator 431, linking engine 405 and transfer engine 406. In the present example, control 506 initially sends controls to agent generator 431 and linking engine 405 (FIG. 4) causing agent generator to generate an initial MPC and initial policies, and sends control and a detected-Downloadable to linking engine 405. Linking engine 405 forms an initial sandboxed package, which transfer engine causes (in conjunction with further controls) to be downloaded to the Downloadable destination

(643a of FIG. 6b). An initial MPC within the sandboxed package includes an installer and a communicator and performs installation as indicated below. The initial MPC also communicates via the communicator controls to control 506 (FIG. 5) in response to which control 506 similarly causes generation of MPC-M and policy-M modules 643c, which linking engine 405 links and transfer engine 406 causes to be sent to the Downloadable destination, and so on for any further such modules.

(It will be appreciated, however, that an initial package might be otherwise configured or sent prior to receipt of a Downloadable in accordance with configuration parameters or user interaction. Information can also be sent to other user devices, such as that of an administrator. Further MPCs/policies might also be coordinated by control 506 or other elements, or other suitable mechanisms might be utilized in accordance with the teachings herein.)

Regarding the remaining detection engine elements illustrated in FIG. 5, where content analysis is utilized, parser 502 can also provide a Downloadable or portions thereof to content detector 505. Content detector 505 can then provide one or more content analyses. Binary detector 551, for example, performs detection of binary information; pattern detector 552 further analyzes the Downloadable for patterns indicating executable code, or other detectors can also be utilized. Analysis results therefrom can be used in an absolute manner, where a first testing result indicating executable code confirms Downloadable detection, which result is then sent to control 506. Alternatively, however, composite results from such analyses can also be sent to control 506 for evaluation. Control 506 can further conduct such evaluation in a summary manner (determining whether a Downloadable is detected according to a

majority or minimum number of indicators), or based on a weighting of different analysis results. Operation then continues as indicated above. (Such analysis can also be conducted in accordance with aspects of a destination user device or other parameters.)

FIG. 6a illustrates more specific examples of indicators/parameters and known (or "knowledge base") elements that can be utilized to facilitate the above-discussed system 400 configurability and detection. For clarity sake, indicators, parameters and knowledge base elements are combined as indicated "parameters." It will be appreciated, however, that the particular parameters utilized can differ in accordance with a particular application, and indicators, parameters or known elements, where utilized, can vary and need not correspond exactly with one another. Any suitable explicit or referencing list, database or other storage structure(s) or storage structure configuration(s) can also be utilized to implement a suitable user/device based protection scheme, such as in the above examples, or other desired protection schema.

Executable parameters 601 comprise, in accordance with the above examples, executable file type parameters 611, executable code parameters 612 and code pattern parameters 613 (including known executable file type indicators, header/code indicators and patterns respectively, where code patterns are utilized). Use parameters 602 further comprise user parameters 621, system parameters 622 and general parameters 623 corresponding to one or more users, user classifications, user-system correspondences or destination system, device or processes, etc. (e.g. for generating corresponding MPCs/policies, providing other protection, etc.). The remaining parameters include interface parameters 631 for providing MPC/policy (or further) configurability in

accordance with a particular device or for enabling communication with a device user (see below), and other parameters 632.

FIG. 6b illustrates a linking engine 405 according to an embodiment of the invention. As already discussed, linking engine 405 includes a linker for combining MPCs, policies or agents via concatination or other suitable processing in accordance with an OS, JVM or other host executor or other applicable factors that might apply. Linking engine 405 also includes the aforementioned post-detection processor which, in this example, comprises a compressor 508. As noted, compressor 508 receives linked elements from linker 507 and, where a potential-Downloadable corresponds to a compressed file that was inflated during detection, re-forms the compressed file. (Known file information can be provided via configuration parameters, substantially reversal of inflating or another suitable method.) Encryption or other post-detection processing can also be conducted by linking engine 508.

FIGS. 7a, 7b and 8 illustrate a "sandbox protection" system, as operable within a receiving destination-device, according to an embodiment of the invention.

Beginning with FIG. 7a, a client 146 receiving sandbox package 340 will "recognize" sandbox package 340 as a (mobile) executable and cause a mobile code installer 711 (e.g. an OS loader, JVM, etc.) to be initiated. Mobile code installer 711 will also recognize sandbox package 340 as an executable and will attempt to initiate sandbox package 340 at its "beginning." Protection engine 400 processing corresponding to destination 700 use of a such a loader, however, will have resulted in the "beginning" of sandbox package 340 as corresponding to the beginning of MPC 341, as noted with regard to the above FIG. 4 example.

Such protection engine processing will therefore cause a mobile code installer (e.g. OS loader 711, for clarity sake) to initiate MPC 341. In other cases, other processing might also be utilized for causing such initiation or further protection system operation. Protection engine processing also enables MPC 341 to effectively form a protection "sandbox" around Downloadable (e.g. detected-Downloadable or "XEQ") 343, to monitor Downloadable 343, intercept determinable Downloadable 343 operation (such as attempted accesses of Downloadable 343 to destination resources) and, if "malicious", to cause one or more other operations to occur (e.g. providing an alert, offloading the Downloadable, offloading the MPC, providing only limited resource access, possibly in a particular address space or with regard to a particularly "safe" resource or resource operation, etc.).

MPC 341, in the present OS example, executes MPC element installation and installs any policies, causing MPC 341 and protection policies 342 to be loaded into a first memory space, P1. MPC 341 then initiates loading of Downloadable 343. Such Downloadable initiation causes OS loader 711 to load Downloadable 343 into a further working memory space-P2 703 along with an API import table ("IAT") 731 for providing Downloadable 631 with destination resource access capabilities. It is discovered, however that the IAT can be modified so that any call to an API can be redirected to a function within the MPC. The technique for modifying the IAT is documented within the MSDN (Microsoft Developers Network) Library CD in several articles. The technique is also different for each operating system (e.g. between Windows 9x and Windows NT), which can be accommodated by agent generator configurability, such as that given above.

MPC 341 therefore has at least initial access to API IAT 731 of Downloadable 632, and provides for diverting, evaluating and responding to attempts by Downloadable 632 to utilize system APIs 731, or further in accordance with protection policies 342.

In addition to API diverting, MPC 341 can also install filter drivers, which can be used for controlling access to resources such as a Downloadable-destination file system or registry. Filter driver installation can be conducted as documented in the MSDN or using other suitable methods.

Turning to FIG. 8 with reference to FIG. 7b, an MPC 341 according to an embodiment of the invention includes a package extractor 801, executable installer 802, sandbox engine installer 803, resource access diverter 804, resource access (attempt) analyzer 805, policy enforcer 806 and MPC de-installer 807. Package extractor 801 is initiated upon initiation of MPC 341, and extracts MPC 341 elements and protection policies 342. Executable installer 802 further initiates installation of a Downloadable by extracting the downloadable from the protected package, and loading the process into memory in suspended mode (so it only loads into memory, but does not start to run). Such installation further causes the operating system to initialize the Downloadable's IAT 731 in the memory space of the downloadable process, P2, as already noted.

Sandbox engine installer 803 (running in process space P1) then installs the sandbox engine (803-805) and policies 342 into the downloadable process space P2. This is done in different way in each operating system (e.g. see above). Resource access diverter 804 further modifies those Downloadable-API IAT entries that correspond with protection policies 342, thereby causing corresponding Downloadable accesses via Downloadable-API IAT 731 to be diverted resource access analyzer 805.

During Downloadable operation, resource access analyzer or "RAA" 805 receives and determines a response to diverted Downloadable (i.e. "malicious") operations in accordance with corresponding protection policies of policies 342. (RAA 805 or further elements, which are not shown, can further similarly provide for other security mechanisms that might also be implemented.) Malicious operations can for example include, in a Windows environment: file operations (e.g. reading, writing, deleting or renaming a file), network operations (e.g. listen on or connect to a socket, send/receive data or view intranet), OS registry or similar operations (read/write a registry item), OS operations (exit OS/client, kill or change the priority of a process/thread, dynamically load a class library), resource usage thresholds (e.g. memory, CPU, graphics), etc.

Policy enforcer 806 receives RAA 805 results and causes a corresponding response to be implemented, again according to the corresponding policies. Policy enforcer 806 can, for example, interact with a user (e.g. provide an alert, receive instructions, etc.), create a log file, respond, cause a response to be transferred to the Downloadable using "dummy" or limited data, communicate with a server or other networked device (e.g. corresponding to a local or remote administrator), respond more specifically with a better known Downloadable, verify accessibility or user/system information (e.g. via local or remote information), even enable the attempted Downloadable access, among a wide variety of responses that will become apparent in view of the teachings herein.

The FIG. 9 flowchart illustrates a protection method according to an embodiment of the invention. In step 901, a protection engine monitors the receipt, by a server or other re-communicator of information, and receives such information intended for a

protected information-destination (i.e. a potential-Downloadable) in step 903. Steps 905-911 depict an adjunct trustworthiness protection that can also be provided, wherein the protection engine determines whether the source of the received information is known to be "unfriendly" and, if so, prevents current (at least unaltered) delivery of the potential-Downloadable and provides any suitable alerts. (The protection engine might also continue to perform Downloadable detection and nevertheless enable delivery or protected delivery of a non-Downloadable, or avoid detection if the source is found to be "trusted", among other alternatives enabled by the teachings herein.)

If, in step 913, the potential-Downloadable source is found to be of an unknown or otherwise suitably authenticated/certified source, then the protection engine determines whether the potential-Downloadable includes executable code in step 915. If the potential-Downloadable does not include executable code, then the protection engine causes the potential-Downloadable to be delivered to the information-destination in its original form in step 917, and the method ends. If instead the potential-Downloadable is found to include executable code in step 915 (and is thus a "detected-Downloadable"), then the protection engine forms a sandboxed package in step 919 and causes the protection agent to be delivered to the information-Destination in step 921, and the method ends. As was discussed earlier, a suitable protection agent can include mobile protection code, policies and the detected-Downloadable (or information corresponding thereto).

The FIG. 10a flowchart illustrates a method for analyzing a potential-Downloadable, according to an embodiment of the invention. As shown, one or more aspects can provide useful indicators of the inclusion of executable code within the

potential-Downloadable. In step 1001, the protection engine determines whether the potential-Downloadable indicates an executable file type, for example, by comparing one or more included file headers for file type indicators (e.g. extensions or other descriptors). The indicators can be compared against all known file types executable by all protected Downloadable destinations, a subset, in accordance with file types executable or desirably executable by the Downloadable-destination, in conjunction with a particular user, in conjunction with available information or operability at the destination, various combinations, etc.

Where content analysis is conducted, in step 1003 of FIG. 10a, the protection engine analyzes the potential-Downloadable and determines in accordance therewith whether the potential-Downloadable does or is likely to include binary information, which typically indicates executable code. The protection engine further analyzes the potential-Downloadable for patterns indicative of included executable code in step 1003. Finally, in step 1005, the protection engine determines whether the results of steps 1001 and 1003 indicate that the potential-Downloadable more likely includes executable code (e.g. via weighted comparison of the results with a suitable level indicating the inclusion or exclusion of executable code). The protection engine, given a suitably high confidence indicator of the inclusion of executable code, treats the potential-Downloadable as a detected-Downloadable.

The FIG. 10b flowchart illustrates a method for forming a sandboxed package according to an embodiment of the invention. As shown, in step 1011, a protection engine retrieves protection parameters and forms mobile protection code according to the parameters. The protection engine further, in step 1013, retrieves protection parameters

and forms protection policies according to the parameters. Finally, in step 1015, the protection engine couples the mobile protection code, protection policies and received-information to form a sandboxed package. For example, where a Downloadable-destination utilizes a standard windows executable, coupling can further be accomplished via concatenating the MPC for delivery of MPC first, policies second, and received information third. (The protection parameters can, for example, include parameters relating to one or more of the Downloadable destination device/process, user, supervisory constraints or other parameters.)

The FIG. 11 flowchart illustrates how a protection method performed by mobile protection code ("MPC") according to an embodiment of the invention includes the MPC installing MPC elements and policies within a destination device in step 1101. In step 1102, the MPC loads the Downloadable without actually initiating it (i.e. for executables, it will start a process in suspended mode). The MPC further forms an access monitor or "interceptor" for monitoring or "intercepting" downloadable destination device access attempts within the destination device (according to the protection policies in step 1103, and initiates a corresponding Downloadable within the destination device in step 1105.

If, in step 1107, the MPC determines, from monitored/intercepted information, that the Downloadable is attempting or has attempted a destination device access considered undesirable or otherwise malicious, then the MPC performs steps 1109 and 1111; otherwise the MPC returns to step 1107. In step 1109, the MPC determines protection policies in accordance with the access attempt by the Downloadable, and in step 1111, the MPC executes the protection policies. (Protection policies can, for example, be retrieved from a temporary, e.g. memory/cache, or more persistent storage.)

As shown in the FIG. 12a example, the MPC can provide for intercepting Downloadable access attempts by a Downloadable by installing the Downloadable (but not executing it) in step 1201. Such installation will cause a Downloadable executor, such as a the Windows operating system, to provide all required interfaces and parameters (such as the IAT, process ID, etc.) for use by the Downloadable to access device resources of the host device. The MPC can thus cause Downloadable access attempts to be diverted to the MPC by modifying the Downloadable IAT, replacing device resource location indicators with those of the MPC (step 1203).

The FIG. 12b example further illustrates an example of how the MPC can apply suitable policies in accordance with an access attempt by a Downloadable. As shown, the MPC receives the Downloadable access request via the modified IAT in step 1211. The MPC further queries stored policies to determine a policy corresponding to the Downloadable access request in step 1213.

The foregoing description of preferred embodiments of the invention is provided by way of example to enable a person skilled in the art to make and use the invention, and in the context of particular applications and requirements thereof. Various modifications to the embodiments will be readily apparent to those skilled in the art, and the generic principles defined herein may be applied to other embodiments and applications without departing from the spirit and scope of the invention. Thus, the present invention is not intended to be limited to the embodiments shown, but is to be accorded the widest scope consistent with the principles, features and teachings disclosed herein. The embodiments described herein are not intended to be exhaustive or limiting. The present invention is limited only by the following claims.

WHAT IS CLAIMED IS:

1. A method, comprising:

receiving downloadable-information;

determining whether the downloadable-information includes executable code; and

causing mobile protection code to be communicated to at least one information-destination of the downloadable-information, if the downloadable-information is determined to include executable code.

2. The method of claim 1, wherein the receiving includes monitoring received information of an information re-communicator.

3. The method of claim 2, wherein the information re-communicator is a network server.

4. The method of claim 1, wherein the determining comprises analyzing the downloadable-information for an included type indicator indicating an executable file type.

5. The method of claim 1, wherein the determining comprises analyzing the downloadable-information for an included an included type detector indicating an archive file that contains at least one executable.

6. The method of claim 1, wherein the determining comprises analyzing the downloadable-information for an included file type indicator and an information pattern

corresponding to one or more information patterns that tend to be included within executable code.

7. The method of claim 1, further comprising receiving one or more executable code characteristics of executable code that is capable of being executed by the information-destination, and wherein the determining is conducted in accordance with the executable code characteristics.

8. The method of claim 1, wherein the determining comprises performing one or more analyses of the downloadable-information, the analyses producing detection-indicators indicating whether a correspondence is detected between a downloadable-information characteristic and at least one respective executable code characteristic, and evaluating the detection-indicators to determine whether the downloadable-information includes executable code.

9. The method of claim 8, wherein at least one of the detection-indicators indicates a level of downloadable-information characteristic and executable code characteristic correspondence.

10. The method of claim 8, wherein the evaluating includes assigning a weighted level of importance to at least one of the indicators.

11. The method of claim 1, wherein the causing mobile protection code to be

communicated comprises forming a sandboxed package including the mobile protection code and the downloadable-information, and causing the sandboxed package to be communicated to the at least one information-destination.

12. The method of claim 10, wherein the sandboxed package is formed such that the mobile protection code will be executed by the information-destination before the downloadable-information.

13. The method of claim 11, wherein the sandboxed package further includes protection policies according to which the mobile protection code is operable.

14. The method of claim 13, wherein the sandboxed package is formed for receipt by the information-destination such that the mobile protection code is received before the downloadable-information, and the downloadable information before the protection policies.

15. The method of claim 13, wherein the protection policies correspond with at least one of the information-destination and a user of the information destination.

16. A system, comprising:

an information monitor for receiving downloadable-information;

a content inspection engine communicatively coupled to the information monitor for determining whether the downloadable-information includes executable code; and

a protection agent engine communicatively coupled to the content inspection engine for causing mobile protection code ("MPC") to be communicated to at least one information-destination of the downloadable-information, if the downloadable-information is determined to include executable code.

17. The system of claim 16, wherein the information monitor intercepts received information received by an information re-communicator.

18. The system of claim 17, wherein the information re-communicator is a network server.

19. The system of claim 16, wherein the content inspection engine comprises a file type detector for determining whether the downloadable-information includes a file type indicator indicating an executable file type.

20. The system of claim 16, wherein the content inspection engine comprises a parser for parsing the downloadable-information and a content analyzer communicatively coupled to the parser for determining whether one or more downloadable-information elements of the downloadable-information correspond with executable code elements are executable code elements.

21. The system of claim 16, wherein the content inspection engine comprises one or more downloadable-information analyzers for analyzing the downloadable-information,

each analyzer producing therefrom a detection indicator indicating whether a downloadable-information characteristic corresponds with an executable code characteristic, and an inspection controller communicatively coupled to the analyzers for determining whether the indicators indicate that the downloadable-information includes executable code.

22. The system of claim 21, wherein at least one of the detection-indicators indicates a level of downloadable-information characteristic and executable code characteristic correspondence.

23. The system of claim 21, wherein the evaluating includes assigning a weighted level of importance to at least one of the detection-indicators.

24. The system of claim 16, wherein the sandboxed package engine comprises an MPC generator for providing the MPC, a linking engine coupled to the MPC generator for forming a protection agent including the MPC and the downloadable-information, and a transfer engine for causing the protection agent to be communicated to the at least one information-destination.

25. The system of claim 24, wherein the protection agent engine further comprises a policy generator communicatively coupled to the linking engine for providing protection policies according to which the MPC is operable.

26. The system of claim 25, wherein the sandboxed package is formed for receipt by the information-destination such that the mobile protection code is executed before the downloadable-information.

27. The system of claim 26, wherein the protection policies correspond with policies of at least one of the information-destination and a user of the information destination.

28. A system, comprising:

means for receiving downloadable-information;

means for determining whether the downloadable-information includes executable code; and

means for causing mobile protection code to be communicated to at least one information-destination of the downloadable-information, if the downloadable-information is determined to include executable code.

29. A computer-readable storage medium storing program code for causing a computer to perform the steps of:

receiving downloadable-information;

determining whether the downloadable-information includes executable code; and

causing mobile protection code to be communicated to at least one information-destination of the downloadable-information, if the downloadable-information is determined to include executable code.

30. A method, comprising:

receiving, at an information re-communicator, downloadable-information, including executable code; and

causing mobile protection code to be executed by a mobile code executor at a downloadable-information destination such that one or more operations of the executable code at the destination, if attempted, will be processed by the mobile protection code.

31. The method of claim 30, wherein the mobile code executor is a Java Virtual Machine.

32. The method of claim 30, wherein the mobile code executor is the operating system, running native code executables.

33. The method of claim 30, wherein the mobile code executor is ActiveX subsystem of the windows operating system

34. The method of claim 30, wherein the mobile code executor is the Microsoft Windows scripting host

35. The method of claim 30, wherein the causing is accomplished by forming a sandboxed package including the mobile protection code and the downloadable-information, and causing the sandboxed package to be delivered to the downloadable-information destination.

36. The method of claim 35, wherein the sandboxed package further includes protection policies according to which the processing by the mobile protection code is conducted.

37. A sandboxed package formed according to the method of claim 35.

38. A sandboxed package formed according to the method of claim 36.

39. The method of claim 36, wherein the forming comprises generating the mobile protection code, generating the sandboxed package, and linking the mobile protection code, protection policies and downloadable-information.

40. The method of claim 39, wherein the generating of at least one of the mobile protection code and the protection policies is conducted in accordance with one or more destination-characteristics of the destination.

41. The method of claim 40, wherein the destination-characteristics include characteristics corresponding to at least one of a destination user, a destination device and a destination process.

42. The method of claim 35, wherein the causing the sandboxed package to be executed includes communicating the sandboxed package to a communication buffer of the information re-communicator.

43. The method of claim 30, wherein the re-communicator is at least one of a firewall and a network server.

44. The method of claim 30, wherein the sandboxed package has a same file type as the downloadable-information, thereby causing the mobile code executor to be unaware that the protected package is not a normal downloadable.

45. The method of claim 44, wherein the sandboxed package is formed using concatenation of a mobile protection code, a policy, and a downloadable.

46. The method of claim 30, wherein executing the mobile protection code at the destination causes downloadable interfaces to resources at the destination to be modified such that at least one attempted operation of the executable code is diverted to the mobile protection code.

47. A system, comprising:

receiving means for receiving, at an information re-communicator, downloadable-information, including executable code; and

mobile code means communicatively coupled to the receiving means for causing mobile protection code to be executed by a mobile code executor at a downloadable-information destination such that one or more operations of the executable code at the destination, if attempted, will be processed by the mobile protection code.

48. The system of claim 47, wherein the mobile code executor is a Java Virtual Machine.

49. The system of claim 47, wherein the mobile code executor is an operating system, running native code executables.

50. The system of claim 47, wherein the mobile code executor is an ActiveX subsystem of the windows operating system.

51. The system of claim 47, wherein the mobile code executor is a Microsoft Windows scripting host.

52. The system of claim 47, wherein the causing is accomplished by forming a sandboxed package including the mobile protection code and the downloadable-information, and causing the sandboxed package to be delivered to the downloadable-information destination.

53. The system of claim 52, wherein the sandboxed package further includes protection policies according to which the processing by the mobile protection code is conducted.

54. The system of claim 53, wherein the forming comprises generating the mobile protection code, generating the protection policies, and linking the mobile protection code, protection policies and downloadable-information.

55. The system of claim 54, wherein the generating of at least one of the mobile protection code and the protection policies is conducted in accordance with one or more destination-characteristics of the destination.

56. The system of claim 55, wherein the destination-characteristics include characteristics corresponding to at least one of a destination user, a destination device and a destination process.

57. The system of claim 46, wherein the causing the sandboxed package to be executed includes communicating the sandboxed package to a communication buffer of the information re-communicator.

58. The system of claim 47, wherein the re-communicator is at least one of a firewall and a network server.

59. The system of claim 47, wherein executing the mobile protection code at the destination causes downloadable interfaces a resource at the destination to be modified such that at least one attempted operation of the executable code is diverted to the mobile protection code.

60. A computer-readable storage medium storing program code for causing a computer to perform the steps of:

receiving, at an information re-communicator, downloadable-information, including executable code; and

causing mobile protection code to be executed by a mobile code executor at a downloadable-information destination such that one or more operations of the executable code at the destination, if attempted, will be processed by the mobile protection code.

61. A method, comprising:

receiving mobile protection code ("MPC") and a Downloadable at a Downloadable-destination;

causing, by the MPC, one or more operations attempted by the Downloadable to be received by the MPC;

receiving, by the MPC, an attempted operation of the Downloadable; and

initiating, by the MPC, a protection policy corresponding to the attempted operation.

62. The method of claim 61, wherein the receiving comprises receiving a sandboxed package that includes the MPC, the Downloadable and one or more protection policies.

63. The method of claim 62, wherein the sandboxed package is configured such that the MPC is executed first, the Downloadable is executed by the MPC and the protection policies are accessible to the MPC.

64. The method of claim 61, wherein the causing comprises modifying, by the MPC,

interfaces of a corresponding downloadable to resources at the destination.

65. The method of claim 64, wherein the modifying is accomplished by initiating a loading of the Downloadable, thereby causing a mobile code executor to provide and initialize the interfaces, modifying one or more interface elements to divert corresponding attempted Downloadable operations to the MPC, and initiating execution of the Downloadable.

66. The method of claim 64, wherein the interfaces comprise an import address table ("IAT") of a native code executable downloadable.

67. The method of claim 64, wherein modifying the interfaces installs a filter-driver between the downloadable and the resources.

68. A system, comprising:

a mobile code executer for initiating received mobile code; and

a sandboxed package capable of being received and initiated by the mobile code executer, the sandboxed package including a Downloadable and mobile protection code ("MPC") for causing one or more Downloadable operations to be intercepted and for processing the intercepted operations, if the Downloadable attempts to initiate the operations.

69. The system of claim 60, wherein the MPC comprises:

an MPC installer for causing MPC elements to be installed;

a Downloadable installer communicatively coupled to the MPC element installer for installing the Downloadable;

a resource access diverter communicatively coupled to the MPC installer for causing the Downloadable operations to be intercepted;

a resource access analyzer communicatively coupled to the MPC installer for receiving an intercepted Downloadable operation and determining a protection policy corresponding to the intercepted Downloadable operation; and

a policy enforcer communicatively coupled to the resource access analyzer for processing the intercepted Downloadable operation.

70. The system of claim 69, wherein the resource access diverter modifies one or more elements of an interface usable by the Downloadable to effectuate the Downloadable operations.

71. The system of claim 69, wherein the mobile code executer is a Java Virtual Machine.

72. The system of claim 69, wherein the mobile code executor is an operating system, running native code executables.

73. The system of claim 69, wherein the mobile code executor is an ActiveX subsystem of the windows operating system.

74. The system of claim 69, wherein the mobile code executor is an Microsoft Windows scripting host.

75. A system, comprising

receiving means for receiving mobile protection code ("MPC") and a Downloadable at a Downloadable-destination;

monitoring means for causing, by the MPC, one or more operations attempted by the Downloadable to be received by the MPC;

second receiving means receiving, by the MPC, an attempted operation of the Downloadable; and

initiating means for initiating, by the MPC, a protection policy corresponding to the attempted operation.

76. A computer-readable storage medium storing program code for causing a computer to perform the steps of:

receiving mobile protection code ("MPC") and a Downloadable at a Downloadable-destination;

causing, by the MPC, one or more operations attempted by the Downloadable to be received by the MPC;

receiving, by the MPC, an attempted operation of the Downloadable; and

initiating, by the MPC, a protection policy corresponding to the attempted operation.

ABSTRACT OF THE DISCLOSURE

MALICIOUS MOBILE CODE RUNTIME MONITORING SYSTEM AND METHODS

Protection systems and methods provide for protecting one or more personal computers ("PCs") and/or other intermittently or persistently network accessible devices or processes from undesirable or otherwise malicious operations of Java™ applets, ActiveX™ controls, JavaScript™ scripts, Visual Basic scripts, add-ins, downloaded/ uploaded programs or other "Downloadables" or "mobile code" in whole or part. A protection engine embodiment provides, within a server, firewall or other suitable "re-communicator," for monitoring information received by the communicator, determining whether received information does or is likely to include executable code, and if so, causes mobile protection code (MPC) to be transferred to and rendered operable within a destination device of the received information, more suitably by forming a protection agent including the MPC, protection policies and a detected-Downloadable. An MPC embodiment further provides, within a Downloadable-destination, for initiating the Downloadable, enabling malicious Downloadable operation attempts to be received by the MPC, and causing (predetermined) corresponding operations to be executed in response to the attempts, more suitably in conjunction with protection policies.

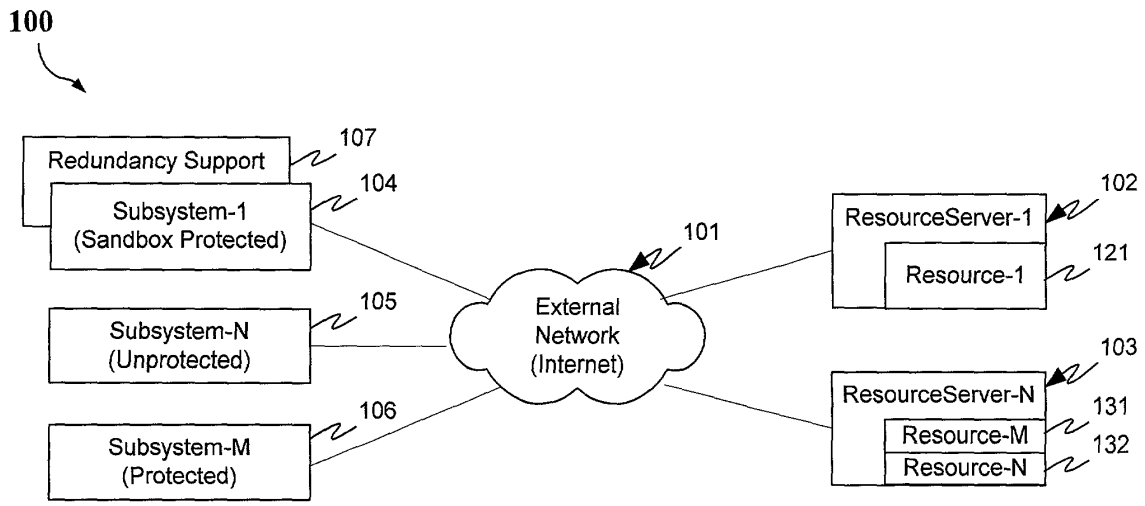

FIG. 1a

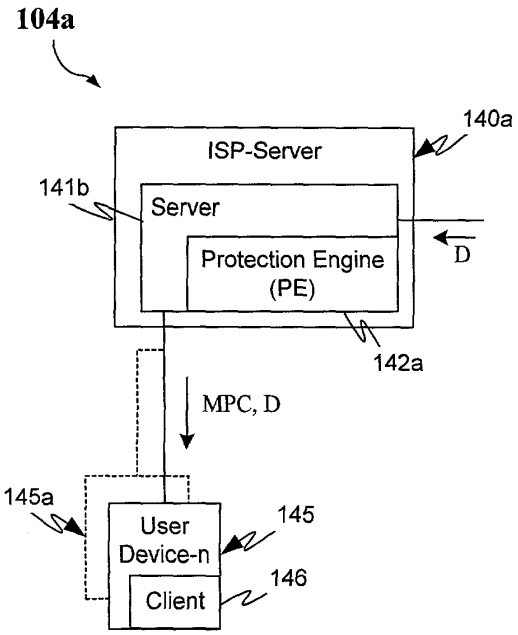

FIG. 1b

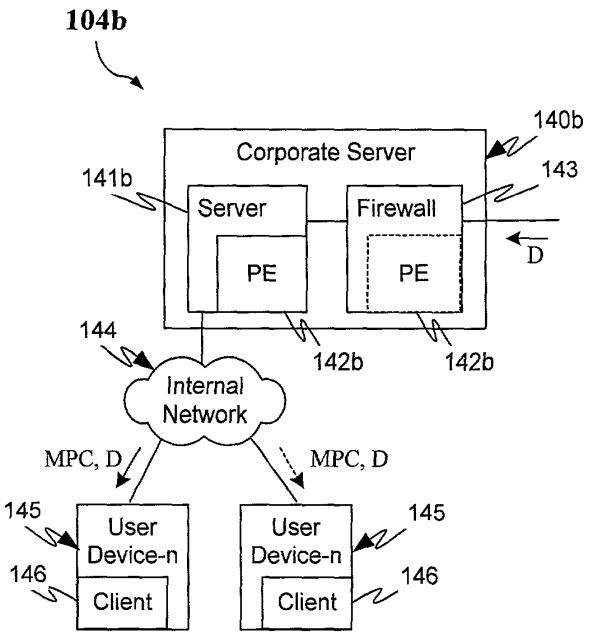

FIG. 1c

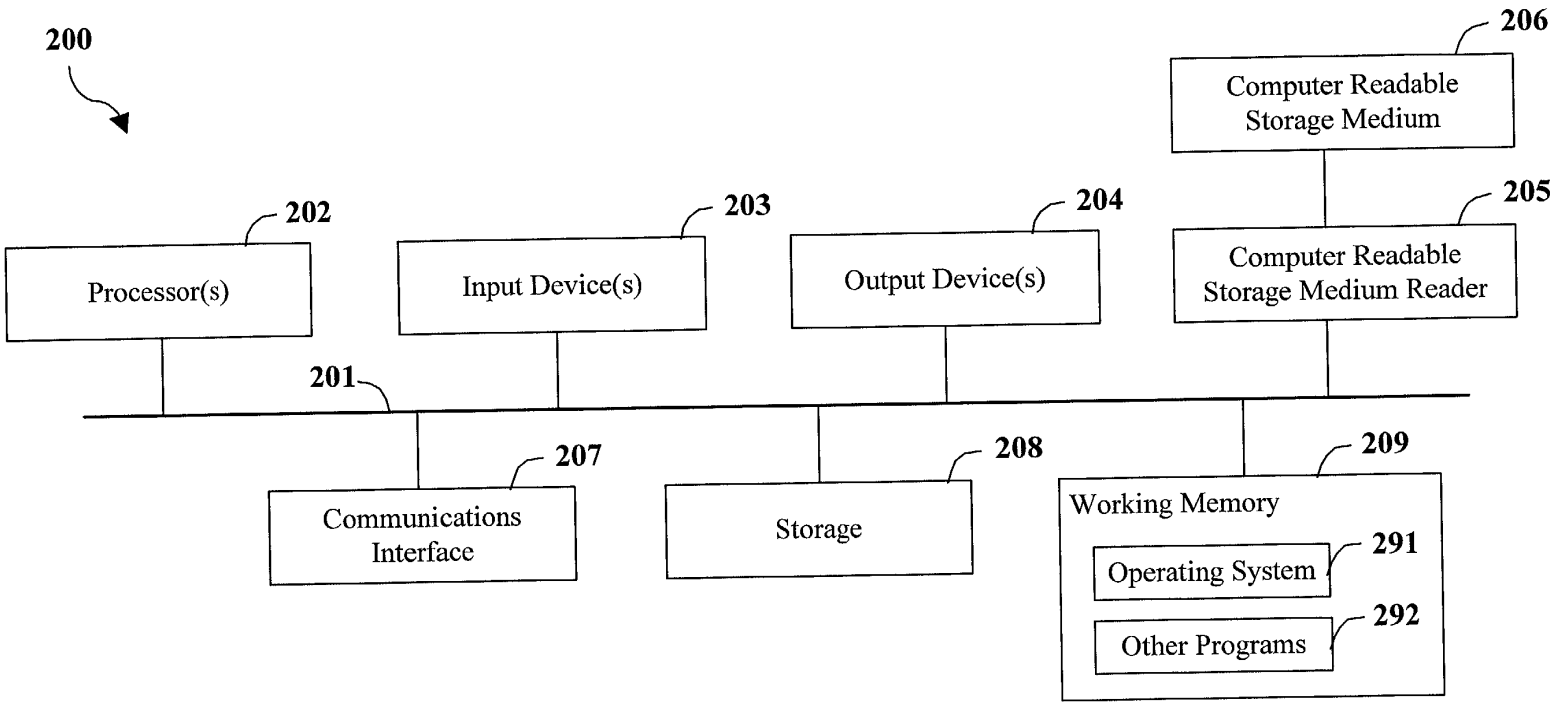
200
201
202
Processor(s)
203
Input Device(s)
204
Output Device(s)
205
Computer Readable Storage Medium Reader
206
Computer Readable Storage Medium
207
Communications Interface
208
Storage
209
Working Memory
291
Operating System
292
Other Programs

FIG. 2

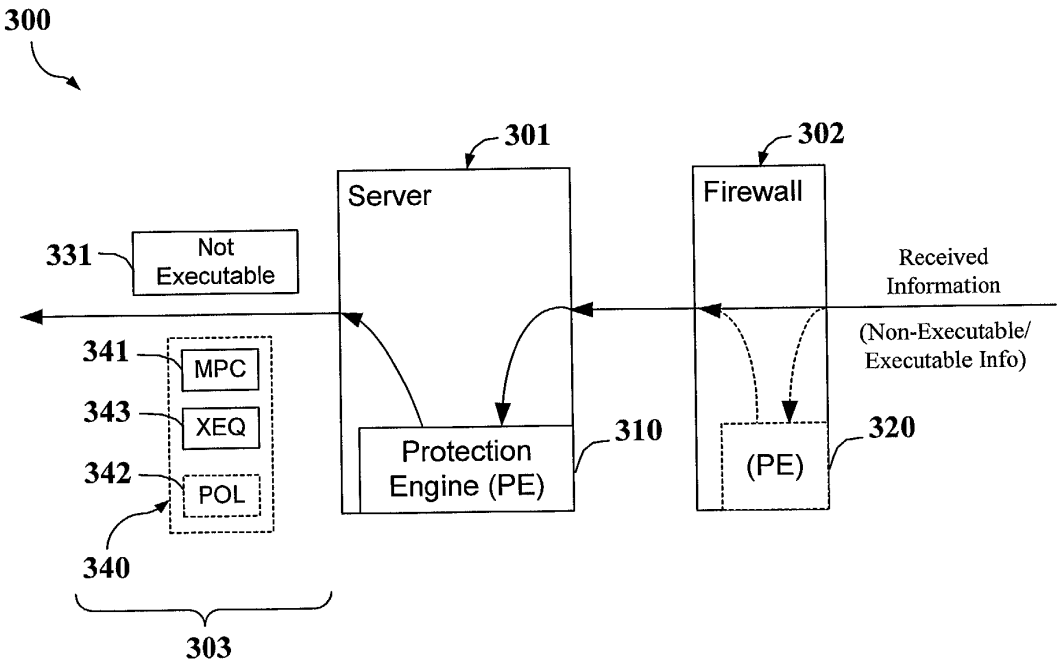
300
301
302
Server
Firewall
331
Not Executable
Received Information
(Non-Executable/ Executable Info)
341
MPC
343
XEQ
342
POL
340
303
Protection Engine (PE)
310
(PE)
320

FIG. 3

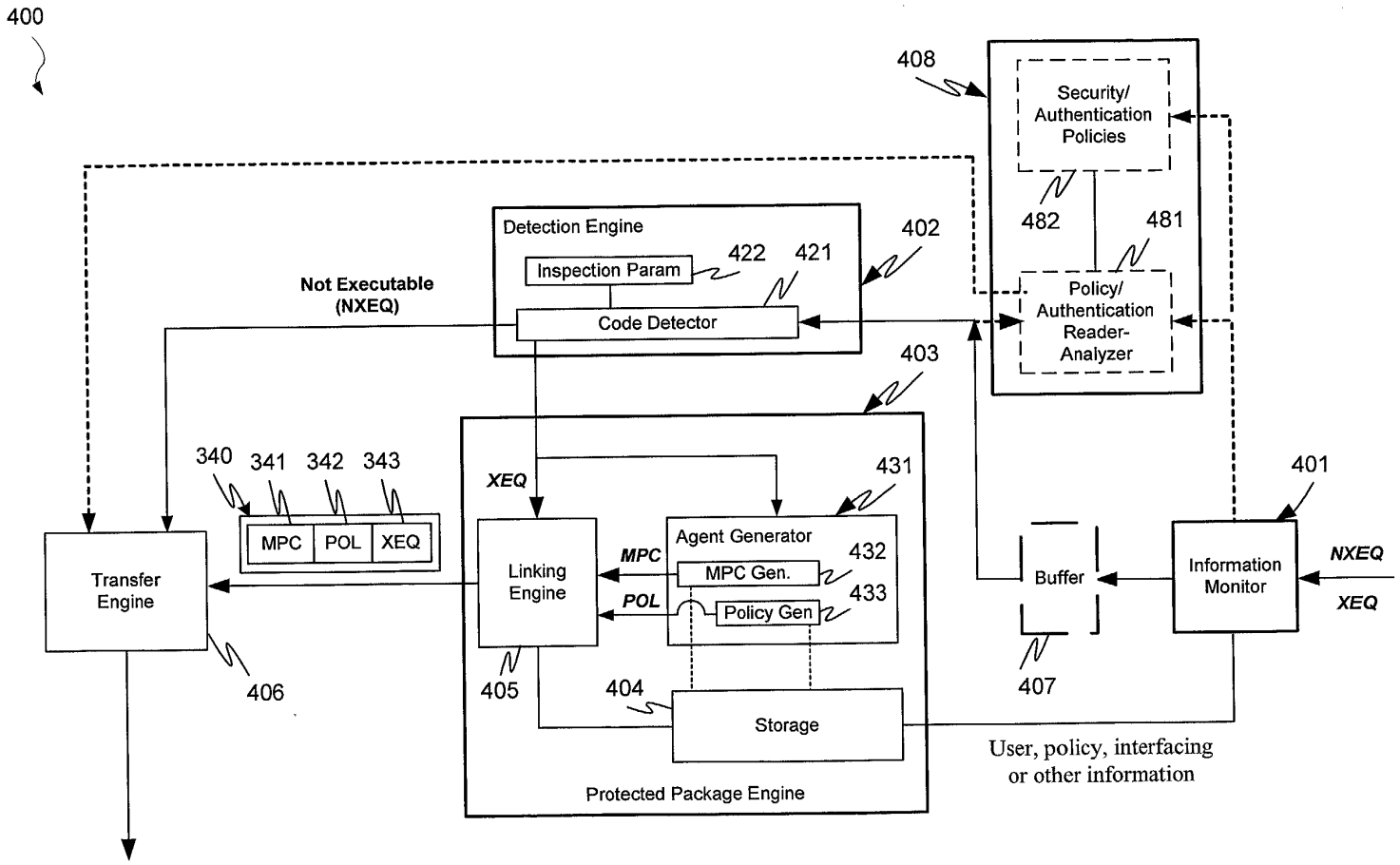
400
408
Security/ Authentication Policies
482
481
Policy/ Authentication Reader-Analyzer
Detection Engine
Inspection Param
422
421
402
Code Detector
Not Executable (NXEQ)
403
340
341
342
343
MPC
POL
XEQ
XEQ
431
Agent Generator
MPC Gen.
432
Policy Gen
433
MPC
POL
Linking Engine
Transfer Engine
406
405
404
Storage
Protected Package Engine
Buffer
407
Information Monitor
401
NXEQ
XEQ
User, policy, interfacing or other information

FIG. 4

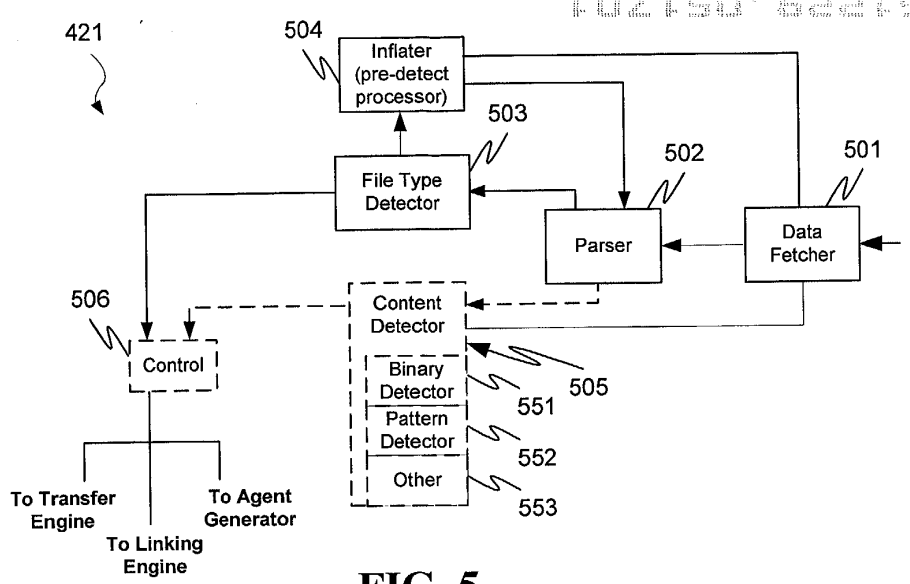

FIG. 5

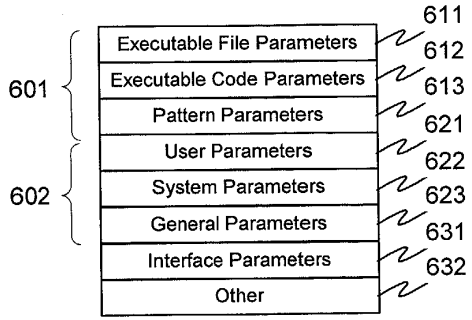

FIG. 6a

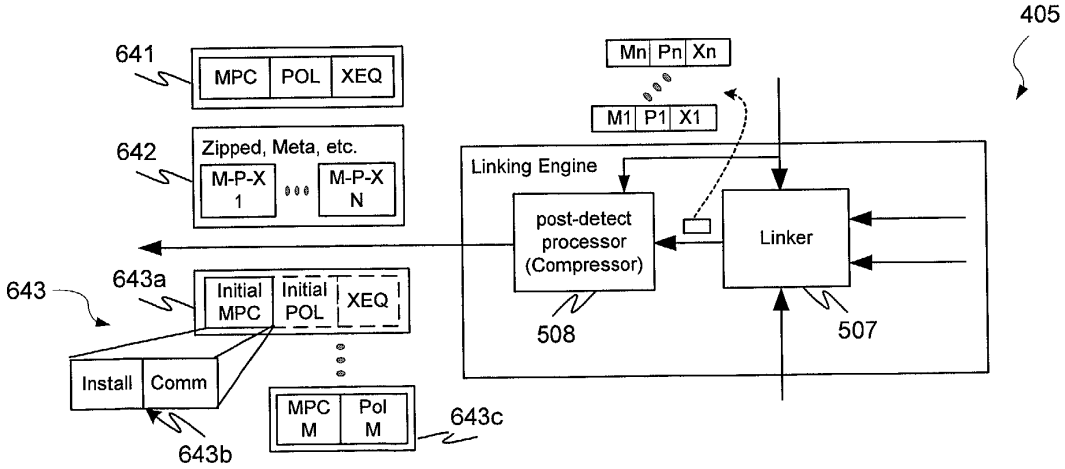

FIG. 6b

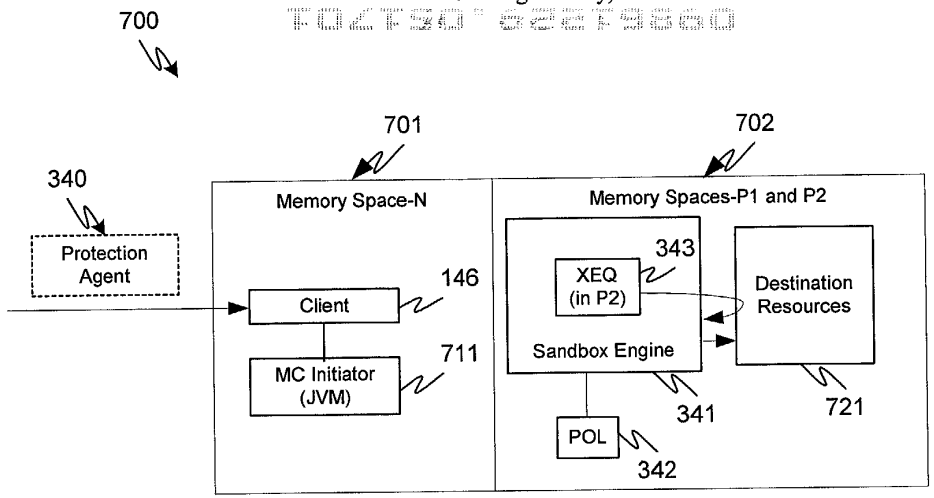

FIG. 7a

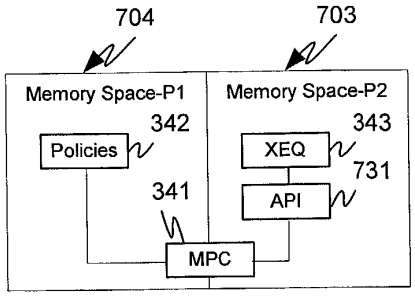

FIG. 7b

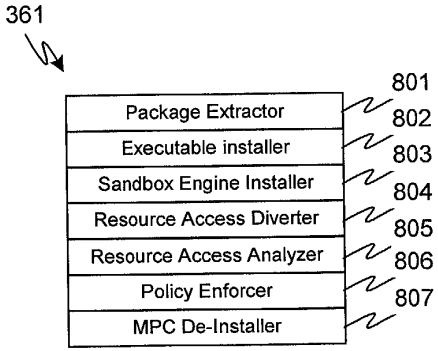

FIG. 8

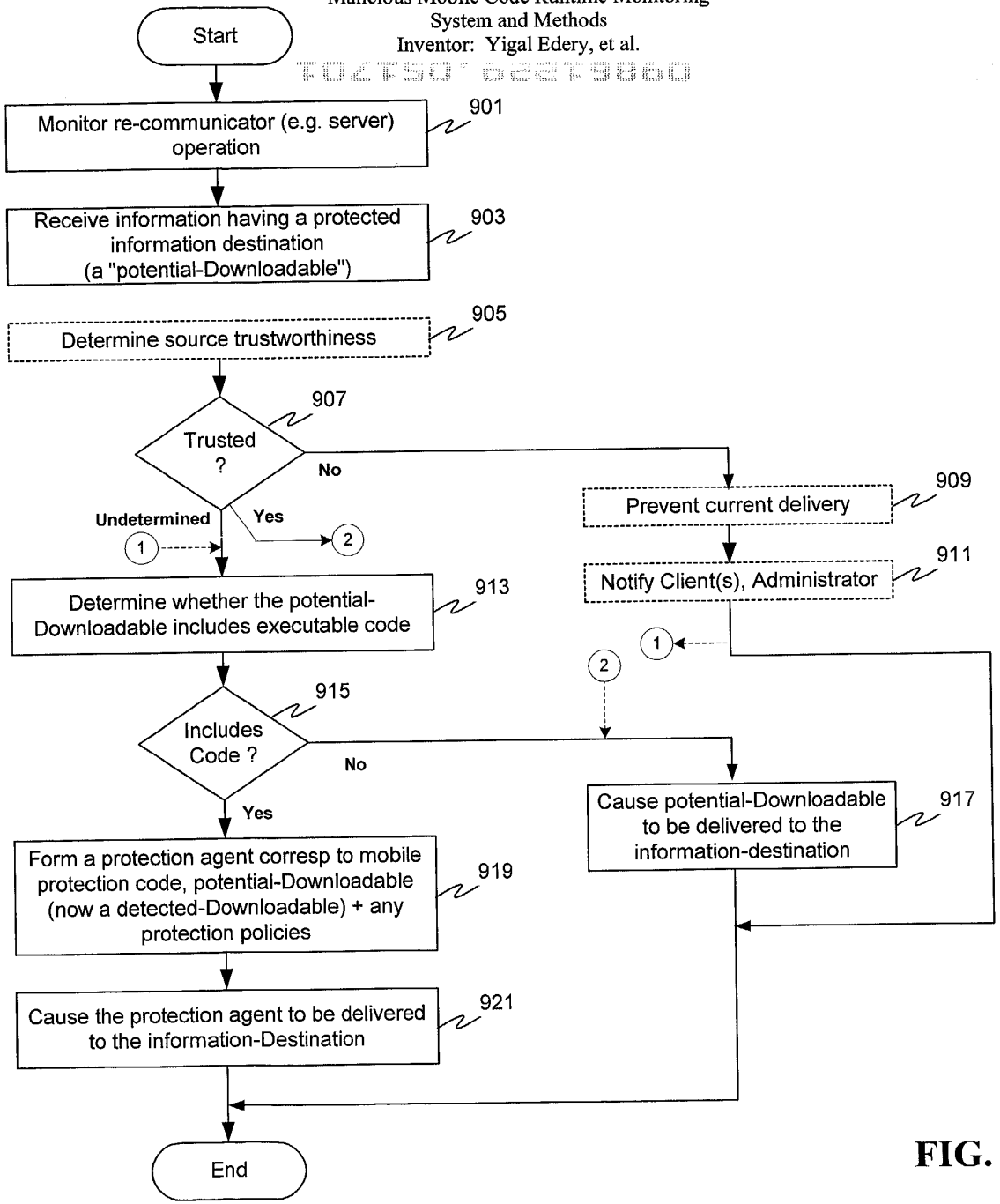
Start
Monitor re-communicator (e.g. server) operation
901
Receive information having a protected information destination (a "potential-Downloadable")
903
Determine source trustworthiness
905
Trusted ?
907
No
Yes
Undetermined
1
2
Prevent current delivery
909
Notify Client(s), Administrator
911
Determine whether the potential-Downloadable includes executable code
913
Includes Code ?
915
No
Yes
Cause potential-Downloadable to be delivered to the information-destination
917
Form a protection agent corresp to mobile protection code, potential-Downloadable (now a detected-Downloadable) + any protection policies
919
Cause the protection agent to be delivered to the information-Destination
921
End

FIG. 9

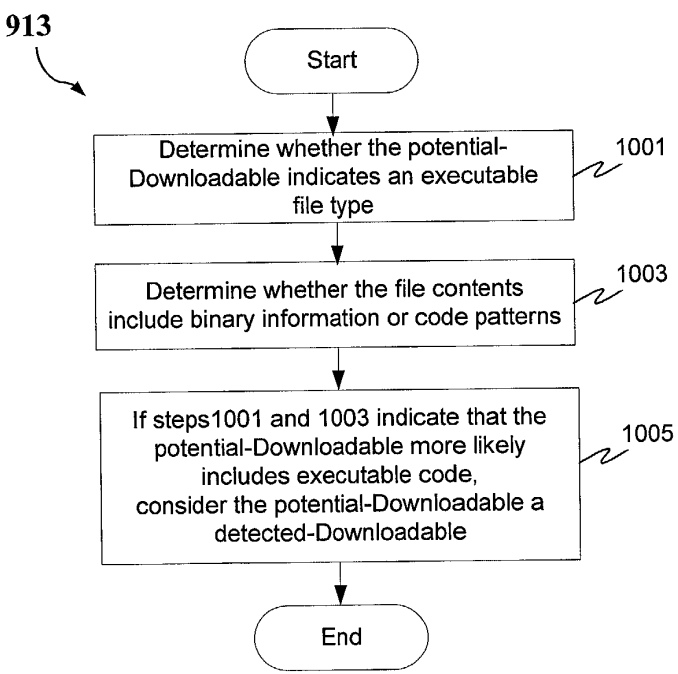

FIG. 10A

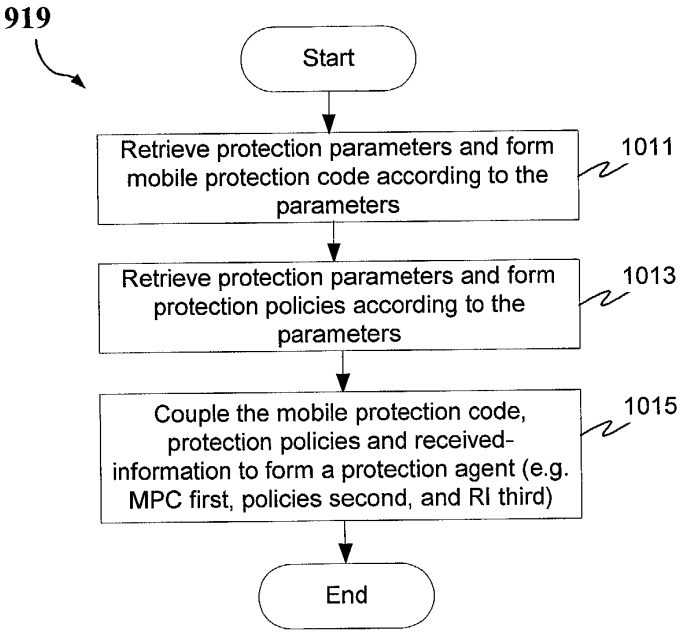

FIG. 10B

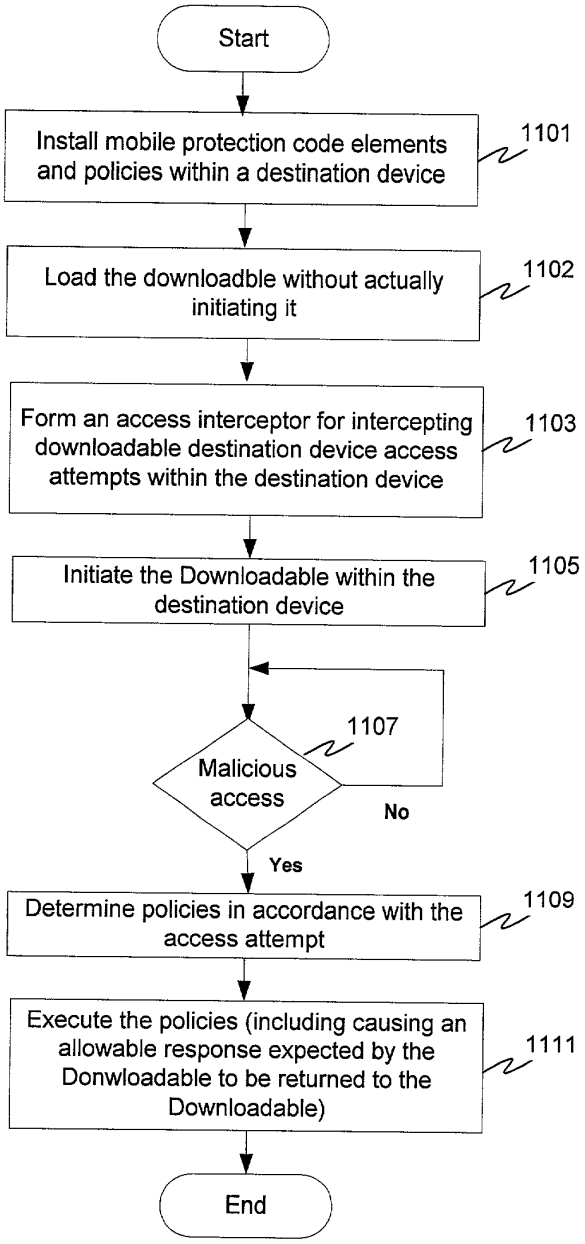
Start
Install mobile protection code elements and policies within a destination device
1101
Load the downloadble without actually initiating it
1102
Form an access interceptor for intercepting downloadable destination device access attempts within the destination device
1103
Initiate the Downloadable within the destination device
1105
Malicious access
1107
No
Yes
Determine policies in accordance with the access attempt
1109
Execute the policies (including causing an allowable response expected by the Donwloadable to be returned to the Downloadable)
1111
End

FIG. 11

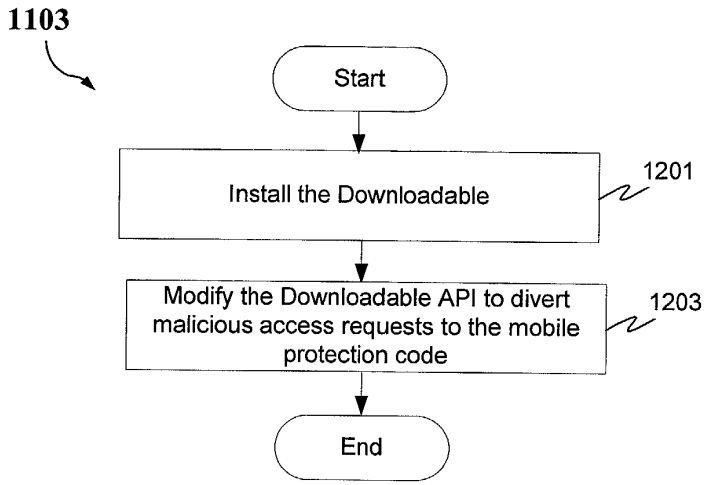

FIG. 12a

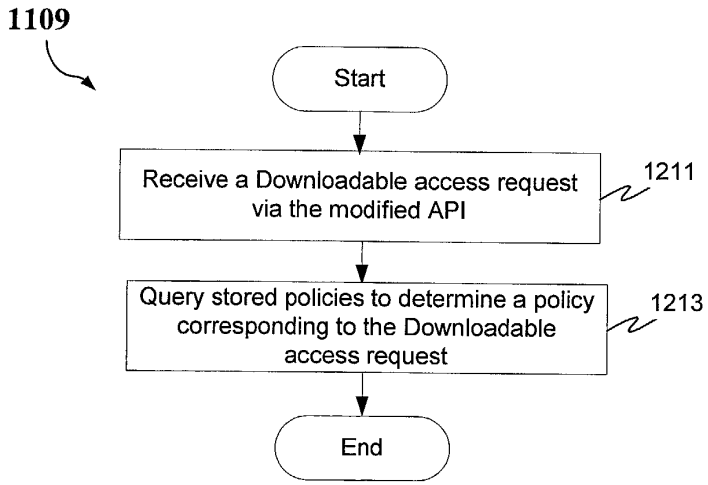

FIG. 12b

Class | ISSUE CLASSIFICATION

U.S. UTILITY Patent Application

O.I.P.E.	PATENT DATE
SCANNED GTH Q.A. A.G.	

APPLICATION NO.	CONT/PRIOR	CLASS	SUBCLASS	ART UNIT	EXAMINER
09/861229	D	~~709~~ 713	201	~~2152~~ 2131	Rewalt

APPLICANTS
Yigal Edery
Nimrod Vered
David Kroll

TITLE
Malicious mobile code runtime monitoring system and methods

PTO-2040
12/99

ISSUING CLASSIFICATION

ORIGINAL		CROSS REFERENCE(S)						
CLASS	SUBCLASS	CLASS	SUBCLASS (ONE SUBCLASS PER BLOCK)					
INTERNATIONAL CLASSIFICATION								
			☐ Continued on Issue Slip Inside File Jacket					

☐ TERMINAL DISCLAIMER	DRAWINGS			CLAIMS ALLOWED	
	Sheets Drwg.	Figs. Drwg.	Print Fig.	Total Claims	Print Claim for O.G.
☐ The term of this patent subsequent to _____ (date) has been disclaimed.	(Assistant Examiner) (Date)			NOTICE OF ALLOWANCE MAILED	
☐ The term of this patent shall not extend beyond the expiration date of U.S Patent. No. _____	(Primary Examiner) (Date)			ISSUE FEE	
				Amount Due	Date Paid
☐ The terminal ____months of this patent have been disclaimed.	(Legal Instruments Examiner) (Date)			ISSUE BATCH NUMBER	

WARNING:
The information disclosed herein may be restricted. Unauthorized disclosure may be prohibited by the United States Code Title 35, Sections 122, 181 and 368. Possession outside the U.S. Patent & Trademark Office is restricted to authorized employees and contractors only.

Form PTO-436A
(Rev. 6/99)

FILED WITH: ☐ DISK (CRF) ☐ FICHE ☐ CD-ROM
(Attached in pocket on right inside flap)

(FACE)

SEARCHED

Class	Sub.	Date	Exmr.
713	176, 175, 200, 201	12/4/04	CM
709	223-229		
717	123, 124, 126, 127, 130, 131, 134, 135		

INTERFERENCE SEARCHED

Class	Sub.	Date	Exmr.

SEARCH NOTES
(INCLUDING SEARCH STRATEGY)

	Date	Exmr.
BRS Text Search USPAT, DERWENT, JPO, EPO, IBM TDB, USPGPub, USOCR	12/4/04	CM
DIALOG COMPSCI, ELECTRONIC, SOFTWARE		
PALM Inventor Name Search		

(RIGHT OUTSIDE)

ISSUE SLIP STAPLE AREA (for additional cross references)

POSITION	INITIALS	ID NO.	DATE
FEE DETERMINATION	[illegible]		5/22/01
O.I.P.E. CLASSIFIER		420	6/6
FORMALITY REVIEW	[illegible]		07-19-01
RESPONSE FORMALITY REVIEW	Zm	927	10-19-01

INDEX OF CLAIMS

✓ ..... Rejected
= ..... Allowed
— (Through numeral)... Canceled
÷ ..... Restricted
N ..... Non-elected
I ..... Interference
A ..... Appeal
O ..... Objected

Claim Final	Claim Original	Date 12/5/01
	1	✓
	2	✓
	3	✓
	4	✓
	5	✓
	6	✓
	7	✓
	8	O
	9	O
	10	O
	11	O
	12	O
	13	O
	14	O
	15	O
	16	✓
	17	✓
	18	✓
	19	✓
	20	✓
	21	O
	22	O
	23	O
	24	O
	25	O
	26	O
	27	O
	28	✓
	29	✓
	30	✓
	31	✓
	32	✓
	33	✓
	34	✓
	35	O
	36	O
	37	O
	38	O
	39	O
	40	O
	41	O
	42	O
	43	✓
	44	✓
	45	✓
	46	✓
	47	✓
	48	✓
	49	✓
	50	✓

Claim Final	Claim Original	Date 12/5/01
	51	✓
	52	O
	53	O
	54	O
	55	O
	56	O
	57	O
	58	O
	59	O
	60	✓
	61	✓
	62	✓
	63	✓
	64	✓
	65	✓
	66	✓
	67	✓
	68	✓
	69	✓
	70	✓
	71	✓
	72	✓
	73	✓
	74	✓
	75	✓
	76	✓
	77	
	78	
	79	
	80	
	81	
	82	
	83	
	84	
	85	
	86	
	87	
	88	
	89	
	90	
	91	
	92	
	93	
	94	
	95	
	96	
	97	
	98	
	99	
	100	

Claim Final	Claim Original	Date
	101	
	102	
	103	
	104	
	105	
	106	
	107	
	108	
	109	
	110	
	111	
	112	
	113	
	114	
	115	
	116	
	117	
	118	
	119	
	120	
	121	
	122	
	123	
	124	
	125	
	126	
	127	
	128	
	129	
	130	
	131	
	132	
	133	
	134	
	135	
	136	
	137	
	138	
	139	
	140	
	141	
	142	
	143	
	144	
	145	
	146	
	147	
	148	
	149	
	150	

If more than 150 claims or 10 actions
staple additional sheet here

(LEFT INSIDE)

05/21/01

PATENT TRANSMITTAL LETTER
(SMALL ENTITY)

Attorney Docket No.
43426.00014

JC796 U.S. PTO 05/17/01

JC978 U.S. PTO 09/861229 05/17/01

TO THE COMMISSIONER FOR PATENTS:

Transmitted herewith for filing under 35 U.S.C. 111 and 37 C.F.R. is the patent application of:

Yigal Edery, **Nimrod Vered and David Kroll**

FOR:

Malicious Mobile Code Runtime Monitoring System and Methods:

☒ **Certificate of Mailing with Express Mailing Label No.: EL 701 364 462 US;**

☒ **10 Informal Sheets of Drawings: FIGS 1a-1c; 2, 3, 4; 5, 6a and 6b; 7a-7b and 8; 9 10A-10B; 11; 12a-12b**

☒ **Unsigned Combined Declaration and Power of Attorney;**

☒ **General Authorization and Request to Petition for Extension of Time; and**

☒ **Return Receipt Postcard**

CLAIMS AS FILED

FOR	FILED	ALLOWED	Extra	Rate	Additional Fee
Total Claims	76	-20	56	x $ 9.00	$ 504.00
Indep. Claims	11	-3	8	x $40.00	$ 320.00
Multiple Dependent Claims (check if applicable) ☐					$
				Basic Fee	$ 355.00
				Total Filing Fee	$1,179.00

☐ No additional fee is required for amendment.

☒ Please charge Deposit Account No. 05-0150 in the amount of $ 1,179.00

☒ The Commissioner is hereby authorized to charge and credit Deposit Account No. . 05-0150 As described below. A duplicate copy of this sheet is enclosed.

- ☒ Charge the amount of $1,179.00 as filing fee.
- ☒ Credit any overpayment.
- ☒ Charge any additional filing fees required under 37 C.F.R. 1.16.
- ☒ Charge any patent application processing fees under 37 C.F.R. 1.17.
- ☐ Charge the issue fee set in 37 C.F.R. 1.18 at the mailing of the Notice of Allowance, pursuant to 37 C.F.R. 1.311(b).

Daryl C. Josephson Date: 5/17/01

Daryl C. Josephson Reg. No. 37,365
Attorney for Applicants
Squire, Sanders & Dempsey L.L.P.
600 Hansen Way
Palo Alto, CA 94304-1043
Telephone: (650) 856-6500
Facsimile: (650) 856-3619

Attorney Docket No.: 43426.00014

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In Re Application Of: Examiner: Unknown

Yigal Edery, et al. Art Unit: Unknown

Serial No: Unknown

Filed: Date Herewith

For: Malicious Mobile Code Runtime Monitoring System and Methods

---

BOX PATENT APPLICATION
Commissioner of Patents
Washington, D.C. 20231

GENERAL AUTHORIZATION TO PETITION FOR EXTENSIONS OF TIME

Dear Sir:

With reference to the subject application, and pursuant to 37 C.F.R. § 1.136, Applicants hereby authorize and request the Commissioner to treat any correspondence requiring a petition for extension of time as containing such a request therefor for the appropriate length of time. This general authorization is effective during the pendency of this application, including any division or continuing application therefrom.

Where no check is received by the Commissioner, you are hereby authorized to charge payment of the requisite petition fees, or charge any additional fee required under 37 C.F.R. §

1.17, or credit any overpayment of same, to Deposit Account No. **05-0150**.

Date: 5/17/01

Respectfully submitted,
Yigal Edery

By: Daryl C. Josephson
Daryl C. Josephson
Attorney for Applicants
Reg. No. 32,365

Squire, Sanders & Dempsey L.L.P.
600 Hansen Way
Palo Alto, CA 94304-1043
Telephone (650) 856-6500
Facsimile (650) 856-3619

CERTIFICATE OF EXPRESS MAILING

I hereby certify that this paper (along with any paper referred to as being attached or enclosed) is being deposited with the United States Postal Service on the date shown below with sufficient postage as EXPRESS MAIL LABEL **EL 701 364 624 U.S.** in an envelope addressed to the Commissioner for Patents, Washington, D.C. 20231, on

Date: 5/17/01 By: Daryl C. Josephson
Daryl C.Josephson

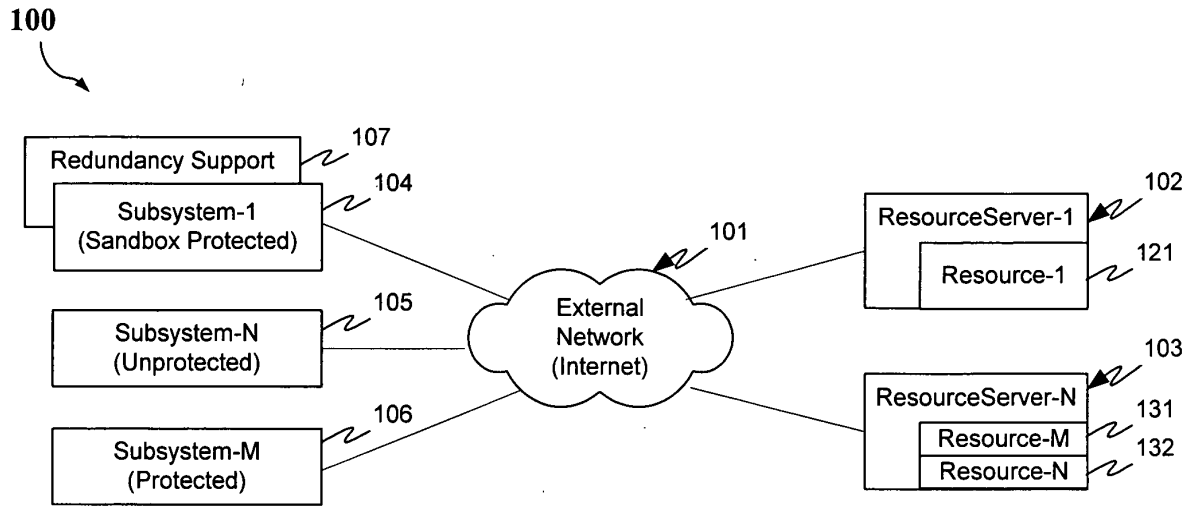

FIG. 1a

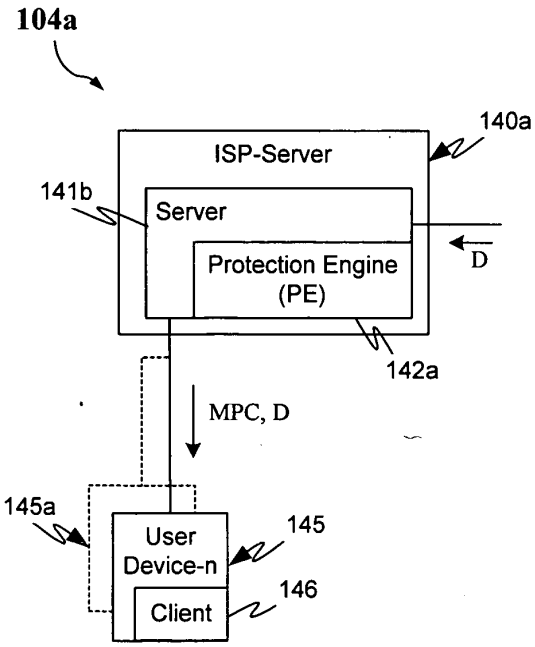

FIG. 1b

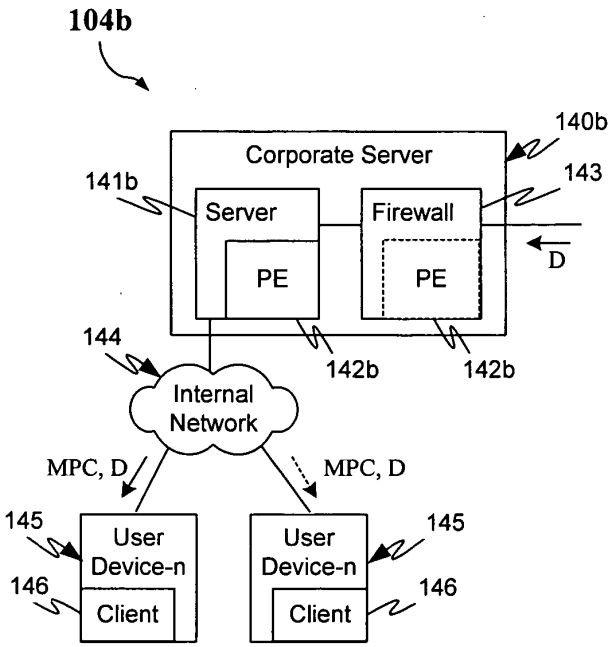

FIG. 1c

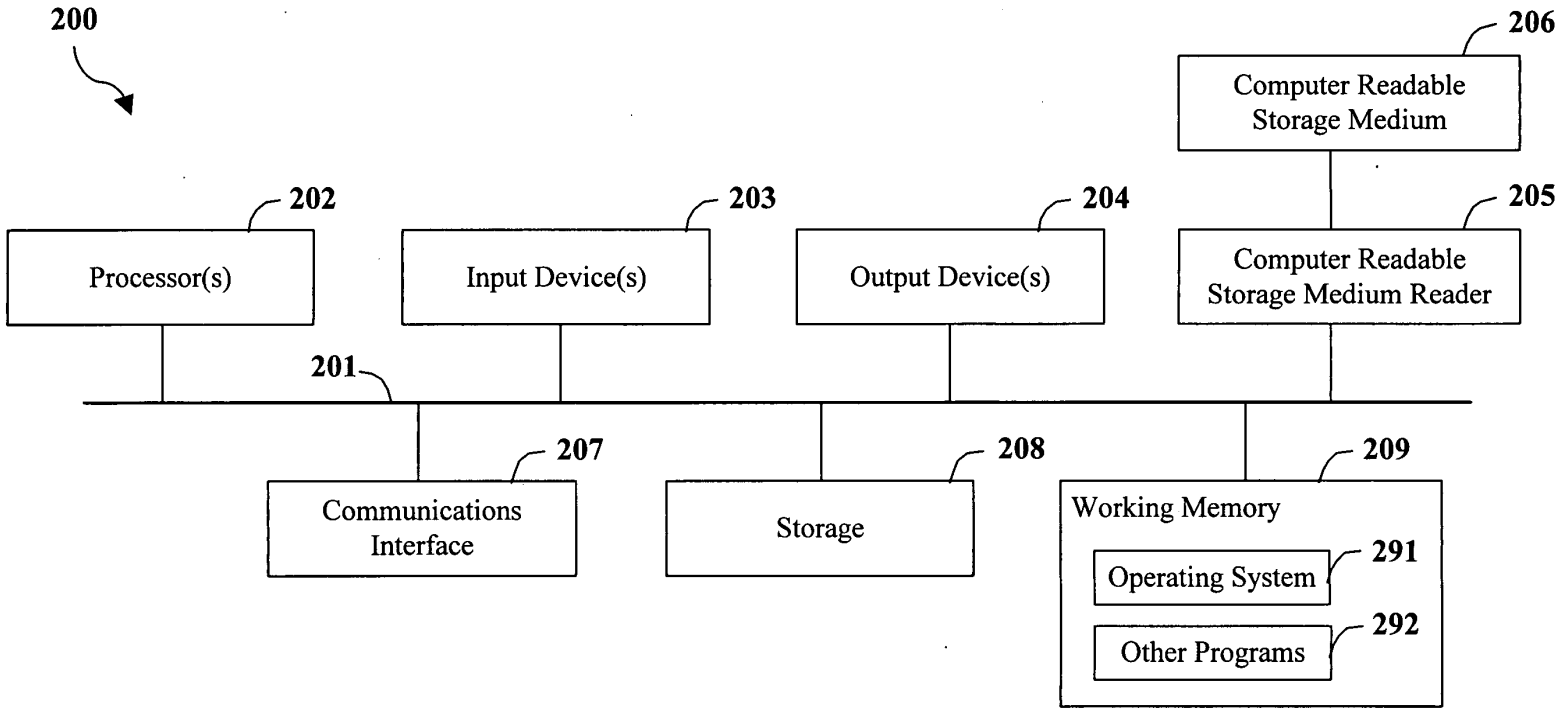
200
206
Computer Readable Storage Medium
202
Processor(s)
203
Input Device(s)
204
Output Device(s)
205
Computer Readable Storage Medium Reader
201
207
Communications Interface
208
Storage
209
Working Memory
Operating System
291
Other Programs
292

FIG. 2

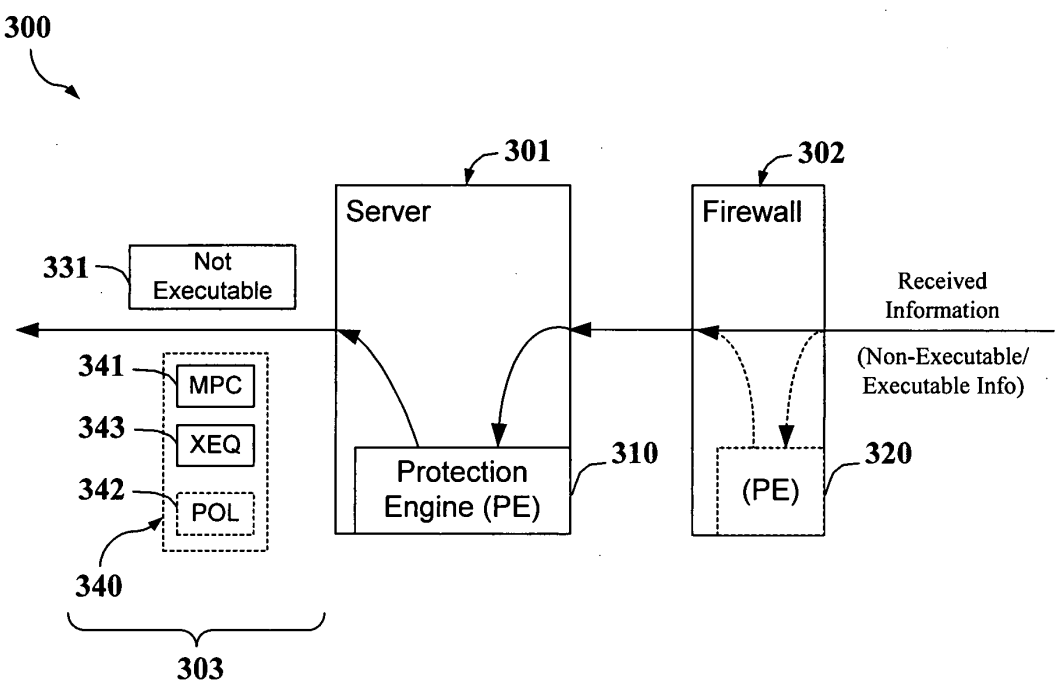
300
301
Server
302
Firewall
331
Not Executable
Received Information
(Non-Executable/ Executable Info)
341
MPC
343
XEQ
342
POL
340
303
Protection Engine (PE)
310
(PE)
320

FIG. 3

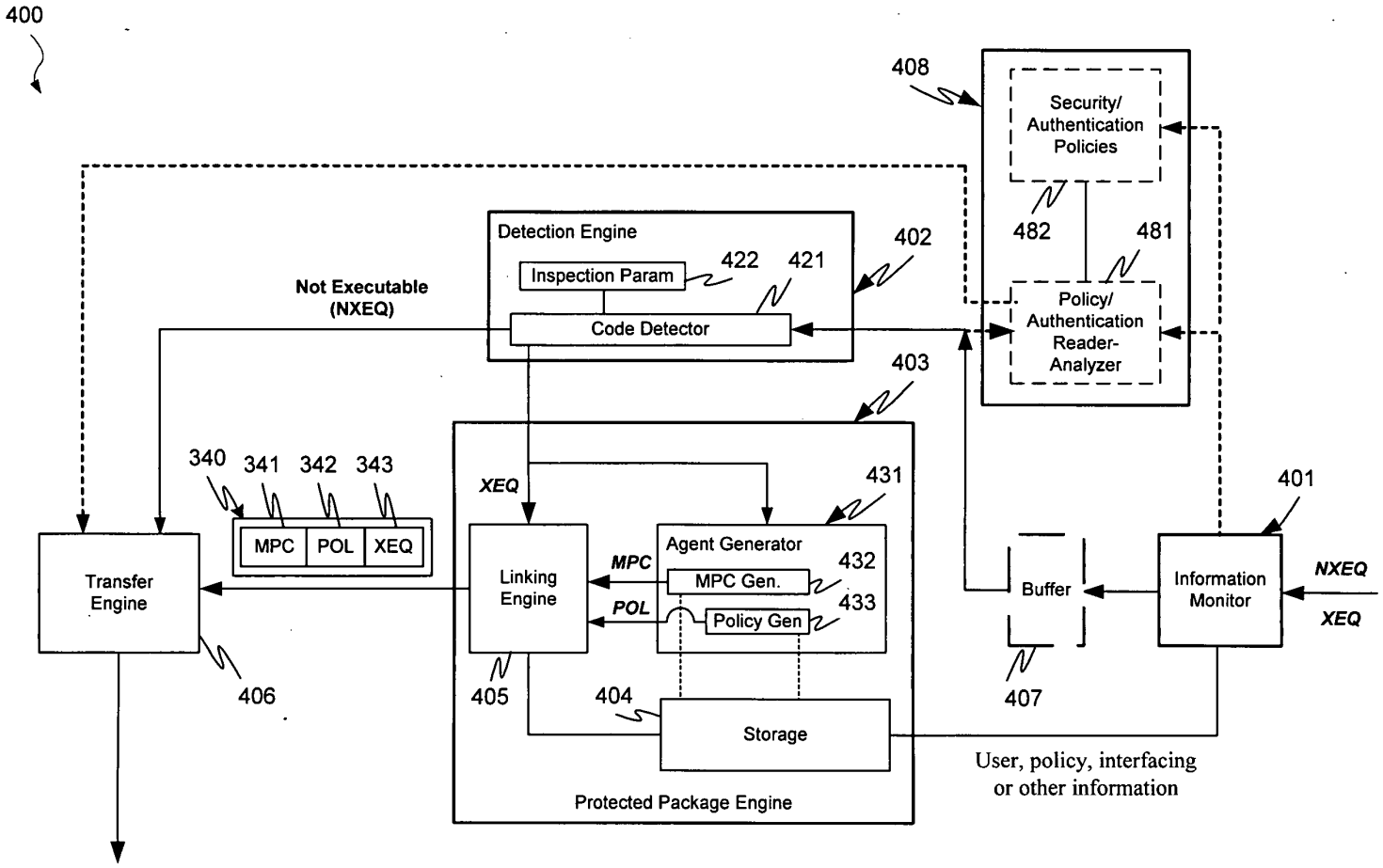
400
408
Security/ Authentication Policies
482
481
Policy/ Authentication Reader-Analyzer
Detection Engine
Inspection Param
422
421
402
Code Detector
Not Executable (NXEQ)
403
340
341
342
343
MPC
POL
XEQ
XEQ
431
Agent Generator
MPC
MPC Gen.
432
POL
Policy Gen
433
Linking Engine
Transfer Engine
406
405
404
Storage
Protected Package Engine
Buffer
407
Information Monitor
401
NXEQ
XEQ
User, policy, interfacing or other information

FIG. 4

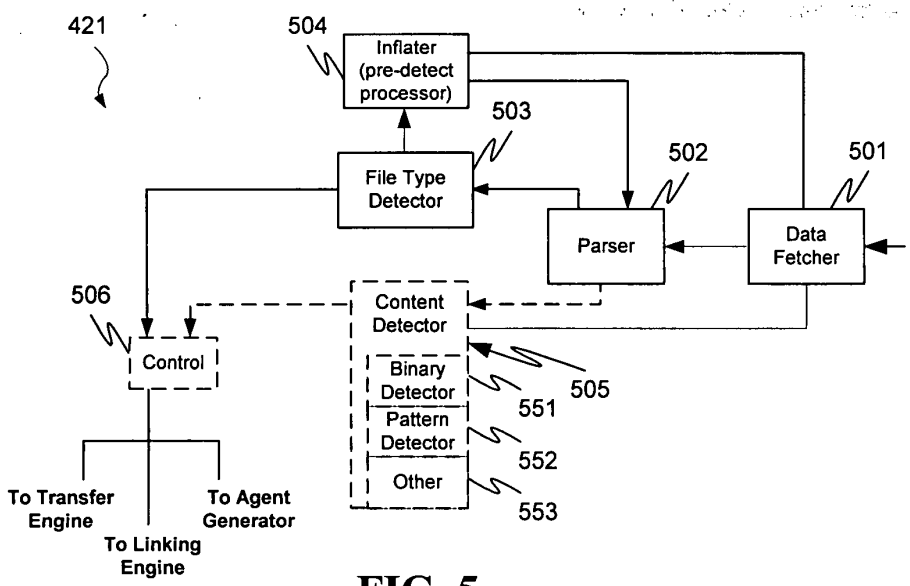

FIG. 5

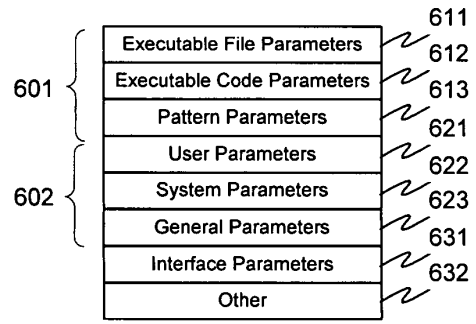

FIG. 6a

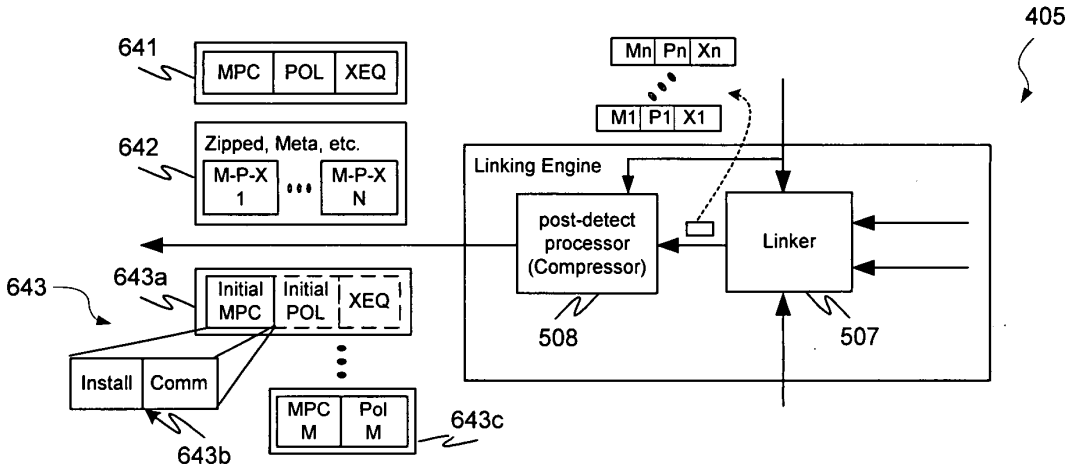

FIG. 6b

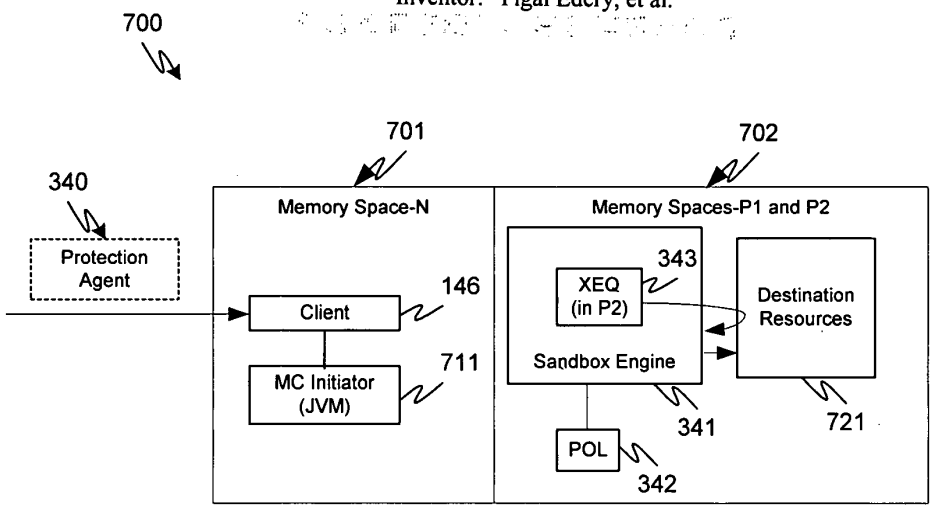

FIG. 7a

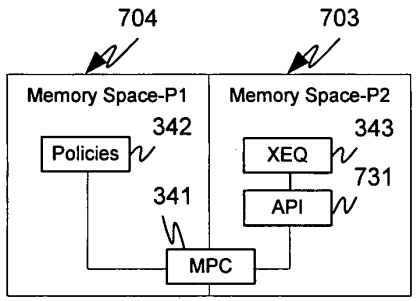

FIG. 7b

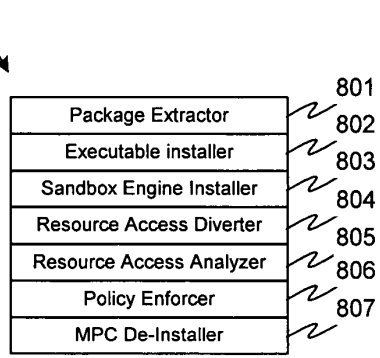

FIG. 8

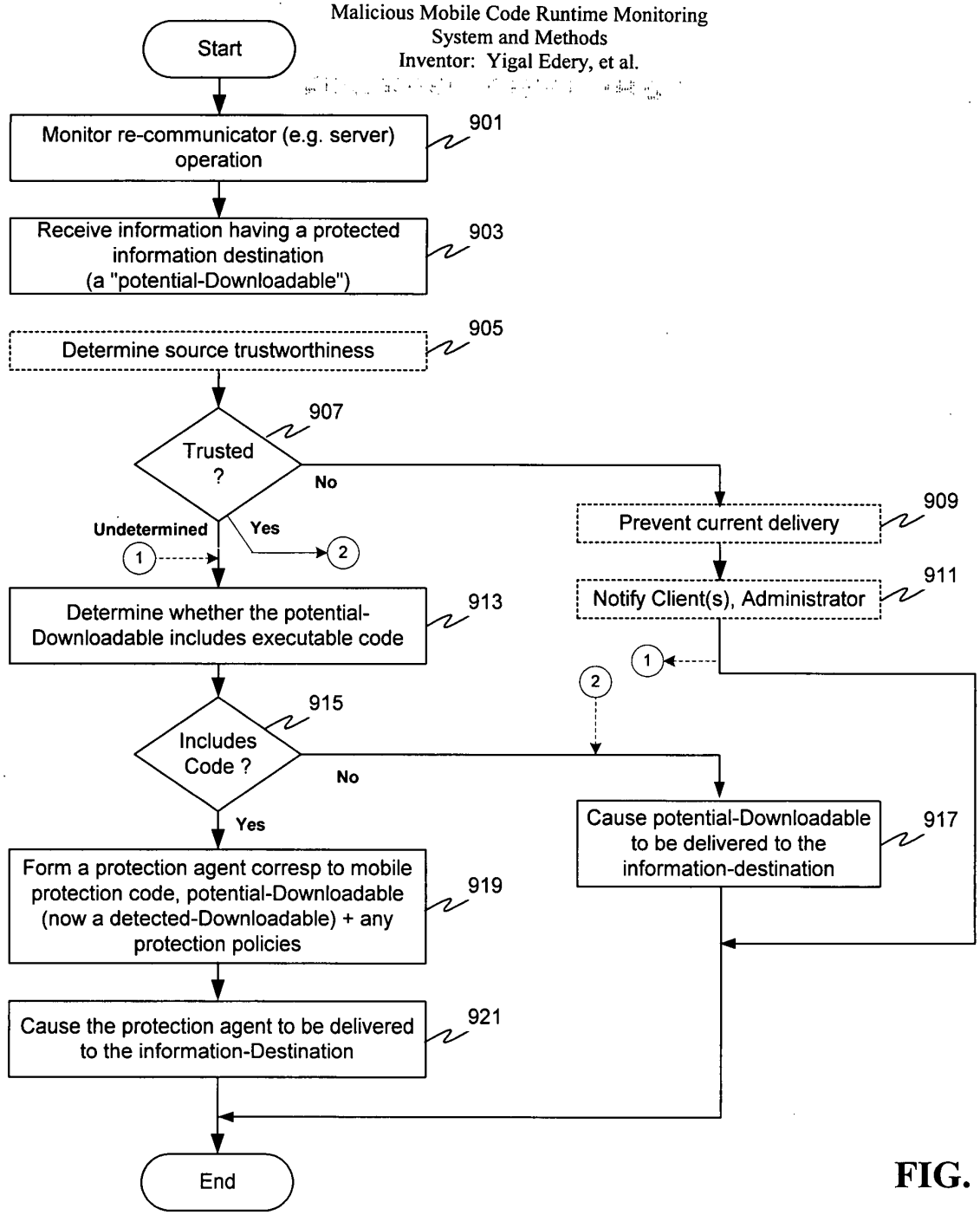
Start
Monitor re-communicator (e.g. server) operation
901
Receive information having a protected information destination (a "potential-Downloadable")
903
Determine source trustworthiness
905
Trusted ?
907
No
Yes
Undetermined
1
2
Prevent current delivery
909
Notify Client(s), Administrator
911
Determine whether the potential-Downloadable includes executable code
913
Includes Code ?
915
No
Yes
Cause potential-Downloadable to be delivered to the information-destination
917
Form a protection agent corresp to mobile protection code, potential-Downloadable (now a detected-Downloadable) + any protection policies
919
Cause the protection agent to be delivered to the information-Destination
921
End

FIG. 9

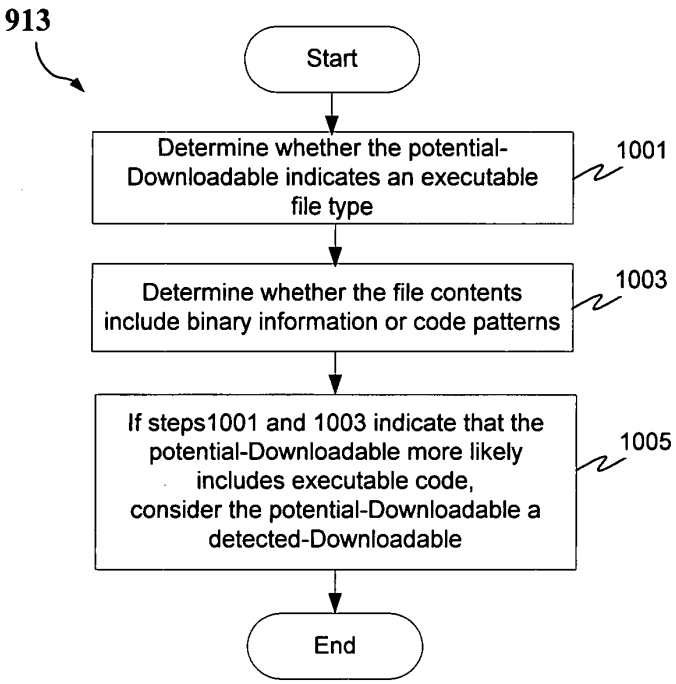

FIG. 10A

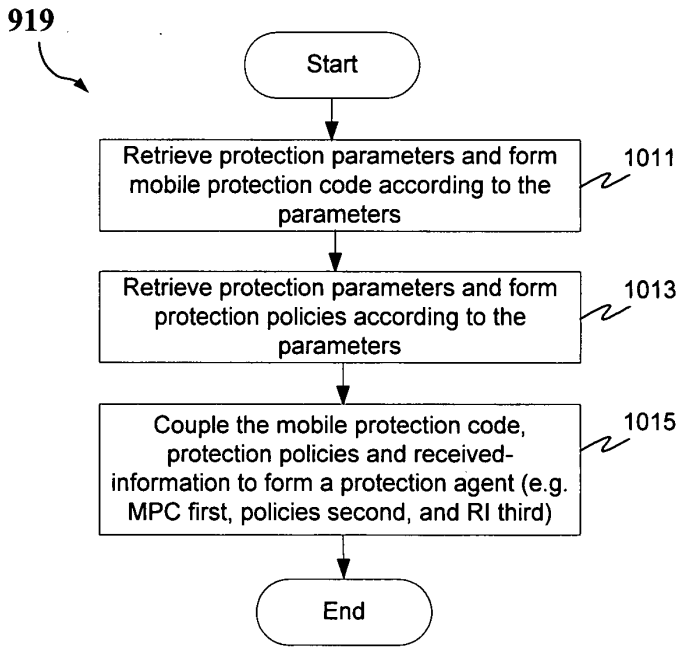

FIG. 10B

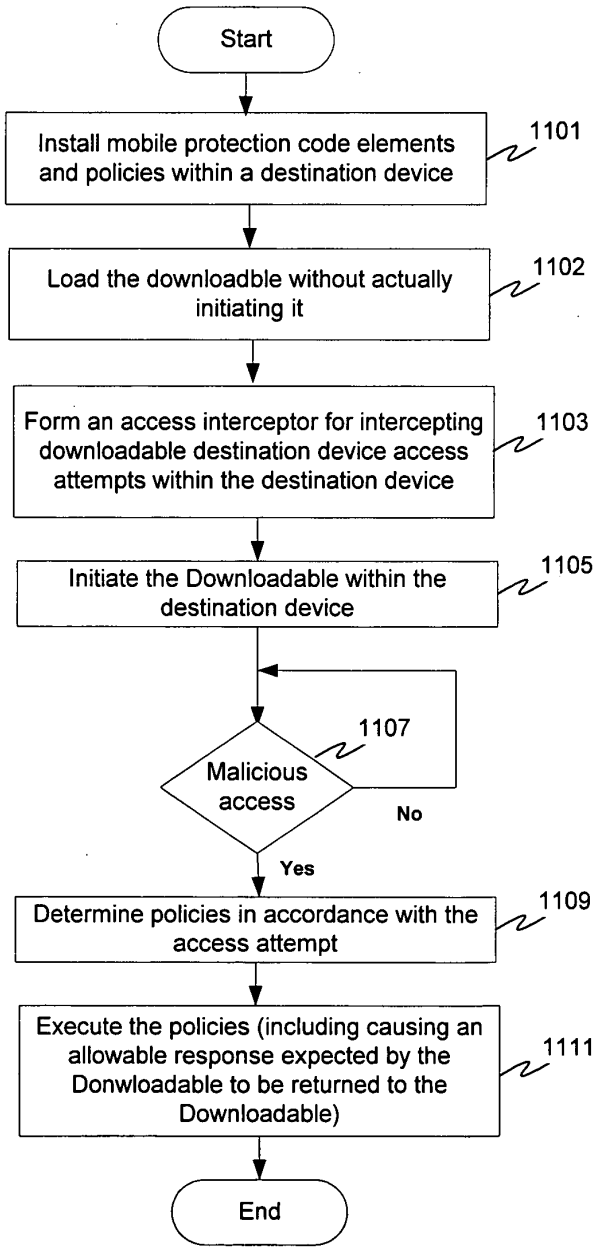
Start
Install mobile protection code elements and policies within a destination device
1101
Load the downloadble without actually initiating it
1102
Form an access interceptor for intercepting downloadable destination device access attempts within the destination device
1103
Initiate the Downloadable within the destination device
1105
Malicious access
1107
No
Yes
Determine policies in accordance with the access attempt
1109
Execute the policies (including causing an allowable response expected by the Donwloadable to be returned to the Downloadable)
1111
End

FIG. 11

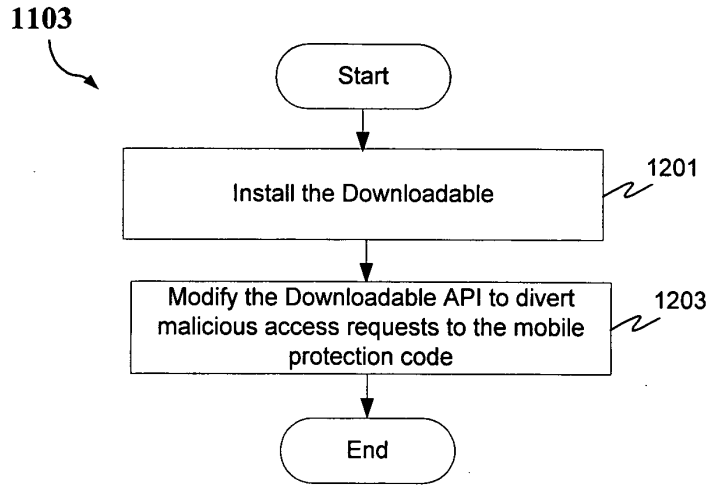

FIG. 12a

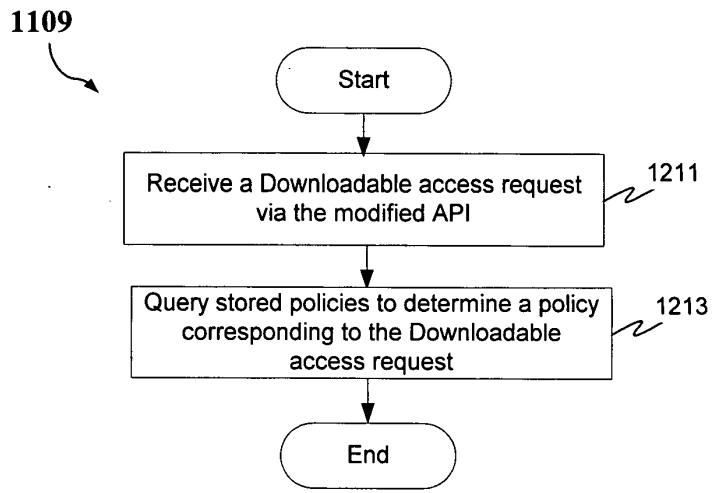

FIG. 12b

APPLICATION FOR

UNITED STATES PATENT

IN THE NAME OF

Yigal Edery, Nimrod Vered and David Kroll

OF

FINJAN SOFTWARE, LTD.

MALICIOUS MOBILE CODE RUNTIME MONITORING SYSTEM AND METHODS

DOCKET NO. 43426.00014

Please direct communications to:

Intellectual Property Department
Squire, Sanders & Dempsey L.L.P.
600 Hansen Way
Palo Alto, CA 94304-1043
(650) 856-6500

Express Mail Number **EL 701 364 624**

MALICIOUS MOBILE CODE RUNTIME MONITORING SYSTEM AND METHODS

PRIORITY REFERENCE TO RELATED APPLICATIONS

This application claims benefit of and hereby incorporates by reference provisional application serial number 60/205,591, entitled "Computer Network Malicious Code Run-time Monitoring," filed on May 17, 2000 by inventors Nimrod Itzhak Vered, et al. This application is also a Continuation-In-Part of and hereby incorporates by reference patent application serial number 09/539,667, entitled "System and Method for Protecting a Computer and a Network From Hostile Downloadables" filed on March 30, 2000 by inventor Shlomo Touboul. This application is also a Continuation-In-Part of and hereby incorporates by reference patent application serial number 09/551,302, entitled "System and Method for Protecting a Client During Runtime From Hostile Downloadables", filed on April 18, 2000 by inventor Shlomo Touboul.

BACKGROUND OF THE INVENTION

Field of the Invention

This invention relates generally to computer networks, and more particularly provides a system and methods for protecting network-connectable devices from undesirable downloadable operation.

Description of the Background Art

Advances in networking technology continue to impact an increasing number and diversity of users. The Internet, for example, already provides to expert, intermediate and even novice users the informational, product and service resources of over 100,000 interconnected networks owned by governments, universities, nonprofit groups, companies, etc. Unfortunately, particularly the Internet and other public networks have also become a major source of potentially system-fatal or otherwise damaging computer code commonly referred to as "viruses."

Efforts to forestall viruses from attacking networked computers have thus far met with only limited success at best. Typically, a virus protection program designed to identify and remove or protect against the initiating of known viruses is installed on a network firewall or individually networked computer. The program is then inevitably surmounted by some new virus that often causes damage to one or more computers. The damage is then assessed and, if isolated, the new virus is analyzed. A corresponding new virus protection program (or update thereof) is then developed and installed to combat the new virus, and the new program operates successfully until yet another new virus appears - and so on. Of course, damage has already typically been incurred.

To make matters worse, certain classes of viruses are not well recognized or understood, let alone protected against. It is observed by this inventor, for example, that Downloadable information comprising program code can include distributable components (e.g. Java™ applets and JavaScript scripts, ActiveX™ controls, Visual Basic, add-ins and/or others). It can also include, for example, application programs, Trojan horses, multiple compressed programs such as zip or meta files, among others. U.S. Patent 5,983,348 to Shuang, however, teaches a protection system for protecting

against only distributable components including "Java applets or ActiveX controls", and further does so using resource intensive and high bandwidth static Downloadable content and operational analysis, and modification of the Downloadable component; Shuang further fails to detect or protect against additional program code included within a tested Downloadable. U.S. Patent 5,974,549 to Golan teaches a protection system that further focuses only on protecting against ActiveX controls and not other distributable components, let alone other Downloadable types. U.S. patent 6,167,520 to Touboul enables more accurate protection than Shuang or Golan, but lacks the greater flexibility and efficiency taught herein, as do Shuang and Golan.

Accordingly, there remains a need for efficient, accurate and flexible protection of computers and other network connectable devices from malicious Downloadables.

SUMMARY OF THE INVENTION

The present invention provides protection systems and methods capable of protecting a personal computer ("PC") or other persistently or even intermittently network accessible devices or processes from harmful, undesirable, suspicious or other "malicious" operations that might otherwise be effectuated by remotely operable code. While enabling the capabilities of prior systems, the present invention is not nearly so limited, resource intensive or inflexible, and yet enables more reliable protection. For example, remotely operable code that is protectable against can include downloadable application programs, Trojan horses and program code groupings, as well as software "components", such as Java™ applets, ActiveX™ controls, JavaScript™/Visual Basic scripts, add-ins, etc., among others. Protection can also be provided in a distributed

interactively, automatically or mixed configurable manner using protected client, server or other parameters, redirection, local/remote logging, etc., and other server/client based protection measures can also be separately and/or interoperably utilized, among other examples.

In one aspect, embodiments of the invention provide for determining, within one or more network "servers" (e.g. firewalls, resources, gateways, email relays or other devices/processes that are capable of receiving-and-transferring a Downloadable) whether received information includes executable code (and is a "Downloadable"). Embodiments also provide for delivering static, configurable and/or extensible remotely operable protection policies to a Downloadable-destination, more typically as a sandboxed package including the mobile protection code, downloadable policies and one or more received Downloadables. Further client-based or remote protection code/policies can also be utilized in a distributed manner. Embodiments also provide for causing the mobile protection code to be executed within a Downloadable-destination in a manner that enables various Downloadable operations to be detected, intercepted or further responded to via protection operations. Additional server/information-destination device security or other protection is also enabled, among still further aspects.

A protection engine according to an embodiment of the invention is operable within one or more network servers, firewalls or other network connectable information re-communicating devices (as are referred to herein summarily one or more "servers" or "re-communicators"). The protection engine includes an information monitor for monitoring information received by the server, and a code detection engine for determining whether the received information includes executable code. The protection

engine also includes a packaging engine for causing a sandboxed package, typically including mobile protection code and downloadable protection policies to be sent to a Downloadable-destination in conjunction with the received information, if the received information is determined to be a Downloadable.

A sandboxed package according to an embodiment of the invention is receivable by and operable with a remote Downloadable-destination. The sandboxed package includes mobile protection code ("MPC") for causing one or more predetermined malicious operations or operation combinations of a Downloadable to be monitored or otherwise intercepted. The sandboxed package also includes protection policies (operable alone or in conjunction with further Downloadable-destination stored or received policies/MPCs) for causing one or more predetermined operations to be performed if one or more undesirable operations of the Downloadable is/are intercepted. The sandboxed package can also include a corresponding Downloadable and can provide for initiating the Downloadable in a protective "sandbox". The MPC/policies can further include a communicator for enabling further MPC/policy information or "modules" to be utilized and/or for event logging or other purposes.

A sandbox protection system according to an embodiment of the invention comprises an installer for enabling a received MPC to be executed within a Downloadable-destination (device/process) and further causing a Downloadable application program, distributable component or other received downloadable code to be received and installed within the Downloadable-destination. The protection system also includes a diverter for monitoring one or more operation attempts of the Downloadable, an operation analyzer for determining one or more responses to the attempts, and a

security enforcer for effectuating responses to the monitored operations. The protection system can further include one or more security policies according to which one or more protection system elements are operable automatically (e.g. programmatically) or in conjunction with user intervention (e.g. as enabled by the security enforcer). The security policies can also be configurable/extensible in accordance with further downloadable and/or Downloadable-destination information.

A method according to an embodiment of the invention includes receiving downloadable information, determining whether the downloadable information includes executable code, and causing a mobile protection code and security policies to be communicated to a network client in conjunction with security policies and the downloadable information if the downloadable information is determined to include executable code. The determining can further provide multiple tests for detecting, alone or together, whether the downloadable information includes executable code.

A further method according to an embodiment of the invention includes forming a sandboxed package that includes mobile protection code ("MPC"), protection policies, and a received, detected-Downloadable, and causing the sandboxed package to be communicated to and installed by a receiving device or process ("user device") for responding to one or more malicious operation attempts by the detected-Downloadable from within the user device. The MPC/policies can further include a base "module" and a "communicator" for enabling further up/downloading of one or more further "modules" or other information (e.g. events, user/user device information, etc.).

Another method according to an embodiment of the invention includes installing, within a user device, received mobile protection code ("MPC") and protection policies in

conjunction with the user device receiving a downloadable application program, component or other Downloadable(s). The method also includes determining, by the MPC, a resource access attempt by the Downloadable, and initiating, by the MPC, one or more predetermined operations corresponding to the attempt. (Predetermined operations can, for example, comprise initiating user, administrator, client, network or protection system determinable operations, including but not limited to modifying the Downloadable operation, extricating the Downloadable, notifying a user/another, maintaining a local/remote log, causing one or more MPCs/policies to be downloaded, etc.)

Advantageously, systems and methods according to embodiments of the invention enable potentially damaging, undesirable or otherwise malicious operations by even unknown mobile code to be detected, prevented, modified and/or otherwise protected against without modifying the mobile code. Such protection is further enabled in a manner that is capable of minimizing server and client resource requirements, does not require pre-installation of security code within a Downloadable-destination, and provides for client specific or generic and readily updateable security measures to be flexibly and efficiently implemented. Embodiments further provide for thwarting efforts to bypass security measures (e.g. by "hiding" undesirable operation causing information within apparently inert or otherwise "friendly" downloadable information) and/or dividing or combining security measures for even greater flexibility and/or efficiency.

Embodiments also provide for determining protection policies that can be downloaded and/or ascertained from other security information (e.g. browser settings, administrative policies, user input, uploaded information, etc.). Different actions in response to different Downloadable operations, clients, users and/or other criteria are also

enabled, and embodiments provide for implementing other security measures, such as verifying a downloadable source, certification, authentication, etc. Appropriate action can also be accomplished automatically (e.g. programmatically) and/or in conjunction with alerting one or more users/administrators, utilizing user input, etc. Embodiments further enable desirable Downloadable operations to remain substantially unaffected, among other aspects.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1a is a block diagram illustrating a network system in accordance with an embodiment of the present invention;

FIG. 1b is a block diagram illustrating a network subsystem example in accordance with an embodiment of the invention;

FIG. 1c is a block diagram illustrating a further network subsystem example in accordance with an embodiment of the invention;

FIG. 2 is a block diagram illustrating a computer system in accordance with an embodiment of the invention;

FIG. 3 is a flow diagram broadly illustrating a protection system host according to an embodiment of the invention;

FIG. 4 is a block diagram illustrating a protection engine according to an embodiment of the invention;

FIG. 5 is a block diagram illustrating a content inspection engine according to an embodiment of the invention;

FIG. 6a is a block diagram illustrating protection engine parameters according to an embodiment of the invention;

FIG. 6b is a flow diagram illustrating a linking engine use in conjunction with ordinary, compressed and distributable sandbox package utilization, according to an embodiment of the invention;

FIG. 7a is a flow diagram illustrating a sandbox protection system operating within a destination system, according to an embodiment of the invention;

FIG. 7b is a block diagram illustrating memory allocation usable in conjunction with the protection system of FIG. 7a, according to an embodiment of the invention;

FIG. 7c is a block diagram illustrating a mobile protection code according to an embodiment of the invention;

FIG. 8 is a flowchart illustrating a method for examining a Downloadable in accordance with the present invention;

FIG. 9 is a flowchart illustrating a server based protection method according to an embodiment of the invention;

FIG. 10a is a flowchart illustrating method for determining if a potential-Downloadable includes or is likely to include executable code, according to an embodiment of the invention;

FIG. 10b is a flowchart illustrating a method for forming a protection agent, according to an embodiment of the invention;

FIG. 11 is a flowchart illustrating a method for protecting a Downloadable destination according to an embodiment of the invention;

FIG. 12a is a flowchart illustrating a method for forming a Downloadable access interceptor according to an embodiment of the invention; and

FIG. 12b is a flowchart illustrating a method for implementing mobile protection policies according to an embodiment of the invention.

DETAILED DESCRIPTION

In providing malicious mobile code runtime monitoring systems and methods, embodiments of the invention enable actually or potentially undesirable operations of even unknown malicious code to be efficiently and flexibly avoided. Embodiments provide, within one or more "servers" (e.g. firewalls, resources, gateways, email relays or other information re-communicating devices), for receiving downloadable-information and detecting whether the downloadable-information includes one or more instances of executable code (e.g. as with a Trojan horse, zip/meta file etc.). Embodiments also provide for separately or interoperably conducting additional security measures within the server, within a Downloadable-destination of a detected-Downloadable, or both.

Embodiments further provide for causing mobile protection code ("MPC") and downloadable protection policies to be communicated to, installed and executed within one or more received information destinations in conjunction with a detected-Downloadable. Embodiments also provide, within an information-destination, for detecting malicious operations of the detected-Downloadable and causing responses thereto in accordance with the protection policies (which can correspond to one or more user, Downloadable, source, destination, or other parameters), or further downloaded or downloadable-destination based policies (which can also be configurable or extensible). (Note that the term "or", as used herein, is generally intended to mean "and/or" unless otherwise indicated.)

FIGS. 1a through 1c illustrate a computer network system 100 according to an embodiment of the invention. FIG. 1a broadly illustrates system 100, while FIGS. 1b and

1c illustrate exemplary protectable subsystem implementations corresponding with system 104 or 106 of FIG. 1a.

Beginning with FIG. 1a, computer network system 100 includes an external computer network 101, such as a Wide Area Network or "WAN" (e.g. the Internet), which is coupled to one or more network resource servers (summarily depicted as resource server-1 102 and resource server-N 103). Where external network 101 includes the Internet, resource servers 1-N (102, 103) might provide one or more resources including web pages, streaming media, transaction-facilitating information, program updates or other downloadable information, summarily depicted as resources 121, 131 and 132. Such information can also include more traditionally viewed "Downloadables" or "mobile code" (i.e. distributable components), as well as downloadable application programs or other further Downloadables, such as those that are discussed herein. (It will be appreciated that interconnected networks can also provide various other resources as well.)

Also coupled via external network 101 are subsystems 104-106. Subsystems 104-106 can, for example, include one or more servers, personal computers ("PCs"), smart appliances, personal information managers or other devices/processes that are at least temporarily or otherwise intermittently directly or indirectly connectable in a wired or wireless manner to external network 101 (e.g. using a dialup, DSL, cable modem, cellular connection, IR/RF, or various other suitable current or future connection alternatives). One or more of subsystems 104-106 might further operate as user devices that are connectable to external network 101 via an internet service provider ("ISP") or

local area network ("LAN"), such as a corporate intranet, or home, portable device or smart appliance network, among other examples.

FIG. 1a also broadly illustrates how embodiments of the invention are capable of selectively, modifiably or extensibly providing protection to one or more determinable ones of networked subsystems 104-106 or elements thereof (not shown) against potentially harmful or other undesirable ("malicious") effects in conjunction with receiving downloadable information. "Protected" subsystem 104, for example, utilizes a protection in accordance with the teachings herein, while "unprotected" subsystem-N 105 employs no protection, and protected subsystem-M 106 might employ one or more protections including those according to the teachings herein, other protection, or some combination.

System 100 implementations are also capable of providing protection to redundant elements 107 of one or more of subsystems 104-106 that might be utilized, such as backups, failsafe elements, redundant networks, etc. Where included, such redundant elements are also similarly protectable in a separate, combined or coordinated manner using embodiments of the present invention either alone or in conjunction with other protection mechanisms. In such cases, protection can be similarly provided singly, as a composite of component operations or in a backup fashion. Care should, however, be exercised to avoid potential repeated protection engine execution corresponding to a single Downloadable; such "chaining" can cause a Downloadable to operate incorrectly or not at all, unless a subsequent detection engine is configured to recognize a prior packaging of the Downloadable..

FIGS. 1b and 1c further illustrate, by way of example, how protection systems according to embodiments of the invention can be utilized in conjunction with a wide variety of different system implementations. In the illustrated examples, system elements are generally configurable in a manner commonly referred to as a "client-server" configuration, as is typically utilized for accessing Internet and many other network resources. For clarity sake, a simple client-server configuration will be presumed unless otherwise indicated. It will be appreciated, however, that other configurations of interconnected elements might also be utilized (e.g. peer-peer, routers, proxy servers, networks, converters, gateways, services, network reconfiguring elements, etc.) in accordance with a particular application.

The FIG. 1b example shows how a suitable protected system 104a (which can correspond to subsystem-1 104 or subsystem-M 106 of FIG. 1) can include a protection-initiating host "server" or "re-communicator" (e.g. ISP server140a), one or more user devices or "Downloadable-destinations" 145, and zero or more redundant elements (which elements are summarily depicted as redundant client device/process 145a). In this example, ISP server 140a includes one or more email, Internet or other servers 141a, or other devices or processes capable of transferring or otherwise "re-communicating" downloadable information to user devices 145. Server 141a further includes protection engine or "PE" 142a, which is capable of supplying mobile protection code ("MPC") and protection policies for execution by client devices 145. One or more of user devices 145 can further include a respective one or more clients 146 for utilizing information received via server 140a, in accordance with which MPC and protection policies are operable to

protect user devices 145 from detrimental, undesirable or otherwise "malicious" operations of downloadable information also received by user device 145.

The FIG. 1c example shows how a further suitable protected system 104b can include, in addition to a "re-communicator", such as server 142b, a firewall 143c (e.g. as is typically the case with a corporate intranet and many existing or proposed home/smart networks.) In such cases, a server 141b or firewall 143 can operate as a suitable protection engine host. A protection engine can also be implemented in a more distributed manner among two or more protection engine host systems or host system elements, such as both of server 141b and firewall 143, or in a more integrated manner, for example, as a standalone device. Redundant system or system protection elements can also be similarly provided in a more distributed or integrated manner (see above).

System 104b also includes internal network 144 and user devices 145. User devices 145 further include a respective one or more clients 146 for utilizing information received via server 140a, in accordance with which the MPCs or protection policies are operable. (As in the previous example, one or more of user devices 145 can also include or correspond with similarly protectable redundant system elements, which are not shown.)

It will be appreciated that the configurations of FIGS 1a-1c are merely exemplary. Alternative embodiments might, for example, utilize other suitable connections, devices or processes. One or more devices can also be configurable to operate as a network server, firewall, smart router, a resource server servicing deliverable third-party/manufacturer postings, a user device operating as a firewall/server, or other information-suppliers or intermediaries (i.e. as a "re-communicator" or "server") for

servicing one or more further interconnected devices or processes or interconnected levels of devices or processes. Thus, for example, a suitable protection engine host can include one or more devices or processes capable of providing or supporting the providing of mobile protection code or other protection consistent with the teachings herein. A suitable information-destination or "user device" can further include one or more devices or processes (such as email, browser or other clients) that are capable of receiving and initiating or otherwise hosting a mobile code execution.

FIG. 2 illustrates an exemplary computing system 200, that can comprise one or more of the elements of FIGS. 1a through 1c. While other application-specific alternatives might be utilized, it will be presumed for clarity sake that system 100 elements (FIGS. 1a-c) are implemented in hardware, software or some combination by one or more processing systems consistent therewith, unless otherwise indicated.

Computer system 200 comprises elements coupled via communication channels (e.g. bus 201) including one or more general or special purpose processors 202, such as a Pentium® or Power PC®, digital signal processor ("DSP"), etc. System 200 elements also include one or more input devices 203 (such as a mouse, keyboard, microphone, pen, etc.), and one or more output devices 204, such as a suitable display, speakers, actuators, etc., in accordance with a particular application.

System 200 also includes a computer readable storage media reader 205 coupled to a computer readable storage medium 206, such as a storage/memory device or hard or removable storage/memory media; such devices or media are further indicated separately as storage device 208 and memory 209, which can include hard disk variants, floppy/compact disk variants, digital versatile disk ("DVD") variants, smart cards, read

only memory, random access memory, cache memory, etc., in accordance with a particular application. One or more suitable communication devices 207 can also be included, such as a modem, DSL, infrared or other suitable transceiver, etc. for providing inter-device communication directly or via one or more suitable private or public networks that can include but are not limited to those already discussed.

Working memory further includes operating system ("OS") elements and other programs, such as application programs, mobile code, data, etc. for implementing system 100 elements that might be stored or loaded therein during use. The particular OS can vary in accordance with a particular device, features or other aspects in accordance with a particular application (e.g. Windows, Mac, Linux, Unix or Palm OS variants, a proprietary OS, etc.). Various programming languages or other tools can also be utilized, such as C++, Java, Visual Basic, etc. As will be discussed, embodiments can also include a network client such as a browser or email client, e.g. as produced by Netscape, Microsoft or others, a mobile code executor such as an OS task manager, Java Virtual Machine ("JVM"), etc., and an application program interface ("API"), such as a Microsoft Windows or other suitable element in accordance with the teachings herein. (It will also become apparent that embodiments might also be implemented in conjunction with a resident application or combination of mobile code and resident application components.)

One or more system 200 elements can also be implemented in hardware, software or a suitable combination. When implemented in software (e.g. as an application program, object, downloadable, servlet, etc. in whole or part), a system 200 element can be communicated transitionally or more persistently from local or remote storage to

memory (or cache memory, etc.) for execution, or another suitable mechanism can be utilized, and elements can be implemented in compiled or interpretive form. Input, intermediate or resulting data or functional elements can further reside more transitionally or more persistently in a storage media, cache or more persistent volatile or non-volatile memory, (e.g. storage device 207 or memory 208) in accordance with a particular application.

FIG. 3 illustrates an interconnected re-communicator 300 generally consistent with system 140b of FIG. 1, according to an embodiment of the invention. As with system 140b, system 300 includes a server 301, and can also include a firewall 302. In this implementation, however, either server 301 or firewall 302 (if a firewall is used) can further include a protection engine (310 or 320 respectively). Thus, for example, an included firewall can process received information in a conventional manner, the results of which can be further processed by protection engine 310 of server 301, or information processed by protection engine 320 of an included firewall 302 can be processed in a conventional manner by server 301. (For clarity sake, a server including a singular protection engine will be presumed, with or without a firewall, for the remainder of the discussion unless otherwise indicated. Note, however, that other embodiments consistent with the teachings herein might also be utilized.)

FIG. 3 also shows how information received by server 301 (or firewall 302) can include non-executable information, executable information or a combination of non-executable and one or more executable code portions (e.g. so-called Trojan horses that include a hostile Downloadable within a friendly one, combined, compressed or otherwise encoded files, etc.). Particularly such combinations will likely remain

undetected by a firewall or other more conventional protection systems. Thus, for convenience, received information will also be referred to as a "potential-Downloadable", and received information found to include executable code will be referred to as a "Downloadable" or equivalently as a "detected-Downloadable" (regardless of whether the executable code includes one or more application programs, distributable "components" such as Java, ActiveX, add-in, etc.).

Protection engine 310 provides for detecting whether received potential-Downloadables include executable code, and upon such detection, for causing mobile protection code ("MPC") to be transferred to a device that is a destination of the potential-Downloadable (or "Downloadable-destination"). Protection engine 310 can also provide protection policies in conjunction with the MPC (or thereafter as well), which MPC/policies can be automatically (e.g. programmatically) or interactively configurable in accordance user, administrator, downloadable source, destination, operation, type or various other parameters alone or in combination (see below). Protection engine 310 can also provide or operate separately or interoperably in conjunction with one or more of certification, authentication, downloadable tagging, source checking, verification, logging, diverting or other protection services via the MPC, policies, other local/remote server or destination processing, etc. (e.g. which can also include protection mechanisms taught by the above-noted prior applications; see FIG. 4).

Operationally, protection engine 310 of server 301 monitors information received by server 301 and determines whether the received information is deliverable to a protected destination, e.g. using a suitable monitor/data transfer mechanism and comparing a destination-address of the received information to a protected destination set,

such as a protected destinations list, array, database, etc. (All deliverable information or one or more subsets thereof might also be monitored.) Protection engine 310 further analyzes the potential-Downloadable and determines whether the potential-Downloadable includes executable code. If not, protection engine 310 enables the not executable potential-Downloadable 331 to be delivered to its destination in an unaffected manner.

In conjunction with determining that the potential-Downloadable is a detected-Downloadable, protection engine 310 also causes mobile protection code or "MPC" 341 to be communicated to the Downloadable-destination of the Downloadable, more suitably in conjunction with the detected-Downloadable 343 (see below). Protection engine 310 further causes downloadable protection policies 342 to be delivered to the Downloadable-destination, again more suitably in conjunction with the detected-Downloadable. Protection policies 342 provide parameters (or can additionally or alternatively provide additional mobile code) according to which the MPC is capable of determining or providing applicable protection to a Downloadable-destination against malicious Downloadable operations.

(One or more "checked", tag, source, destination, type, detection or other security result indicators, which are not shown, can also be provided as corresponding to determined non-Downloadables or Downloadables, e.g. for testing, logging, further processing, further identification tagging or other purposes in accordance with a particular application.)

Further MPCs, protection policies or other information are also deliverable to a the same or another destination, for example, in accordance with communication by an MPC/protection policies already delivered to a downloadable-destination. Initial or

subsequent MPCs/policies can further be selected or configured in accordance with a Downloadable-destination indicated by the detected-Downloadable, destination-user or administrative information, or other information providable to protection engine 310 by a user, administrator, user system, user system examination by a communicated MPC, etc. (Thus, for example, an initial MPC/policies can also be initially provided that are operable with or optimized for more efficient operation with different Downloadable-destinations or destination capabilities.)

While integrated protection constraints within the MPC might also be utilized, providing separate protection policies has been found to be more efficient, for example, by enabling more specific protection constraints to be more easily updated in conjunction with detected-Downloadable specifics, post-download improvements, testing, etc. Separate policies can further be more efficiently provided (e.g. selected, modified, instantiated, etc.) with or separately from an MPC, or in accordance with the requirements of a particular user, device, system, administration, later improvement, etc., as might also be provided to protection engine 310 (e.g. via user/MPC uploading, querying, parsing a Downloadable, or other suitable mechanism implemented by one or more servers or Downloadable-destinations).

(It will also become apparent that performing executable code detection and communicating to a downloadable-Destination an MPC and any applicable policies as separate from a detected-Downloadable is more accurate and far less resource intensive than, for example, performing content and operation scanning, modifying a Downloadable, or providing completely Downloadable-destination based security.)

System 300 enables a single or extensible base-MPC to be provided, in anticipation or upon receipt of a first Downloadable, that is utilized thereafter to provide protection of one or more Downloadable-destinations. It is found, however, that providing an MPC upon each detection of a Downloadable (which is also enabled) can provide a desirable combination of configurability of the MPC/policies and lessened need for management (e.g. given potentially changing user/destination needs, enabling testing, etc.).

Providing an MPC upon each detection of a Downloadable also facilitates a lessened demand on destination resources, e.g. since information-destination resources used in executing the MPC/policies can be re-allocated following such use. Such alternatives can also be selectively, modifiably or extensibly provided (or further in accordance with other application-specific factors that might also apply.) Thus, for example, a base-MPC or base-policies might be provided to a user device that is/are extensible via additionally downloadable "modules" upon server 301 detection of a Downloadable deliverable to the same user device, among other alternatives.

In accordance with a further aspect of the invention, it is found that improved efficiency can also be achieved by causing the MPC to be executed within a Downloadable-destination in conjunction with, and further, prior to initiation of the detected Downloadable. One mechanism that provides for greater compatibility and efficiency in conjunction with conventional client-based Downloadable execution is for a protection engine to form a sandboxed package 340 including MPC 341, the detected-Downloadable 343 and any policies 342. For example, where the Downloadable is a binary executable to be executed by an operating system, protection engine 310 forms a

protected package by concatenating, within sandboxed package 340, MPC 341 for delivery to a Downloadable-destination first, followed by protection policies 342 and Downloadable 343. (Concatenation or techniques consistent therewith can also be utilized for providing a protecting package corresponding to a Java applet for execution by a JVM of a Downloadable-destination, or with regard to ActiveX controls, add-ins or other distributable components, etc.)

The above concatenation or other suitable processing will result in the following. Upon receipt of sandboxed package 340 by a compatible browser, email or other destination-client and activating of the package by a user or the destination-client, the operating system (or a suitable responsively initiated distributed component host) will attempt to initiate sandboxed package 340 as a single Downloadable. Such processing will, however, result in initiating the MPC 341 and -in accordance with further aspects of the invention- the MPC will initiate the Downloadable in a protected manner, further in accordance with any applicable included or further downloaded protection policies 342. (While system 300 is also capable of ascertaining protection policies stored at a Downloadable-destination, e.g. by poll, query, etc. of available destination information, including at least initial policies within a suitable protecting package is found to avoid associated security concerns or inefficiencies.)

Turning to FIG. 4, a protection engine 400 generally consistent with protection engine 310 (or 320) of FIG. 3 is illustrated in accordance with an embodiment of the invention. Protection engine 400 comprises information monitor 401, detection engine 402, and protected packaging engine 403, which further includes agent generator 431, storage 404, linking engine 405, and transfer engine 406. Protection engine 400 can also

include a buffer 407, for temporarily storing a received potential-Downloadable, or one or more systems for conducting additional authentication, certification, verification or other security processing (e.g. summarily depicted as security system 408) Protection engine 400 can further provide for selectively re-directing, further directing, logging, etc. of a potential/detected Downloadable or information corresponding thereto in conjunction with detection, other security, etc., in accordance with a particular application.

(Note that FIG. 4, as with other figures included herein, also depicts exemplary signal flow arrows; such arrows are provided to facilitate discussion, and should not be construed as exclusive or otherwise limiting.)

Information monitor 401 monitors potential-Downloadables received by a host server and provides the information via buffer 407 to detection engine 402 or to other system 400 elements. Information monitor 401 can be configured to monitor host server download operations in conjunction with a user or a user-device that has logged-on to the server, or to receive information via a server operation hook, servlet, communication channel or other suitable mechanism.

Information monitor 401 can also provide for transferring, to storage 404 or other protection engine elements, configuration information including, for example, user, MPC, protection policy, interfacing or other configuration information (e.g. see FIG. 6). Such configuration information monitoring can be conducted in accordance with a user/device logging onto or otherwise accessing a host server, via one or more of configuration operations, using an applet to acquire such information from or for a particular user, device or devices, via MPC/policy polling of a user device, or via other suitable mechanisms.

Detection engine 402 includes code detector 421, which receives a potential-Downloadable and determines, more suitably in conjunction with inspection parameters 422, whether the potential-Downloadable includes executable code and is thus a "detected-Downloadable". (Code detector 421 can also include detection processors for performing file decompression or other "decoding", or such detection-facilitating processing as decryption, utilization/support of security system 408, etc. in accordance with a particular application.)

Detection engine 402 further transfers a detected-downloadable ("XEQ") to protected packaging engine 403 along with indicators of such detection, or a determined non-executable ("NXEQ") to transfer engine 406. (Inspection parameters 422 enable analysis criteria to be readily updated or varied, for example, in accordance with particular source, destination or other potential Downloadable impacting parameters, and are discussed in greater detail with reference to FIG. 5). Detection engine 402 can also provide indicators for delivery of initial and further MPCs/policies, for example, prior to or in conjunction with detecting a Downloadable and further upon receipt of an indicator from an already downloaded MPC/policy. A downloaded MPC/policy can further remain resident at a user device with further modules downloaded upon or even after delivery of a sandboxed package. Such distribution can also be provided in a configurable manner, such that delivery of a complete package or partial packages are automatically or interactively determinable in accordance with user/administrative preferences/policies, among other examples.

Packaging engine 403 provides for generating mobile protection code and protection policies, and for causing delivery thereof (typically with a detected-

Downloadable) to a Downloadable-destination for protecting the Downloadable-destination against malicious operation attempts by the detected Downloadable. In this example, packaging engine 403 includes agent generator 431, storage 404 and linking engine 405.

Agent generator 431 includes an MPC generator 432 and a protection policy generator 433 for "generating" an MPC and a protection policy (or set of policies) respectively upon receiving one or more "generate MPC/policy" indicators from detection engine 402, indicating that a potential-Downloadable is a detected-Downloadable. MPC generator 432 and protection policy generator 433 provide for generating MPCs and protection policies respectively in accordance with parameters retrieved from storage 404. Agent generator 431 is further capable of providing multiple MPCs/policies, for example, the same or different MPCs/policies in accordance with protecting ones of multiple executables within a zip file, or for providing initial MPCs/policies and then further MPCs/policies or MPC/policy "modules" as initiated by further indicators such as given above, via an indicator of an already downloaded MPC/policy or via other suitable mechanisms. (It will be appreciated that pre-constructed MPCs/policies or other processing can also be utilized, e.g. via retrieval from storage 404, but with a potential decrease in flexibility.)

MPC generator 432 and protection policy generator 433 are further configurable. Thus, for example, more generic MPCs/policies can be provided to all or a grouping of serviced destination-devices (e.g. in accordance with a similarly configured/administered intranet), or different MPCs/policies that can be configured in accordance with one or more of user, network administration, Downloadable-destination or other parameters (e.g.

see FIG. 6). As will become apparent, a resulting MPC provides an operational interface to a destination device/process. Thus, a high degree of flexibility and efficiency is enabled in providing such an operational interface within different or differently configurable user devices/processes or other constraints.

Such configurability further enables particular policies to be utilized in accordance with a particular application (e.g. particular system uses, access limitations, user interaction, treating application programs or Java components from a particular known source one way and unknown source ActiveX components, or other considerations). Agent generator 431 further transfers a resulting MPC and protection policy pair to linking engine 405.

Linking engine 405 provides for forming from received component elements (see above) a sandboxed package that can include one or more initial or complete MPCs and applicable protection policies, and a Downloadable, such that the sandboxed package will protect a receiving Downloadable-destination from malicious operation by the Downloadable. Linking engine 405 is implementable in a static or configurable manner in accordance, for example, with characteristics of a particular user device/process stored intermittently or more persistently in storage 404. Linking engine 405 can also provide for restoring a Downloadable, such as a compressed, encrypted or otherwise encoded file that has been decompressed, decrypted or otherwise decoded via detection processing (e.g. see FIG. 6b).

It is discovered, for example, that the manner in which the Windows OS initiates a binary executable or an ActiveX control can be utilized to enable protected initiation of a detected-Downloadable. Linking engine 405 is, for example, configurable to form, for

an ordinary single-executable Downloadable (e.g. an application program, applet, etc.) a sandboxed package 340 as a concatenation of ordered elements including an MPC 341, applicable policies 342 and the Downloadable or "XEQ" 343 (e.g. see FIG. 4).

Linking engine 405 is also configurable to form, for a Downloadable received by a server as a compressed single or multiple-executable Downloadable such as a zipped or meta file, a protecting package 340 including one or more MPCs, applicable policies and the one or more included executables of the Downloadable. For example, a sandboxed package can be formed in which a single MPC and policies precede and thus will affect all such executables as a result of inflating and installation. An MPC and applicable policies can also, for example, precede each executable, such that each executable will be separately sandboxed in the same or a different manner according to MPC/policy configuration (see above) upon inflation and installation. (See also FIGS. 5 and 6)

Linking engine is also configurable to form an initial MPC, MPC-policy or sandboxed package (e.g. prior to upon receipt of a downloadable) or an additional MPC, MPC-policy or sandboxed package (e.g. upon or following receipt of a downloadable), such that suitable MPCs/policies can be provided to a Downloadable-destination or other destination in a more distributed manner. In this way, requisite bandwidth or destination resources can be minimized (via two or more smaller packages) in compromise with latency or other considerations raised by the additional required communication.

A configurable linking engine can also be utilized in accordance with other requirements of particular devices/processes, further or different elements or other permutations in accordance with the teachings herein. (It might, for example be desirable to modify the ordering of elements, to provide one or more elements separately, to

provide additional information, such as a header, etc., or perform other processing in accordance with a particular device, protocol or other application considerations.)

Policy/authentication reader-analyzer 481 summarily depicts other protection mechanisms that might be utilized in conjunction with Downloadable detection, such as already discussed, and that can further be configurable to operate in accordance with policies or parameters (summarily depicted by security/authentication policies 482). Integration of such further protection in the depicted configuration, for example, enables a potential-Downloadable from a known unfriendly source, a source failing authentication or a provided-source that is confirmed to be fictitious to be summarily discarded, otherwise blocked, flagged, etc. (with or without further processing). Conversely, a potential-Downloadable from a known friendly source (or one confirmed as such) can be transferred with or without further processing in accordance with particular application considerations. (Other configurations including pre or post Downloadable detection mechanisms might also be utilized.)

Finally, transfer engine 406 of protection agent engine 303 provides for receiving and causing linking engine 405 (or other protection) results to be transferred to a destination user device/process. As depicted, transfer engine 406 is configured to receive and transfer a Downloadable, a determined non-executable or a sandboxed package. However, transfer engine 406 can also be provided in a more configurable manner, such as was already discussed for other system 400 elements. (Any one or more of system 400 elements might be configurably implemented in accordance with a particular application.) Transfer engine 406 can perform such transfer, for example, by adding the information to a server transfer queue (not shown) or utilizing another suitable method.

Turning to FIG. 5 with reference to FIG. 4, a code detector 421 example is illustrated in accordance with an embodiment of the invention. As shown, code detector 421 includes data fetcher 501, parser 502, file-type detector 503, inflator 504 and control 506; other depicted elements. While implementable and potentially useful in certain instances, are found to require substantial overhead, to be less accurate in certain instances (see above) and are not utilized in a present implementation; these will be discussed separately below. Code detector elements are further configurable in accordance with stored parameters retrievable by data fetcher 501. (A coupling between data fetcher 501 and control 506 has been removed for clarity sake.)

Data fetcher 501 provides for retrieving a potential-Downloadable or portions thereof stored in buffer 407 or parameters from storage 404, and communicates such information or parameters to parser 502. Parser 502 receives a potential-Downloadable or portions thereof from data fetcher 501 and isolates potential-Downloadable elements, such as file headers, source, destination, certificates, etc. for use by further processing elements.

File type detector 502 receives and determines whether the potential-Downloadable (likely) is or includes an executable file type. File-reader 502 can, for example, be configured to analyze a received potential-Downloadable for a file header, which is typically included in accordance with conventional data transfer protocols, such as a portable executable or standard ".exe" file format for Windows OS application programs, a Java class header for Java applets, and so on for other applications, distributed components, etc. "Zipped", meta or other compressed files, which might include one or more executables, also typically provide standard single or multi-level

headers that can be read and used to identify included executable code (or other included information types). File type detector 502 is also configurable for analyzing potential-Downloadables for all potential file type delimiters or a more limited subset of potential file type delimiters (e.g. ".exe" or ".com" in conjunction with a DOS or Microsoft Windows OS Downloadable-destination).

Known file type delimiters can, for example, be stored in a more temporary or more persistent storage (e.g. storage 404 of FIG. 4) which file type detector 502 can compare to a received potential-Downloadable. (Such delimiters can thus also be updated in storage 404 as a new file type delimiter is provided, or a more limited subset of delimiters can also be utilized in accordance with a particular Downloadable-destination or other considerations of a particular application.) File type detector 502 further transfers to controller 506 a detected file type indicator indicating that the potential-Downloadable includes or does not include (i.e. or likely include) an executable file type.

In this example, the aforementioned detection processor is also included as pre-detection processor or, more particularly, a configurable file inflator 504. File inflator 504 provides for opening or "inflating" compressed files in accordance with a compressed file type received from file type detector 503 and corresponding file opening parameters received from data fetcher 501. Where a compressed file (e.g. a meta file) includes nested file type information not otherwise reliably provided in an overall file header or other information, inflator 504 returns such information to parser 502. File inflator 504 also provides any now-accessible included executables to control 506 where one or more included files are to be separately packaged with an MPC or policies.

Control 506, in this example, operates in accordance with stored parameters and provides for routing detected non-Downloadables or Downloadables and control information, and for conducting the aforementioned distributed downloading of packages to Downloadable-destinations. In the case of a non-Downloadable, for example, control 506 sends the non-Downloadable to transfer engine 406 (FIG. 4) along with any indicators that might apply. For an ordinary single-executable Downloadable, control 506 sends control information to agent generator 431 and the Downloadable to linking engine 405 along with any other applicable indicators (see 641 of FIG. 6b). Control 506 similarly handles a compressed single-executable Downloadable or a multiple downloadable to be protected using a single sandboxed package. For a multiple-executable Downloadable, control 506 sends control information for each corresponding executable to agent generator agent generator 431, and sends the executable to linking engine 405 along with controls and any applicable indicators, as in 643b of FIG. 6b. (The above assumes, however, that distributed downloading is not utilized; when used – according to applicable parameters- control 506 also operates in accordance with the following.)

Control 506 conducts distributed protection (e.g. distributed packaging) by providing control signals to agent generator 431, linking engine 405 and transfer engine 406. In the present example, control 506 initially sends controls to agent generator 431 and linking engine 405 (FIG. 4) causing agent generator to generate an initial MPC and initial policies, and sends control and a detected-Downloadable to linking engine 405. Linking engine 405 forms an initial sandboxed package, which transfer engine causes (in conjunction with further controls) to be downloaded to the Downloadable destination

(643a of FIG. 6b). An initial MPC within the sandboxed package includes an installer and a communicator and performs installation as indicated below. The initial MPC also communicates via the communicator controls to control 506 (FIG. 5) in response to which control 506 similarly causes generation of MPC-M and policy-M modules 643c, which linking engine 405 links and transfer engine 406 causes to be sent to the Downloadable destination, and so on for any further such modules.

(It will be appreciated, however, that an initial package might be otherwise configured or sent prior to receipt of a Downloadable in accordance with configuration parameters or user interaction. Information can also be sent to other user devices, such as that of an administrator. Further MPCs/policies might also be coordinated by control 506 or other elements, or other suitable mechanisms might be utilized in accordance with the teachings herein.)

Regarding the remaining detection engine elements illustrated in FIG. 5, where content analysis is utilized, parser 502 can also provide a Downloadable or portions thereof to content detector 505. Content detector 505 can then provide one or more content analyses. Binary detector 551, for example, performs detection of binary information; pattern detector 552 further analyzes the Downloadable for patterns indicating executable code, or other detectors can also be utilized. Analysis results therefrom can be used in an absolute manner, where a first testing result indicating executable code confirms Downloadable detection, which result is then sent to control 506. Alternatively, however, composite results from such analyses can also be sent to control 506 for evaluation. Control 506 can further conduct such evaluation in a summary manner (determining whether a Downloadable is detected according to a

majority or minimum number of indicators), or based on a weighting of different analysis results. Operation then continues as indicated above. (Such analysis can also be conducted in accordance with aspects of a destination user device or other parameters.)

FIG. 6a illustrates more specific examples of indicators/parameters and known (or "knowledge base") elements that can be utilized to facilitate the above-discussed system 400 configurability and detection. For clarity sake, indicators, parameters and knowledge base elements are combined as indicated "parameters." It will be appreciated, however, that the particular parameters utilized can differ in accordance with a particular application, and indicators, parameters or known elements, where utilized, can vary and need not correspond exactly with one another. Any suitable explicit or referencing list, database or other storage structure(s) or storage structure configuration(s) can also be utilized to implement a suitable user/device based protection scheme, such as in the above examples, or other desired protection schema.

Executable parameters 601 comprise, in accordance with the above examples, executable file type parameters 611, executable code parameters 612 and code pattern parameters 613 (including known executable file type indicators, header/code indicators and patterns respectively, where code patterns are utilized). Use parameters 602 further comprise user parameters 621, system parameters 622 and general parameters 623 corresponding to one or more users, user classifications, user-system correspondences or destination system, device or processes, etc. (e.g. for generating corresponding MPCs/policies, providing other protection, etc.). The remaining parameters include interface parameters 631 for providing MPC/policy (or further) configurability in

accordance with a particular device or for enabling communication with a device user (see below), and other parameters 632.

FIG. 6b illustrates a linking engine 405 according to an embodiment of the invention. As already discussed, linking engine 405 includes a linker for combining MPCs, policies or agents via concatination or other suitable processing in accordance with an OS, JVM or other host executor or other applicable factors that might apply. Linking engine 405 also includes the aforementioned post-detection processor which, in this example, comprises a compressor 508. As noted, compressor 508 receives linked elements from linker 507 and, where a potential-Downloadable corresponds to a compressed file that was inflated during detection, re-forms the compressed file. (Known file information can be provided via configuration parameters, substantially reversal of inflating or another suitable method.) Encryption or other post-detection processing can also be conducted by linking engine 508.

FIGS. 7a, 7b and 8 illustrate a "sandbox protection" system, as operable within a receiving destination-device, according to an embodiment of the invention.

Beginning with FIG. 7a, a client 146 receiving sandbox package 340 will "recognize" sandbox package 340 as a (mobile) executable and cause a mobile code installer 711 (e.g. an OS loader, JVM, etc.) to be initiated. Mobile code installer 711 will also recognize sandbox package 340 as an executable and will attempt to initiate sandbox package 340 at its "beginning." Protection engine 400 processing corresponding to destination 700 use of a such a loader, however, will have resulted in the "beginning" of sandbox package 340 as corresponding to the beginning of MPC 341, as noted with regard to the above FIG. 4 example.

Such protection engine processing will therefore cause a mobile code installer (e.g. OS loader 711, for clarity sake) to initiate MPC 341. In other cases, other processing might also be utilized for causing such initiation or further protection system operation. Protection engine processing also enables MPC 341 to effectively form a protection "sandbox" around Downloadable (e.g. detected-Downloadable or "XEQ") 343, to monitor Downloadable 343, intercept determinable Downloadable 343 operation (such as attempted accesses of Downloadable 343 to destination resources) and, if "malicious", to cause one or more other operations to occur (e.g. providing an alert, offloading the Downloadable, offloading the MPC, providing only limited resource access, possibly in a particular address space or with regard to a particularly "safe" resource or resource operation, etc.).

MPC 341, in the present OS example, executes MPC element installation and installs any policies, causing MPC 341 and protection policies 342 to be loaded into a first memory space, P1. MPC 341 then initiates loading of Downloadable 343. Such Downloadable initiation causes OS loader 711 to load Downloadable 343 into a further working memory space-P2 703 along with an API import table ("IAT") 731 for providing Downloadable 631 with destination resource access capabilities. It is discovered, however that the IAT can be modified so that any call to an API can be redirected to a function within the MPC. The technique for modifying the IAT is documented within the MSDN (Microsoft Developers Network) Library CD in several articles. The technique is also different for each operating system (e.g. between Windows 9x and Windows NT), which can be accommodated by agent generator configurability, such as that given above.

MPC 341 therefore has at least initial access to API IAT 731 of Downloadable 632, and provides for diverting, evaluating and responding to attempts by Downloadable 632 to utilize system APIs 731, or further in accordance with protection policies 342.

In addition to API diverting, MPC 341 can also install filter drivers, which can be used for controlling access to resources such as a Downloadable-destination file system or registry. Filter driver installation can be conducted as documented in the MSDN or using other suitable methods.

Turning to FIG. 8 with reference to FIG. 7b, an MPC 341 according to an embodiment of the invention includes a package extractor 801, executable installer 802, sandbox engine installer 803, resource access diverter 804, resource access (attempt) analyzer 805, policy enforcer 806 and MPC de-installer 807. Package extractor 801 is initiated upon initiation of MPC 341, and extracts MPC 341 elements and protection policies 342. Executable installer 802 further initiates installation of a Downloadable by extracting the downloadable from the protected package, and loading the process into memory in suspended mode (so it only loads into memory, but does not start to run). Such installation further causes the operating system to initialize the Downloadable's IAT 731 in the memory space of the downloadable process, P2, as already noted.

Sandbox engine installer 803 (running in process space P1) then installs the sandbox engine (803-805) and policies 342 into the downloadable process space P2. This is done in different way in each operating system (e.g. see above). Resource access diverter 804 further modifies those Downloadable-API IAT entries that correspond with protection policies 342, thereby causing corresponding Downloadable accesses via Downloadable-API IAT 731 to be diverted resource access analyzer 805.

During Downloadable operation, resource access analyzer or "RAA" 805 receives and determines a response to diverted Downloadable (i.e. "malicious") operations in accordance with corresponding protection policies of policies 342. (RAA 805 or further elements, which are not shown, can further similarly provide for other security mechanisms that might also be implemented.) Malicious operations can for example include, in a Windows environment: file operations (e.g. reading, writing, deleting or renaming a file), network operations (e.g. listen on or connect to a socket, send/receive data or view intranet), OS registry or similar operations (read/write a registry item), OS operations (exit OS/client, kill or change the priority of a process/thread, dynamically load a class library), resource usage thresholds (e.g. memory, CPU, graphics), etc.

Policy enforcer 806 receives RAA 805 results and causes a corresponding response to be implemented, again according to the corresponding policies. Policy enforcer 806 can, for example, interact with a user (e.g. provide an alert, receive instructions, etc.), create a log file, respond, cause a response to be transferred to the Downloadable using "dummy" or limited data, communicate with a server or other networked device (e.g. corresponding to a local or remote administrator), respond more specifically with a better known Downloadable, verify accessibility or user/system information (e.g. via local or remote information), even enable the attempted Downloadable access, among a wide variety of responses that will become apparent in view of the teachings herein.

The FIG. 9 flowchart illustrates a protection method according to an embodiment of the invention. In step 901, a protection engine monitors the receipt, by a server or other re-communicator of information, and receives such information intended for a

protected information-destination (i.e. a potential-Downloadable) in step 903. Steps 905-911 depict an adjunct trustworthiness protection that can also be provided, wherein the protection engine determines whether the source of the received information is known to be "unfriendly" and, if so, prevents current (at least unaltered) delivery of the potential-Downloadable and provides any suitable alerts. (The protection engine might also continue to perform Downloadable detection and nevertheless enable delivery or protected delivery of a non-Downloadable, or avoid detection if the source is found to be "trusted", among other alternatives enabled by the teachings herein.)

If, in step 913, the potential-Downloadable source is found to be of an unknown or otherwise suitably authenticated/certified source, then the protection engine determines whether the potential-Downloadable includes executable code in step 915. If the potential-Downloadable does not include executable code, then the protection engine causes the potential-Downloadable to be delivered to the information-destination in its original form in step 917, and the method ends. If instead the potential-Downloadable is found to include executable code in step 915 (and is thus a "detected-Downloadable"), then the protection engine forms a sandboxed package in step 919 and causes the protection agent to be delivered to the information-Destination in step 921, and the method ends. As was discussed earlier, a suitable protection agent can include mobile protection code, policies and the detected-Downloadable (or information corresponding thereto).

The FIG. 10a flowchart illustrates a method for analyzing a potential-Downloadable, according to an embodiment of the invention. As shown, one or more aspects can provide useful indicators of the inclusion of executable code within the

potential-Downloadable. In step 1001, the protection engine determines whether the potential-Downloadable indicates an executable file type, for example, by comparing one or more included file headers for file type indicators (e.g. extensions or other descriptors). The indicators can be compared against all known file types executable by all protected Downloadable destinations, a subset, in accordance with file types executable or desirably executable by the Downloadable-destination, in conjunction with a particular user, in conjunction with available information or operability at the destination, various combinations, etc.

Where content analysis is conducted, in step 1003 of FIG. 10a, the protection engine analyzes the potential-Downloadable and determines in accordance therewith whether the potential-Downloadable does or is likely to include binary information, which typically indicates executable code. The protection engine further analyzes the potential-Downloadable for patterns indicative of included executable code in step 1003. Finally, in step 1005, the protection engine determines whether the results of steps 1001 and 1003 indicate that the potential-Downloadable more likely includes executable code (e.g. via weighted comparison of the results with a suitable level indicating the inclusion or exclusion of executable code). The protection engine, given a suitably high confidence indicator of the inclusion of executable code, treats the potential-Downloadable as a detected-Downloadable.

The FIG. 10b flowchart illustrates a method for forming a sandboxed package according to an embodiment of the invention. As shown, in step 1011, a protection engine retrieves protection parameters and forms mobile protection code according to the parameters. The protection engine further, in step 1013, retrieves protection parameters

and forms protection policies according to the parameters. Finally, in step 1015, the protection engine couples the mobile protection code, protection policies and received-information to form a sandboxed package. For example, where a Downloadable-destination utilizes a standard windows executable, coupling can further be accomplished via concatenating the MPC for delivery of MPC first, policies second, and received information third. (The protection parameters can, for example, include parameters relating to one or more of the Downloadable destination device/process, user, supervisory constraints or other parameters.)

The FIG. 11 flowchart illustrates how a protection method performed by mobile protection code ("MPC") according to an embodiment of the invention includes the MPC installing MPC elements and policies within a destination device in step 1101. In step 1102, the MPC loads the Downloadable without actually initiating it (i.e. for executables, it will start a process in suspended mode). The MPC further forms an access monitor or "interceptor" for monitoring or "intercepting" downloadable destination device access attempts within the destination device (according to the protection policies in step 1103, and initiates a corresponding Downloadable within the destination device in step 1105.

If, in step 1107, the MPC determines, from monitored/intercepted information, that the Downloadable is attempting or has attempted a destination device access considered undesirable or otherwise malicious, then the MPC performs steps 1109 and 1111; otherwise the MPC returns to step 1107. In step 1109, the MPC determines protection policies in accordance with the access attempt by the Downloadable, and in step 1111, the MPC executes the protection policies. (Protection policies can, for example, be retrieved from a temporary, e.g. memory/cache, or more persistent storage.)

As shown in the FIG. 12a example, the MPC can provide for intercepting Downloadable access attempts by a Downloadable by installing the Downloadable (but not executing it) in step 1201. Such installation will cause a Downloadable executor, such as a the Windows operating system, to provide all required interfaces and parameters (such as the IAT, process ID, etc.) for use by the Downloadable to access device resources of the host device. The MPC can thus cause Downloadable access attempts to be diverted to the MPC by modifying the Downloadable IAT, replacing device resource location indicators with those of the MPC (step 1203).

The FIG. 12b example further illustrates an example of how the MPC can apply suitable policies in accordance with an access attempt by a Downloadable. As shown, the MPC receives the Downloadable access request via the modified IAT in step 1211. The MPC further queries stored policies to determine a policy corresponding to the Downloadable access request in step 1213.

The foregoing description of preferred embodiments of the invention is provided by way of example to enable a person skilled in the art to make and use the invention, and in the context of particular applications and requirements thereof. Various modifications to the embodiments will be readily apparent to those skilled in the art, and the generic principles defined herein may be applied to other embodiments and applications without departing from the spirit and scope of the invention. Thus, the present invention is not intended to be limited to the embodiments shown, but is to be accorded the widest scope consistent with the principles, features and teachings disclosed herein. The embodiments described herein are not intended to be exhaustive or limiting. The present invention is limited only by the following claims.

WHAT IS CLAIMED IS:

1. A method, comprising:

receiving downloadable-information;

determining whether the downloadable-information includes executable code; and

causing mobile protection code to be communicated to at least one information-destination of the downloadable-information, if the downloadable-information is determined to include executable code.

2. The method of claim 1, wherein the receiving includes monitoring received information of an information re-communicator.

3. The method of claim 2, wherein the information re-communicator is a network server.

4. The method of claim 1, wherein the determining comprises analyzing the downloadable-information for an included type indicator indicating an executable file type.

5. The method of claim 1, wherein the determining comprises analyzing the downloadable-information for an included an included type detector indicating an archive file that contains at least one executable.

6. The method of claim 1, wherein the determining comprises analyzing the downloadable-information for an included file type indicator and an information pattern

corresponding to one or more information patterns that tend to be included within executable code.

7. The method of claim 1, further comprising receiving one or more executable code characteristics of executable code that is capable of being executed by the information-destination, and wherein the determining is conducted in accordance with the executable code characteristics.

8. The method of claim 1, wherein the determining comprises performing one or more analyses of the downloadable-information, the analyses producing detection-indicators indicating whether a correspondence is detected between a downloadable-information characteristic and at least one respective executable code characteristic, and evaluating the detection-indicators to determine whether the downloadable-information includes executable code.

9. The method of claim 8, wherein at least one of the detection-indicators indicates a level of downloadable-information characteristic and executable code characteristic correspondence.

10. The method of claim 8, wherein the evaluating includes assigning a weighted level of importance to at least one of the indicators.

11. The method of claim 1, wherein the causing mobile protection code to be

communicated comprises forming a sandboxed package including the mobile protection code and the downloadable-information, and causing the sandboxed package to be communicated to the at least one information-destination.

12. The method of claim 10, wherein the sandboxed package is formed such that the mobile protection code will be executed by the information-destination before the downloadable-information.

13. The method of claim 11, wherein the sandboxed package further includes protection policies according to which the mobile protection code is operable.

14. The method of claim 13, wherein the sandboxed package is formed for receipt by the information-destination such that the mobile protection code is received before the downloadable-information, and the downloadable information before the protection policies.

15. The method of claim 13, wherein the protection policies correspond with at least one of the information-destination and a user of the information destination.

16. A system, comprising:

an information monitor for receiving downloadable-information;

a content inspection engine communicatively coupled to the information monitor for determining whether the downloadable-information includes executable code; and

a protection agent engine communicatively coupled to the content inspection engine for causing mobile protection code ("MPC") to be communicated to at least one information-destination of the downloadable-information, if the downloadable-information is determined to include executable code.

17. The system of claim 16, wherein the information monitor intercepts received information received by an information re-communicator.

18. The system of claim 17, wherein the information re-communicator is a network server.

19. The system of claim 16, wherein the content inspection engine comprises a file type detector for determining whether the downloadable-information includes a file type indicator indicating an executable file type.

20. The system of claim 16, wherein the content inspection engine comprises a parser for parsing the downloadable-information and a content analyzer communicatively coupled to the parser for determining whether one or more downloadable-information elements of the downloadable-information correspond with executable code elements are executable code elements.

21. The system of claim 16, wherein the content inspection engine comprises one or more downloadable-information analyzers for analyzing the downloadable-information,

each analyzer producing therefrom a detection indicator indicating whether a downloadable-information characteristic corresponds with an executable code characteristic, and an inspection controller communicatively coupled to the analyzers for determining whether the indicators indicate that the downloadable-information includes executable code.

22. The system of claim 21, wherein at least one of the detection-indicators indicates a level of downloadable-information characteristic and executable code characteristic correspondence.

23. The system of claim 21, wherein the evaluating includes assigning a weighted level of importance to at least one of the detection-indicators.

24. The system of claim 16, wherein the sandboxed package engine comprises an MPC generator for providing the MPC, a linking engine coupled to the MPC generator for forming a protection agent including the MPC and the downloadable-information, and a transfer engine for causing the protection agent to be communicated to the at least one information-destination.

25. The system of claim 24, wherein the protection agent engine further comprises a policy generator communicatively coupled to the linking engine for providing protection policies according to which the MPC is operable.

26. The system of claim 25, wherein the sandboxed package is formed for receipt by the information-destination such that the mobile protection code is executed before the downloadable-information.

27. The system of claim 26, wherein the protection policies correspond with policies of at least one of the information-destination and a user of the information destination.

28. A system, comprising:

means for receiving downloadable-information;

means for determining whether the downloadable-information includes executable code; and

means for causing mobile protection code to be communicated to at least one information-destination of the downloadable-information, if the downloadable-information is determined to include executable code.

29. A computer-readable storage medium storing program code for causing a computer to perform the steps of:

receiving downloadable-information;

determining whether the downloadable-information includes executable code; and

causing mobile protection code to be communicated to at least one information-destination of the downloadable-information, if the downloadable-information is determined to include executable code.

30. A method, comprising:

receiving, at an information re-communicator, downloadable-information, including executable code; and

causing mobile protection code to be executed by a mobile code executor at a downloadable-information destination such that one or more operations of the executable code at the destination, if attempted, will be processed by the mobile protection code.

31. The method of claim 30, wherein the mobile code executor is a Java Virtual Machine.

32. The method of claim 30, wherein the mobile code executor is the operating system, running native code executables.

33. The method of claim 30, wherein the mobile code executor is ActiveX subsystem of the windows operating system

34. The method of claim 30, wherein the mobile code executor is the Microsoft Windows scripting host

35. The method of claim 30, wherein the causing is accomplished by forming a sandboxed package including the mobile protection code and the downloadable-information, and causing the sandboxed package to be delivered to the downloadable-information destination.

36. The method of claim 35, wherein the sandboxed package further includes protection policies according to which the processing by the mobile protection code is conducted.

37. A sandboxed package formed according to the method of claim 35.

38. A sandboxed package formed according to the method of claim 36.

39. The method of claim 36, wherein the forming comprises generating the mobile protection code, generating the sandboxed package, and linking the mobile protection code, protection policies and downloadable-information.

40. The method of claim 39, wherein the generating of at least one of the mobile protection code and the protection policies is conducted in accordance with one or more destination-characteristics of the destination.

41. The method of claim 40, wherein the destination-characteristics include characteristics corresponding to at least one of a destination user, a destination device and a destination process.

42. The method of claim 35, wherein the causing the sandboxed package to be executed includes communicating the sandboxed package to a communication buffer of the information re-communicator.

43. The method of claim 30, wherein the re-communicator is at least one of a firewall and a network server.

44. The method of claim 30, wherein the sandboxed package has a same file type as the downloadable-information, thereby causing the mobile code executor to be unaware that the protected package is not a normal downloadable.

45. The method of claim 44, wherein the sandboxed package is formed using concatenation of a mobile protection code, a policy, and a downloadable.

46. The method of claim 30, wherein executing the mobile protection code at the destination causes downloadable interfaces to resources at the destination to be modified such that at least one attempted operation of the executable code is diverted to the mobile protection code.

47. A system, comprising:

receiving means for receiving, at an information re-communicator, downloadable-information, including executable code; and

mobile code means communicatively coupled to the receiving means for causing mobile protection code to be executed by a mobile code executor at a downloadable-information destination such that one or more operations of the executable code at the destination, if attempted, will be processed by the mobile protection code.

48. The system of claim 47, wherein the mobile code executor is a Java Virtual Machine.

49. The system of claim 47, wherein the mobile code executor is an operating system, running native code executables.

50. The system of claim 47, wherein the mobile code executor is an ActiveX subsystem of the windows operating system.

51. The system of claim 47, wherein the mobile code executor is a Microsoft Windows scripting host.

52. The system of claim 47, wherein the causing is accomplished by forming a sandboxed package including the mobile protection code and the downloadable-information, and causing the sandboxed package to be delivered to the downloadable-information destination.

53. The system of claim 52, wherein the sandboxed package further includes protection policies according to which the processing by the mobile protection code is conducted.

54. The system of claim 53, wherein the forming comprises generating the mobile protection code, generating the protection policies, and linking the mobile protection code, protection policies and downloadable-information.

55. The system of claim 54, wherein the generating of at least one of the mobile protection code and the protection policies is conducted in accordance with one or more destination-characteristics of the destination.

56. The system of claim 55, wherein the destination-characteristics include characteristics corresponding to at least one of a destination user, a destination device and a destination process.

57. The system of claim 46, wherein the causing the sandboxed package to be executed includes communicating the sandboxed package to a communication buffer of the information re-communicator.

58. The system of claim 47, wherein the re-communicator is at least one of a firewall and a network server.

59. The system of claim 47, wherein executing the mobile protection code at the destination causes downloadable interfaces a resource at the destination to be modified such that at least one attempted operation of the executable code is diverted to the mobile protection code.

60. A computer-readable storage medium storing program code for causing a computer to perform the steps of:

receiving, at an information re-communicator, downloadable-information, including executable code; and

causing mobile protection code to be executed by a mobile code executor at a downloadable-information destination such that one or more operations of the executable code at the destination, if attempted, will be processed by the mobile protection code.

61. A method, comprising:

receiving mobile protection code ("MPC") and a Downloadable at a Downloadable-destination;

causing, by the MPC, one or more operations attempted by the Downloadable to be received by the MPC;

receiving, by the MPC, an attempted operation of the Downloadable; and

initiating, by the MPC, a protection policy corresponding to the attempted operation.

62. The method of claim 61, wherein the receiving comprises receiving a sandboxed package that includes the MPC, the Downloadable and one or more protection policies.

63. The method of claim 62, wherein the sandboxed package is configured such that the MPC is executed first, the Downloadable is executed by the MPC and the protection policies are accessible to the MPC.

64. The method of claim 61, wherein the causing comprises modifying, by the MPC,

interfaces of a corresponding downloadable to resources at the destination.

65. The method of claim 64, wherein the modifying is accomplished by initiating a loading of the Downloadable, thereby causing a mobile code executor to provide and initialize the interfaces, modifying one or more interface elements to divert corresponding attempted Downloadable operations to the MPC, and initiating execution of the Downloadable.

66. The method of claim 64, wherein the interfaces comprise an import address table ("IAT") of a native code executable downloadable.

67. The method of claim 64, wherein modifying the interfaces installs a filter-driver between the downloadable and the resources.

68. A system, comprising:

a mobile code executer for initiating received mobile code; and

a sandboxed package capable of being received and initiated by the mobile code executer, the sandboxed package including a Downloadable and mobile protection code ("MPC") for causing one or more Downloadable operations to be intercepted and for processing the intercepted operations, if the Downloadable attempts to initiate the operations.

69. The system of claim 60, wherein the MPC comprises:

an MPC installer for causing MPC elements to be installed;

a Downloadable installer communicatively coupled to the MPC element installer for installing the Downloadable;

a resource access diverter communicatively coupled to the MPC installer for causing the Downloadable operations to be intercepted;

a resource access analyzer communicatively coupled to the MPC installer for receiving an intercepted Downloadable operation and determining a protection policy corresponding to the intercepted Downloadable operation; and

a policy enforcer communicatively coupled to the resource access analyzer for processing the intercepted Downloadable operation.

70. The system of claim 69, wherein the resource access diverter modifies one or more elements of an interface usable by the Downloadable to effectuate the Downloadable operations.

71. The system of claim 69, wherein the mobile code executer is a Java Virtual Machine.

72. The system of claim 69, wherein the mobile code executor is an operating system, running native code executables.

73. The system of claim 69, wherein the mobile code executor is an ActiveX subsystem of the windows operating system.

74. The system of claim 69, wherein the mobile code executor is an Microsoft Windows scripting host.

75. A system, comprising

receiving means for receiving mobile protection code ("MPC") and a Downloadable at a Downloadable-destination;

monitoring means for causing, by the MPC, one or more operations attempted by the Downloadable to be received by the MPC;

second receiving means receiving, by the MPC, an attempted operation of the Downloadable; and

initiating means for initiating, by the MPC, a protection policy corresponding to the attempted operation.

76. A computer-readable storage medium storing program code for causing a computer to perform the steps of:

receiving mobile protection code ("MPC") and a Downloadable at a Downloadable-destination;

causing, by the MPC, one or more operations attempted by the Downloadable to be received by the MPC;

receiving, by the MPC, an attempted operation of the Downloadable; and

initiating, by the MPC, a protection policy corresponding to the attempted operation.

ABSTRACT OF THE DISCLOSURE

MALICIOUS MOBILE CODE RUNTIME MONITORING

SYSTEM AND METHODS

Protection systems and methods provide for protecting one or more personal computers ("PCs") and/or other intermittently or persistently network accessible devices or processes from undesirable or otherwise malicious operations of Java™ applets, ActiveX™ controls, JavaScript™ scripts, Visual Basic scripts, add-ins, downloaded/ uploaded programs or other "Downloadables" or "mobile code" in whole or part. A protection engine embodiment provides, within a server, firewall or other suitable "re-communicator," for monitoring information received by the communicator, determining whether received information does or is likely to include executable code, and if so, causes mobile protection code (MPC) to be transferred to and rendered operable within a destination device of the received information, more suitably by forming a protection agent including the MPC, protection policies and a detected-Downloadable. An MPC embodiment further provides, within a Downloadable-destination, for initiating the Downloadable, enabling malicious Downloadable operation attempts to be received by the MPC, and causing (predetermined) corresponding operations to be executed in response to the attempts, more suitably in conjunction with protection policies.

UNITED STATES PATENT AND TRADEMARK OFFICE

COMMISSIONER FOR PATENTS
UNITED STATES PATENT AND TRADEMARK OFFICE
WASHINGTON, D.C. 20231
www.uspto.gov

Bib Data Sheet

CONFIRMATION NO. 5421

SERIAL NUMBER	FILING DATE	CLASS	GROUP ART UNIT	ATTORNEY DOCKET NO.
09/861,229	05/17/2001 RULE	709	2152	43426.00014

APPLICANTS

Yigal Mordechai Edery, Pardesia, ISRAEL;
CM Nimrod Itzhak Vered, Goosh Tel-Mond, ISRAEL;
David R. Kroll, San Jose, CA;

**** CONTINUING DATA ****************************

THIS APPLN CLAIMS BENEFIT OF 60/205,591 05/17/2000
CH AND A CIP OF 09/539,667 03/30/2000 which is now U.S. Patent 6,804,780
AND A CIP OF 09/551,302 04/18/2000 which is now U.S. Patent 6,480,962

**** FOREIGN APPLICATIONS **********************

CM

IF REQUIRED, FOREIGN FILING LICENSE GRANTED ** SMALL ENTITY **
** 07/18/2001

		STATE OR COUNTRY	SHEETS DRAWING	TOTAL CLAIMS	INDEPENDENT CLAIMS
Foreign Priority claimed	☐ yes ☑ no	ISRAEL	10	76	11
35 USC 119 (a-d) conditions met	☐ yes ☑ no ☐ Met after Allowance				
Verified and Acknowledged	Examiner's Signature / Initials				

ADDRESS

Intellectual Property Department
Squire, Sanders & Dempsey L.L.P.
600 Hansen Way
Palo Alto ,CA 94304-1043

TITLE

Malicious mobile code runtime monitoring system and methods

FILING FEE RECEIVED		
1244	FEES: Authority has been given in Paper No. _____ to charge/credit DEPOSIT ACCOUNT No. _____ for following:	☐ All Fees ☐ 1.16 Fees (Filing) ☐ 1.17 Fees (Processing Ext. of time) ☐ 1.18 Fees (Issue) ☐ Other _____ ☐ Credit

PATENT APPLICATION SERIAL NO. _____

U.S. DEPARTMENT OF COMMERCE
PATENT AND TRADEMARK OFFICE
FEE RECORD SHEET

05/23/2001 MBELETE1 00000028 050150 09861229

01 FC:201	355.00	CH
02 FC:202	320.00	CH
03 FC:203	504.00	CH

PTO-1556
(5/87)

*U.S. GPO: 2000-468-987/39595

PATENT APPLICATION FEE DETERMINATION RECORD
Effective October 1, 2000

Application or Docket Number
09/861229
~~43426.00012~~

CLAIMS AS FILED - PART I

	(Column 1)	(Column 2)
TOTAL CLAIMS	76	
FOR	NUMBER FILED	NUMBER EXTRA
TOTAL CHARGEABLE CLAIMS	76 minus 20=	* 56
INDEPENDENT CLAIMS	11 minus 3 =	* 8
MULTIPLE DEPENDENT CLAIM PRESENT		☐

* If the difference in column 1 is less than zero, enter "0" in column 2

SMALL ENTITY TYPE ☐		OR	OTHER THAN SMALL ENTITY	
RATE	FEE		RATE	FEE
BASIC FEE	355.00	OR	BASIC FEE	710.00
X$ 9=	504.00	OR	X$18=	
X40=	320.00	OR	X80=	
+135=		OR	+270=	
TOTAL	1179.00	OR	TOTAL	

CLAIMS AS AMENDED - PART II

AMENDMENT A

	(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA
Total	*	Minus	**	=
Independent	*	Minus	***	=
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM				☐

SMALL ENTITY		OR	OTHER THAN SMALL ENTITY	
RATE	ADDI-TIONAL FEE		RATE	ADDI-TIONAL FEE
X$ 9=		OR	X$18=	
X40=		OR	X80=	
+135=		OR	+270=	
TOTAL ADDIT. FEE		OR	TOTAL ADDIT. FEE	

AMENDMENT B

	(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA
Total	*	Minus	**	=
Independent	*	Minus	***	=
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM				☐

RATE	ADDI-TIONAL FEE		RATE	ADDI-TIONAL FEE
X$ 9=		OR	X$18=	
X40=		OR	X80=	
+135=		OR	+270=	
TOTAL ADDIT. FEE		OR	TOTAL ADDIT. FEE	

AMENDMENT C

	(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA
Total	*	Minus	**	=
Independent	*	Minus	***	=
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM				☐

RATE	ADDI-TIONAL FEE		RATE	ADDI-TIONAL FEE
X$ 9=		OR	X$18=	
X40=		OR	X80=	
+135=		OR	+270=	
TOTAL ADDIT. FEE		OR	TOTAL ADDIT. FEE	

* If the entry in column 1 is less than the entry in column 2, write "0" in column 3.
** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 20, enter "20."
***If the "Highest Number Previously Paid For" IN THIS SPACE is less than 3, enter "3."
The "Highest Number Previously Paid For" (Total or Independent) is the highest number found in the appropriate box in column 1.

CLAIMS ONLY

SERIAL NO. 09861229

FILING DATE 5/17/01

APPLICANT(S)

CLAIMS

	AS FILED		AFTER 1st AMENDMENT		AFTER 2nd AMENDMENT	
	IND.	DEP.	IND.	DEP.	IND.	DEP.
1	/					
2		/				
3		/				
4		/				
5		/				
6		/				
7		/				
8		/				
9		/				
10		/				
11		/				
12		/				
13		/				
14		/				
15		/				
16	/					
17		/				
18		/				
19		/				
20		/				
21		/				
22		/				
23		/				
24		/				
25		/				
26		/				
27		/				
28	/					
29	/					
30	/					
31		/				
32		/				
33		/				
34		/				
35		/				
36		/				
37		/				
38		/				
39		/				
40		/				
41		/				
42		/				
43		/				
44		/				
45		/				
46		/				
47	/					
48		/				
49		/				
50		/				
TOTAL IND.						
TOTAL DEP.						
TOTAL CLAIMS						

	*		*		*	
	IND.	DEP.	IND.	DEP.	IND.	DEP.
51		/				
52		/				
53		/				
54		/				
55		/				
56		/				
57		/				
58		/				
59		/				
60	/					
61	/					
62		/				
63		/				
64		/				
65		/				
66		/				
67		/				
68	/					
69		/				
70		/				
71		/				
72		/				
73		/				
74		/				
75	/					
76	/					
77						
78						
79						
80						
81						
82						
83						
84						
85						
86						
87						
88						
89						
90						
91						
92						
93						
94						
95						
96						
97						
98						
99						
100						
TOTAL IND.	11					
TOTAL DEP.	65					
TOTAL CLAIMS	76					

*MAY BE USED FOR ADDITIONAL CLAIMS OR ADMENDMENTS

U.S.DEPARTMENT OF COMMERCE
Patent and Trademark Office

FORM PTO-2022 (1-98)

*U.S. GPO: 1998-443-593/89152

UNITED STATES PATENT AND TRADEMARK OFFICE

COMMISSIONER FOR PATENTS
UNITED STATES PATENT AND TRADEMARK OFFICE
WASHINGTON, D.C. 20231
www.uspto.gov

APPLICATION NUMBER	FILING/RECEIPT DATE	FIRST NAMED APPLICANT	ATTORNEY DOCKET NUMBER
09/861,229	05/17/2001	Yigal Edery	43426.00014

CONFIRMATION NO. 5421

FORMALITIES LETTER

OC000000006314695

Intellectual Property Department
Squire, Sanders & Dempsey L.L.P.
600 Hansen Way
Palo Alto, CA 94304-1043

Date Mailed: 07/19/2001

NOTICE TO FILE MISSING PARTS OF NONPROVISIONAL APPLICATION

FILED UNDER 37 CFR 1.53(b)

Filing Date Granted

An application number and filing date have been accorded to this application. The item(s) indicated below, however, are missing. Applicant is given **TWO MONTHS** from the date of this Notice within which to file all required items and pay any fees required below to avoid abandonment. Extensions of time may be obtained by filing a petition accompanied by the extension fee under the provisions of 37 CFR 1.136(a).

- The oath or declaration is missing.
  A properly signed oath or declaration in compliance with 37 CFR 1.63, identifying the application by the above Application Number and Filing Date, is required.
- To avoid abandonment, a late filing fee or oath or declaration surcharge as set forth in 37 CFR 1.16(e) of $65 for a small entity in compliance with 37 CFR 1.27, must be submitted with the missing items identified in this letter.
- **The balance due by applicant is $ 65.**

The application is informal since it does not comply with the regulations for the reason(s) indicated below.

The required item(s) identified below must be timely submitted to avoid abandonment:

- Substitute drawings in compliance with 37 CFR 1.84 because:
  - drawing sheets do not have the appropriate margin(s) (see 37 CFR 1.84(g)). Each sheet must include a top margin of at least 2.5 cm. (1 inch), a left side margin of at least 2.5 cm. (1 inch), a right side margin of at least 1.5 cm. (5/8 inch), and a bottom margin of at least 1.0 cm. (3/8 inch);

A copy of this notice MUST be returned with the reply.

Customer Service Center
Initial Patent Examination Division (703) 308-1202

PART 3 - OFFICE COPY

Please type a plus sign (+) inside this box ⟶ [+]

PTO/SB/21 (08-00)
Approved for use through 10/31/2002. OMB 0651-0031
U.S. Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE

TRANSMITTAL FORM

(to be used for all correspondence after initial filing)

Application Number	09/861,229
Filing Date	May 17, 2001
First Named Inventor	Yigal Edery, et al.
Group Art Unit	2152
Examiner Name	Unknown
Total Number of Pages in This Submission: 27	Attorney Docket Number: 43426.00014

ENCLOSURES *(check all that apply)*

- [x] Fee Transmittal Form (in duplicate)
  - [ ] Fee Attached
  - [x] Deposit Account Authorization on Fee Transmittal Form
- [ ] Amendment / Response
  - [ ] After Final
  - [ ] Affidavits/declaration(s)
- [ ] Extension of Time Request
- [x] Return Postcard
- [ ] Information Disclosure Statement
- [ ] Certified Copy of Priority Document(s)
- [x] Response to Missing Parts/ Incomplete Application (in duplicate)
  - [x] Response to Missing Parts under 37 CFR 1.52 or 1.53
- [ ] Assignment Papers *(for an Application)*
- [x] Informal Drawings consisting of Figures 1a, 1b, 1c, 2, 3, 4, 5, 6a, 6b, 7a, 7b, 8, 9, 10a, 10b, 11, 12a, and 12b
- [ ] Licensing-related Papers
- [ ] Petition
- [ ] Petition to Convert to a Provisional Application
- [x] Combined Power of Attorney and Declaration for Patent Application
- [ ] Terminal Disclaimer
- [ ] Request for Refund
- [ ] CD, Number of CD(s) _____
- [ ] After Allowance Communication to Group
- [ ] Appeal Communication to Board of Appeals and Interferences
- [ ] Appeal Communication to Group *(Appeal Notice, Brief, Reply Brief)*
- [ ] Proprietary Information
- [ ] Status Letter
- [x] Other Enclosure(s) *(please identify below)*:
  Letter to the Official Draftsperson (Request to Substitute Drawings) (in duplicate)

Remarks

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT

Firm *or* Individual name	Daryl C. Josephson, Reg. No. 37,365 Squire, Sanders & Dempsey, L.L.P. 600 Hansen Way Palo Alto, CA 94304-1043
Signature	Daryl C. Josephson
Date	September 10, 2001

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Assistant Commissioner for Patents, Washington, D.C. 20231 on this date: September 10, 2001

Typed or printed name	Sandy Yi		
Signature	Sandy Yi	Date	September 10, 2001

UNITED STATES PATENT AND TRADEMARK OFFICE

COMMISSIONER FOR PATENTS
UNITED STATES PATENT AND TRADEMARK OFFICE
WASHINGTON, D.C. 20231
www.uspto.gov

APPLICATION NUMBER	FILING/RECEIPT DATE	FIRST NAMED APPLICANT	ATTORNEY DOCKET NUMBER
09/861,229	05/17/2001	Yigal Edery	43426.00014

CONFIRMATION NO. 5421

FORMALITIES LETTER

OC000000006314695

Intellectual Property Department
Squire, Sanders & Dempsey L.L.P.
600 Hansen Way
Palo Alto, CA 94304-1043

Date Mailed: 07/19/2001

NOTICE TO FILE MISSING PARTS OF NONPROVISIONAL APPLICATION

FILED UNDER 37 CFR 1.53(b)

Filing Date Granted

An application number and filing date have been accorded to this application. The item(s) indicated below, however, are missing. Applicant is given **TWO MONTHS** from the date of this Notice within which to file all required items and pay any fees required below to avoid abandonment. Extensions of time may be obtained by filing a petition accompanied by the extension fee under the provisions of 37 CFR 1.136(a).

- The oath or declaration is missing.
  A properly signed oath or declaration in compliance with 37 CFR 1.63, identifying the application by the above Application Number and Filing Date, is required.
- To avoid abandonment, a late filing fee or oath or declaration surcharge as set forth in 37 CFR 1.16(e) of $65 for a small entity in compliance with 37 CFR 1.27, must be submitted with the missing items identified in this letter.
- **The balance due by applicant is $ 65.**

The application is informal since it does not comply with the regulations for the reason(s) indicated below.

The required item(s) identified below must be timely submitted to avoid abandonment:

- Substitute drawings in compliance with 37 CFR 1.84 because:
  - drawing sheets do not have the appropriate margin(s) (see 37 CFR 1.84(g)). Each sheet must include a top margin of at least 2.5 cm. (1 inch), a left side margin of at least 2.5 cm. (1 inch), a right side margin of at least 1.5 cm. (5/8 inch), and a bottom margin of at least 1.0 cm. (3/8 inch);

A copy of this notice MUST be returned with the reply.

Customer Service Center
Initial Patent Examination Division (703) 308-1202
PART 2 - COPY TO BE RETURNED WITH RESPONSE

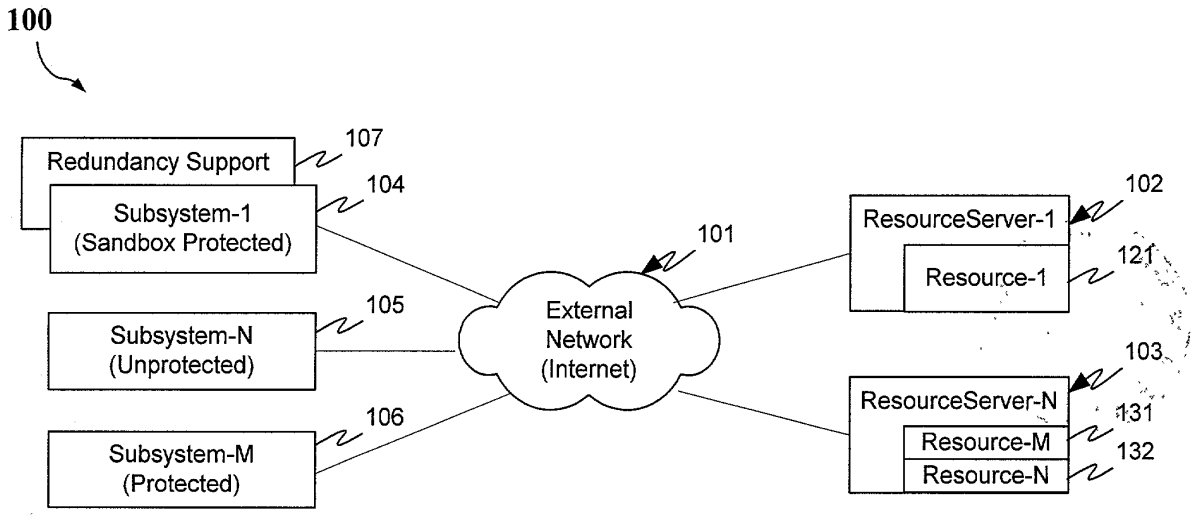

FIG. 1a

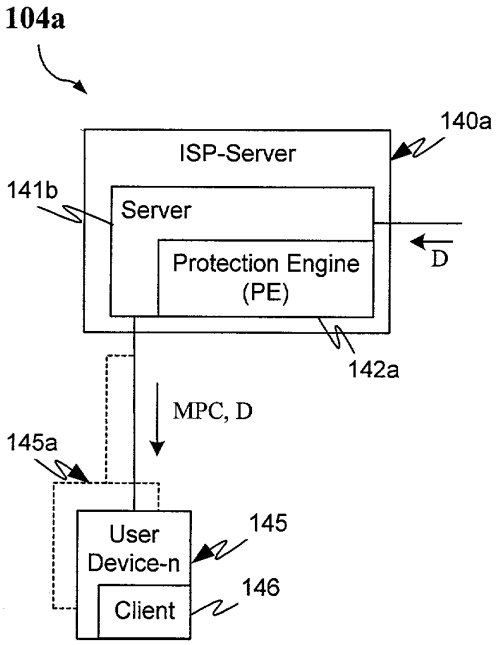

FIG. 1b

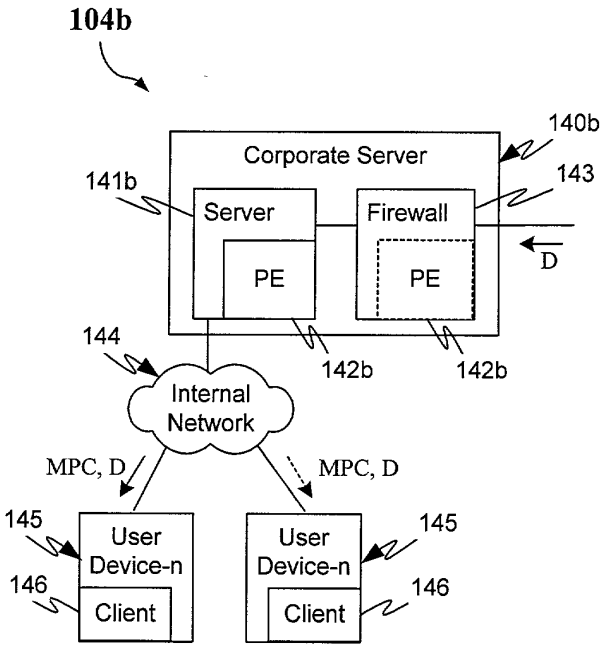

FIG. 1c

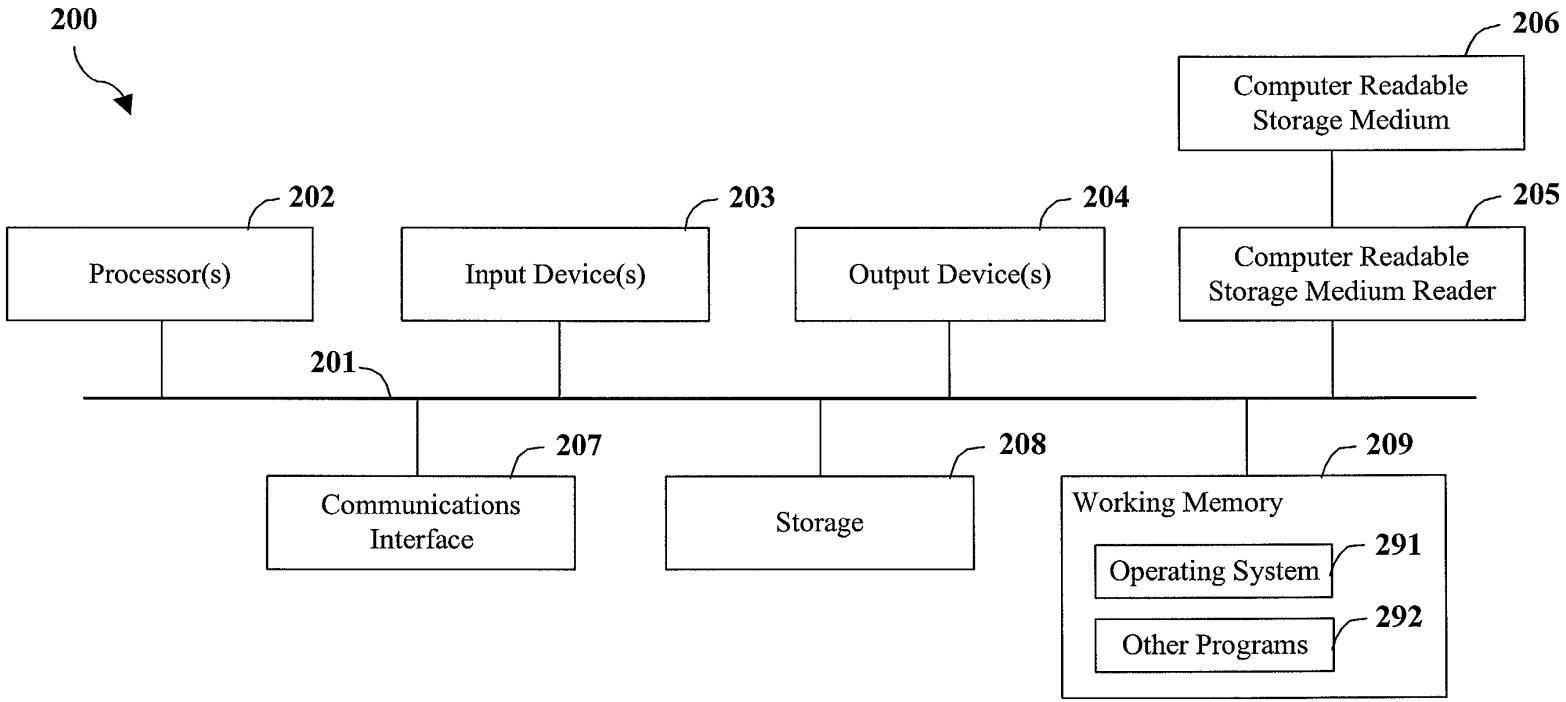
200
206
Computer Readable Storage Medium
202
Processor(s)
203
Input Device(s)
204
Output Device(s)
205
Computer Readable Storage Medium Reader
201
207
Communications Interface
208
Storage
209
Working Memory
Operating System
291
Other Programs
292

FIG. 2

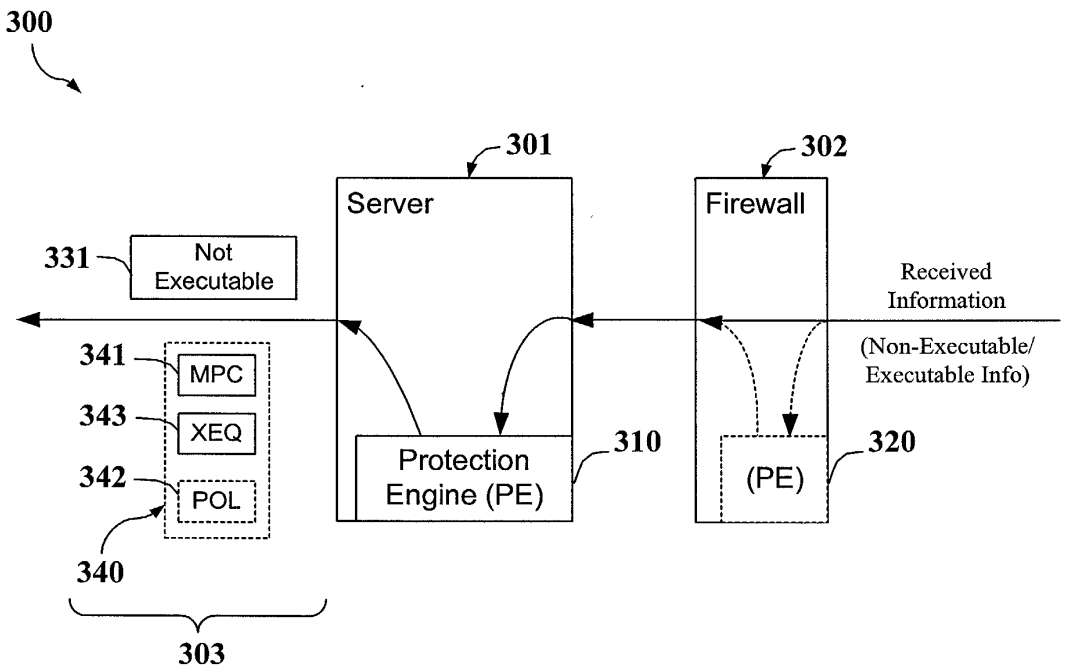
300
301
302
Server
Firewall
331
Not Executable
Received Information
(Non-Executable/ Executable Info)
341
MPC
343
XEQ
342
POL
340
303
Protection Engine (PE)
310
(PE)
320

FIG. 3

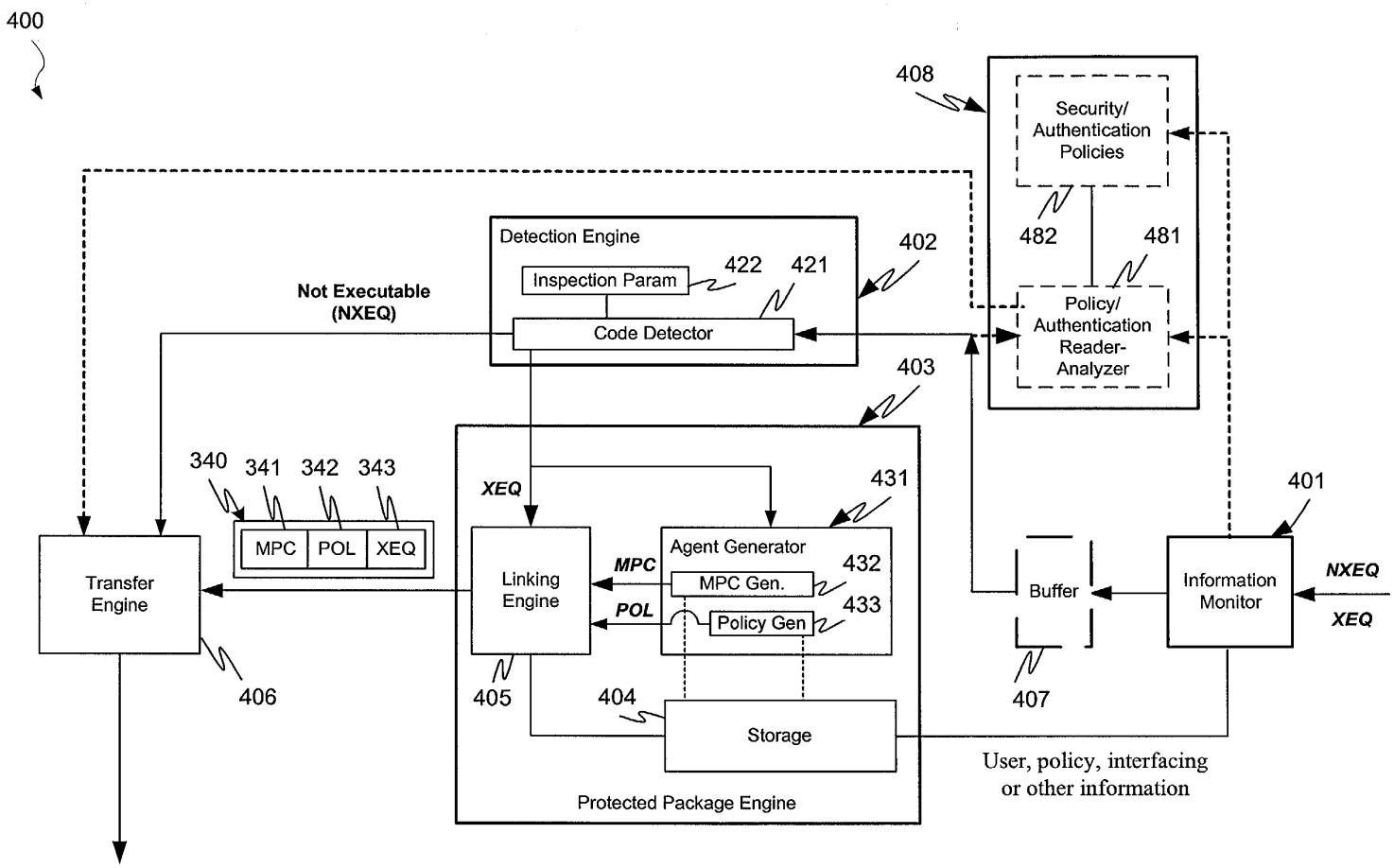
400
408
Security/ Authentication Policies
482
481
Policy/ Authentication Reader-Analyzer
Detection Engine
Inspection Param
422
421
402
Code Detector
Not Executable (NXEQ)
403
340
341
342
343
MPC
POL
XEQ
XEQ
431
Agent Generator
MPC Gen.
432
Policy Gen
433
MPC
POL
Linking Engine
Transfer Engine
406
405
404
Storage
Protected Package Engine
Buffer
407
Information Monitor
401
NXEQ
XEQ
User, policy, interfacing or other information

FIG. 4

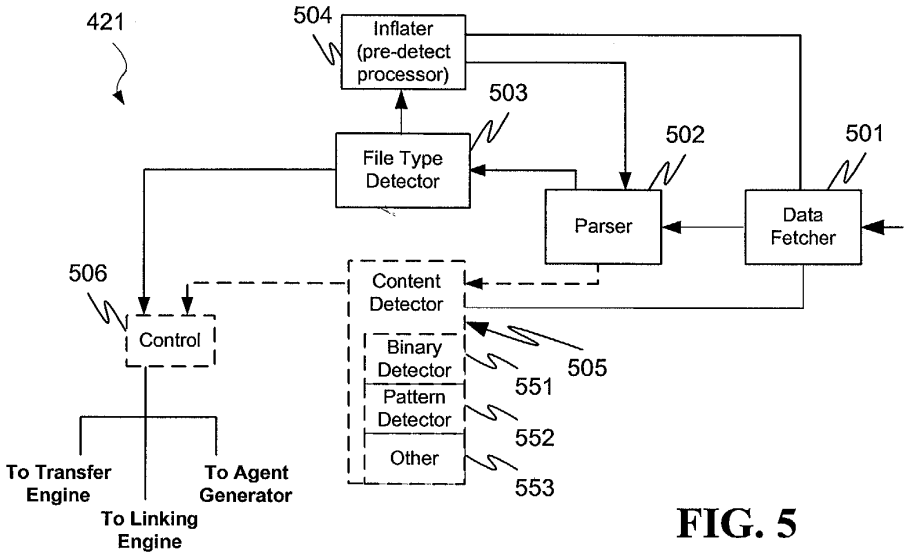

FIG. 5

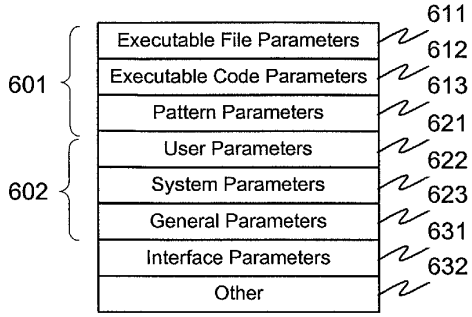

FIG. 6a

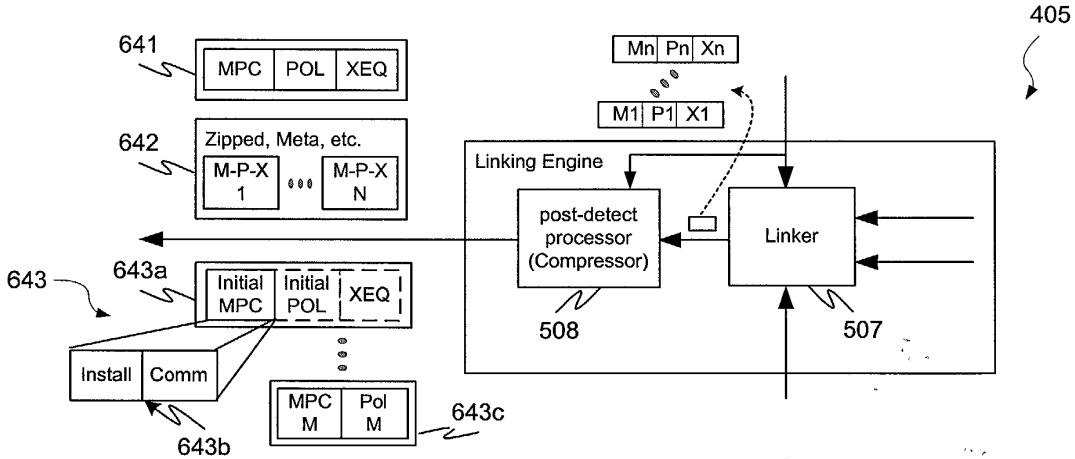

FIG. 6b

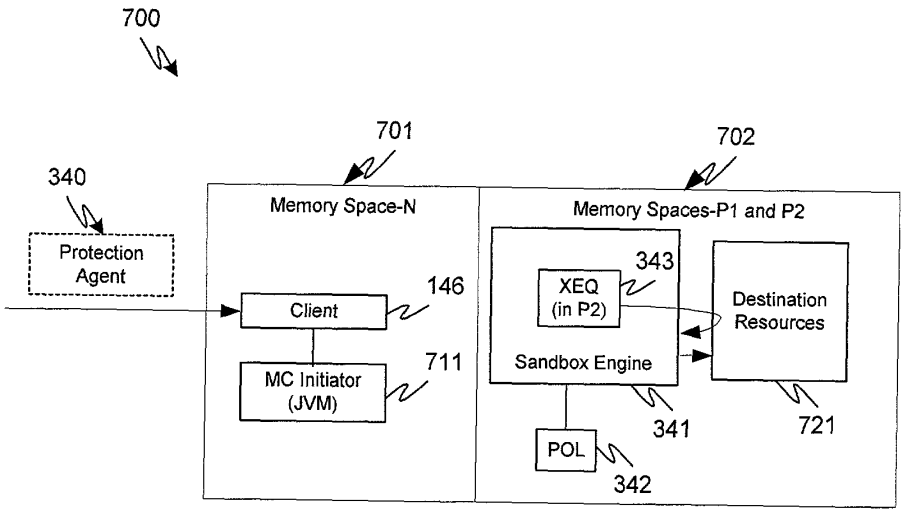

FIG. 7a

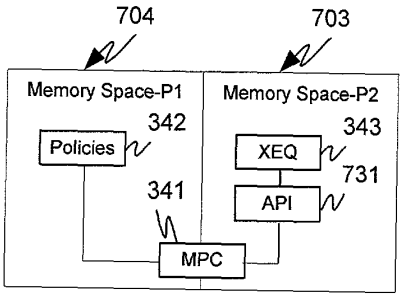

FIG. 7b

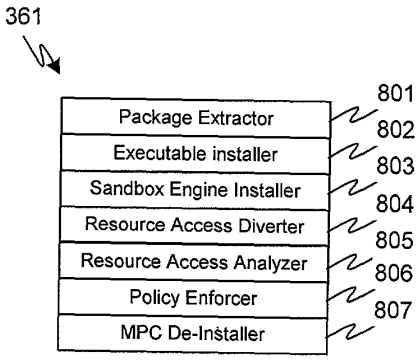

FIG. 8

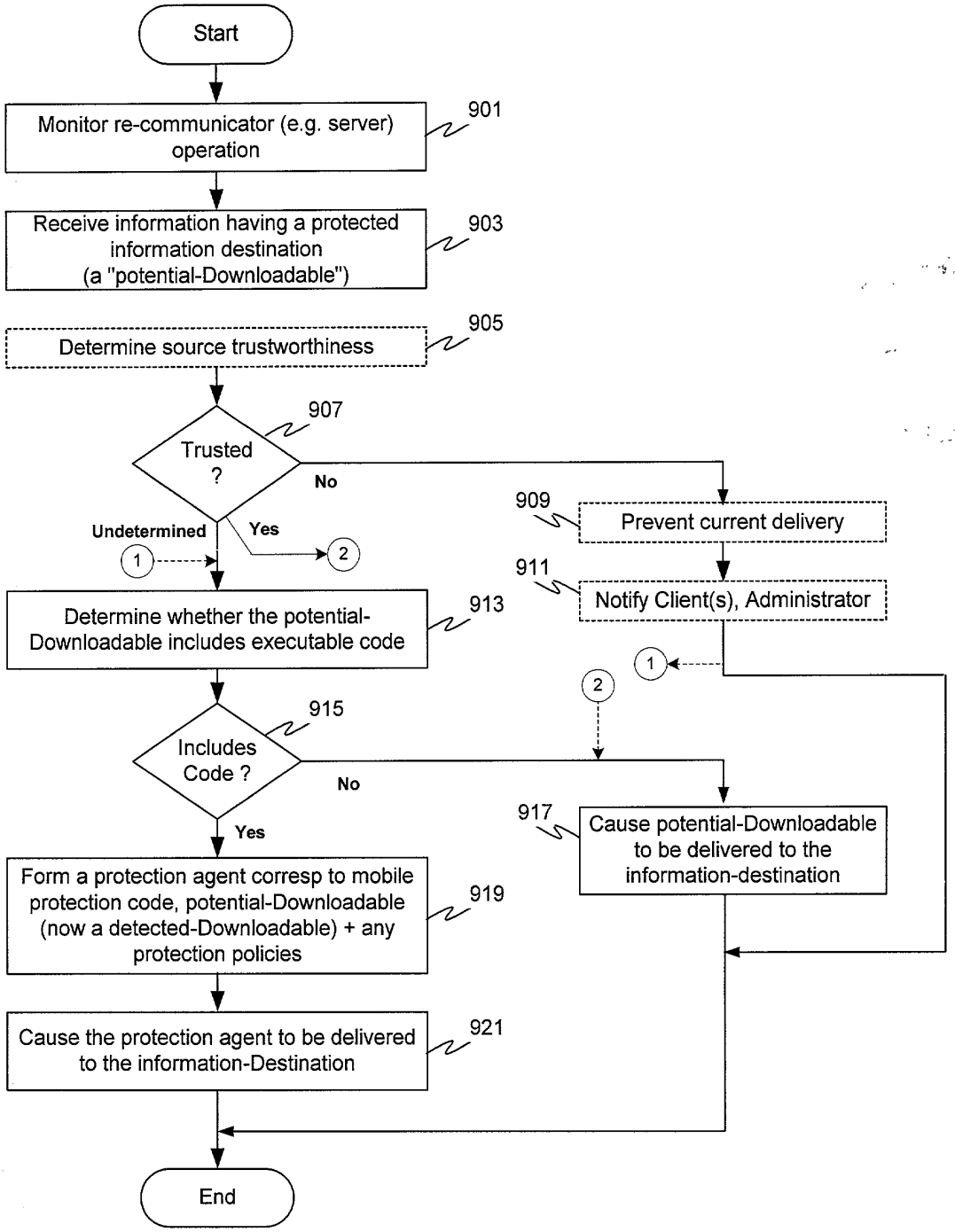
Start
Monitor re-communicator (e.g. server) operation
901
Receive information having a protected information destination (a "potential-Downloadable")
903
Determine source trustworthiness
905
Trusted ?
907
No
Undetermined
Yes
1
2
909
Prevent current delivery
911
Notify Client(s), Administrator
Determine whether the potential-Downloadable includes executable code
913
1
2
Includes Code ?
915
No
Yes
917
Cause potential-Downloadable to be delivered to the information-destination
Form a protection agent corresp to mobile protection code, potential-Downloadable (now a detected-Downloadable) + any protection policies
919
Cause the protection agent to be delivered to the information-Destination
921
End

FIG. 9

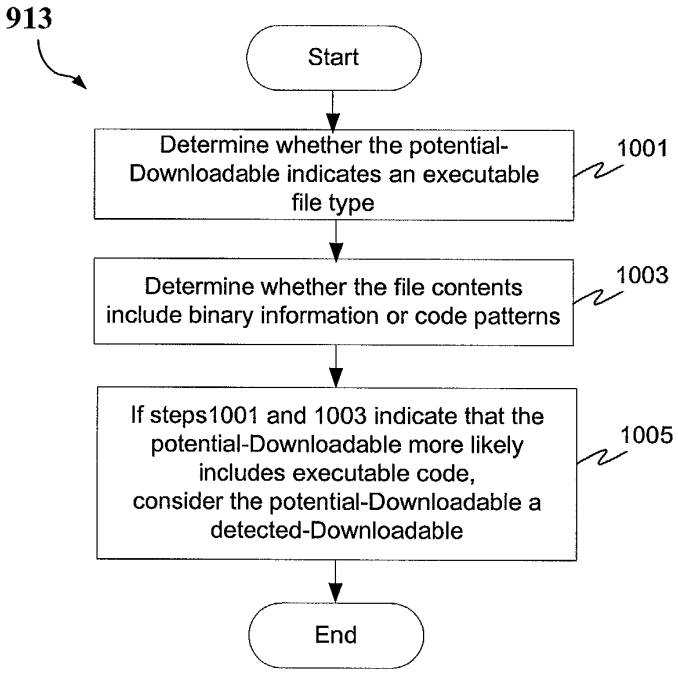

FIG. 10A

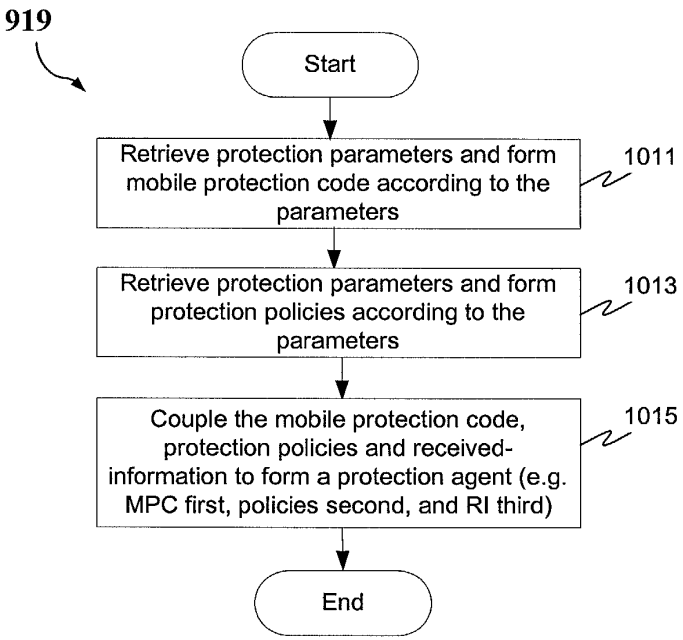

FIG. 10B

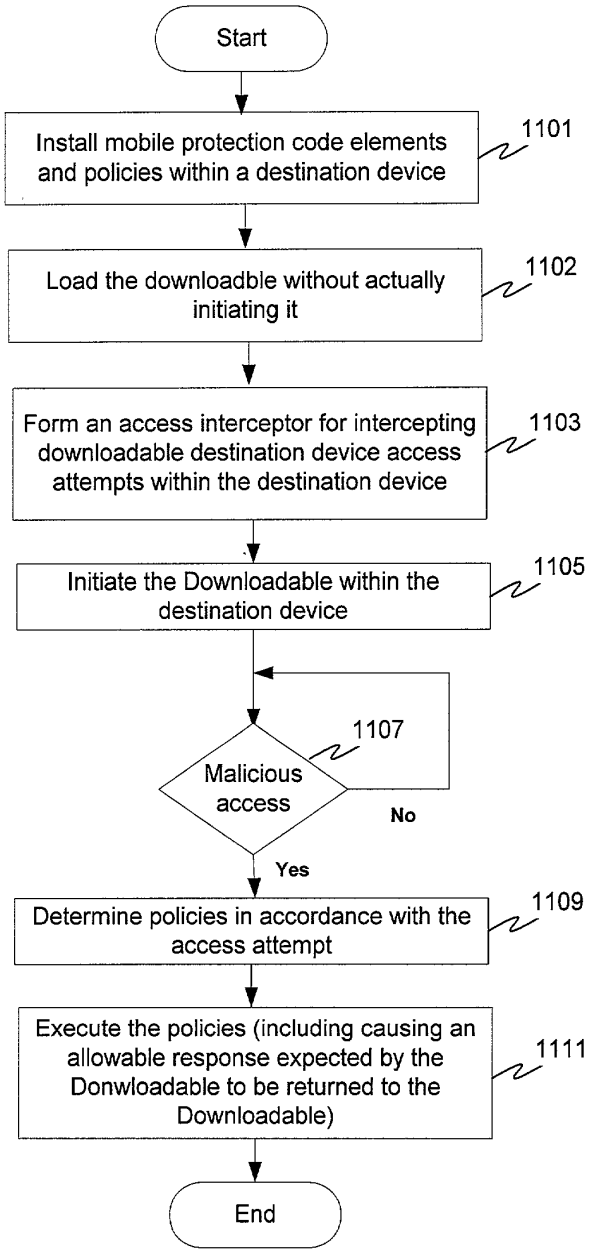
Start
Install mobile protection code elements and policies within a destination device
1101
Load the downloadble without actually initiating it
1102
Form an access interceptor for intercepting downloadable destination device access attempts within the destination device
1103
Initiate the Downloadable within the destination device
1105
Malicious access
1107
No
Yes
Determine policies in accordance with the access attempt
1109
Execute the policies (including causing an allowable response expected by the Donwloadable to be returned to the Downloadable)
1111
End

FIG. 11

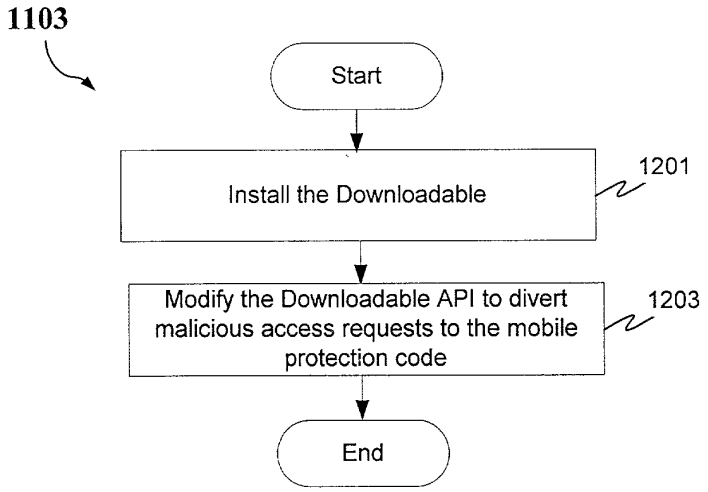

FIG. 12a

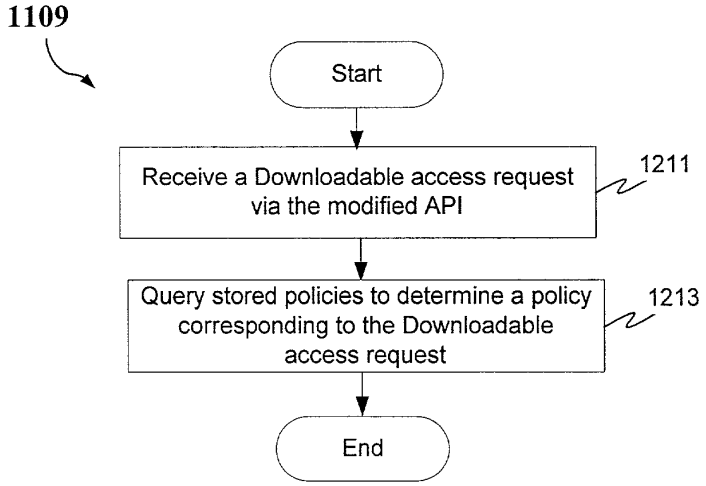

FIG. 12b

Sector
#

OIPE JC139 SEP 10 2001 PATENT & TRADEMARK OFFICE

Please type a plus sign (+) inside this box → [+]

PTO/SB/21 (08-00)
Approved for use through 10/31/2002. OMB 0651-0031
U.S. Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE

TRANSMITTAL FORM

(to be used for all correspondence after initial filing)

Application Number	09/861,229
Filing Date	May 17, 2001
First Named Inventor	Yigal Edery, et al.
Group Art Unit	2152
Examiner Name	Unknown
Total Number of Pages in This Submission: 27	Attorney Docket Number: 43426.00014

ENCLOSURES *(check all that apply)*

- [x] Fee Transmittal Form (in duplicate)
  - [ ] Fee Attached
  - [x] Deposit Account Authorization on Fee Transmittal Form
- [ ] Amendment / Response
  - [ ] After Final
  - [ ] Affidavits/declaration(s)
- [ ] Extension of Time Request
- [x] Return Postcard
- [ ] Information Disclosure Statement
- [ ] Certified Copy of Priority Document(s)
- [x] Response to Missing Parts/ Incomplete Application (in duplicate)
  - [x] Response to Missing Parts under 37 CFR 1.52 or 1.53
- [ ] Assignment Papers *(for an Application)*
- [x] Informal Drawings consisting of Figures 1a, 1b, 1c, 2, 3, 4, 5, 6a, 6b, 7a, 7b, 8, 9, 10a, 10b, 11, 12a, and 12b
- [ ] Licensing-related Papers
- [ ] Petition
- [ ] Petition to Convert to a Provisional Application
- [x] Combined Power of Attorney and Declaration for Patent Application
- [ ] Terminal Disclaimer
- [ ] Request for Refund
- [ ] CD, Number of CD(s) ____
- [ ] After Allowance Communication to Group
- [ ] Appeal Communication to Board of Appeals and Interferences
- [ ] Appeal Communication to Group *(Appeal Notice, Brief, Reply Brief)*
- [ ] Proprietary Information
- [ ] Status Letter
- [x] Other Enclosure(s) *(please identify below)*:

  Letter to the Official Draftsperson (Request to Substitute Drawings) (in duplicate)

Remarks

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT

Firm *or* Individual name	Daryl C. Josephson, Reg. No. 37,365 Squire, Sanders & Dempsey, L.L.P. 600 Hansen Way Palo Alto, CA 94304-1043
Signature	Daryl C. Josephson
Date	September 10, 2001

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Assistant Commissioner for Patents, Washington, D.C. 20231 on this date: September 10, 2001

Typed or printed name	Sandy Yi		
Signature	Sandy Yi	Date	September 10, 2001

PTO/SB/17 (11-00)
Approved for use through 10/31/2002. OMB 0651-0032
U.S. Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE

FEE TRANSMITTAL for FY 2001

Patent fees are subject to annual revision.

TOTAL AMOUNT OF PAYMENT ($) 65

Complete if Known	
Application Number	09/861,229
Filing Date	May 17, 2001
First Named Inventor	Yigal Edery, et al.
Examiner Name	Unknown
Group / Art Unit	2152
Attorney Docket No.	43426.00014

METHOD OF PAYMENT (check one)

1. ☒ The Commissioner is hereby authorized to charge indicated fees and credit any over payments to:

Deposit Account Number: 05-0150

Deposit Account Name: Squire, Sanders & Dempsey, L.L.P.

☒ Charge Any Additional Fee Required Under 37 CFR 1.16 and 1.17

☒ Applicant claims small entity status. See 37 CFR 1.27

2. ☐ Payment Enclosed:

☐ Check ☐ Credit card ☐ Money Order ☐ Other

FEE CALCULATION

1. BASIC FILING FEE

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description	Fee Paid
101	710	201	355	Utility filing fee	
106	320	206	160	Design filing fee	
107	490	207	245	Plant filing fee	
108	710	208	355	Reissue filing fee	
114	150	214	75	Provisional filling fee	

SUBTOTAL (1) ($) 0

2. EXTRA CLAIM FEES

			Extra Claims		Fee from below		Fee Paid
Total Claims		-20	= 0	X		=	0
Independent Claims		-3	= 0	X		=	0
Multiple Dependent				X		=	0

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description
103	18	203	9	Claims in excess of 20
102	80	202	40	Independent claims in excess of 3
104	270	204	135	Multiple dependent claim, if not paid
109	80	209	40	** Reissue independent claims over original patent
110	18	210	9	** Reissue claims in excess of 20 and over original patent

SUBTOTAL (2) ($) 0

**or number previously paid, if greater; For Reissues, see above

FEE CALCULATION (continued)

3. ADDITIONAL FEES

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description	Fee Paid
105	130	205	65	Surcharge - late filing fee or oath	65
127	50	227	25	Surcharge - late provisional filing fee or cover sheet.	
139	130	139	130	Non-English specification	
147	2,520	147	2,520	For filing a request for reexamination	
112	920*	112	920*	Requesting publication of SIR prior to Examiner action	
113	1,840*	113	1,840*	Requesting publication of SIR after Examiner action	
115	110	215	55	Extension for reply within first month	
116	390	216	195	Extension for reply within second month	
117	890	217	445	Extension for reply within third month	
118	1,390	218	695	Extension for reply within fourth month	
128	1,890	228	945	Extension for reply within fifth month	
119	310	219	155	Notice of Appeal	
120	310	220	155	Filing a brief in support of an appeal	
121	270	221	135	Request for oral hearing	
138	1,510	138	1,510	Petition to institute a public use proceeding	
140	110	240	55	Petition to revive – unavoidable	
141	1,240	241	620	Petition to revive – unintentional	
142	1,240	242	620	Utility issue fee (or reissue)	
143	440	243	220	Design issue fee	
144	600	244	300	Plant issue fee	
122	130	122	130	Petitions to the Commissioner	
123	130	123	130	Petitions related to provisional applications	
126	180	126	180	Submission of Information Disclosure Stmt	
581	40	581	40	Recording each patent assignment per property (times number of properties)	
146	710	246	355	Filing a submission after final rejection (37 CFR § 1.129(a))	
149	710	249	355	For each additional invention to be examined (37 CFR § 1.129(b))	
179	710	279	355	Request for Continued Examination (RCE)	
169	900	169	900	Request for expedited examination of a design application	
				Other fee (specify) ____	

*Reduced by Basic Filing Fee Paid

SUBTOTAL (3) ($) 65

SUBMITTED BY				Complete *(if applicable)*	
Name (Print/Type)	Daryl C. Josephson	*Registration No. Attorney/Agent)*	37,365	*Telephone*	650.856.6500
Signature	Daryl C Josephson			*Date*	September 10, 2001

WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.

Burden Hour Statement: This form is estimated to take 0.2 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Assistant Commissioner for Patents, Washington, DC 20231.

PATENT
Attorney Docket No.: 43426.00014

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF MAILING

I hereby certify that this paper (along with any paper referred to as being attached or enclosed) is being deposited with the United States Postal Service on the date shown below with sufficient postage as first class mail in an envelope addressed to Commissioner for Patents, Washington, D.C. 20231, on

Date: 9/10/01 By: Sandy Yi
Sandy Yi

In re Application of:		Examiner:	Unknown
	Yigal Edery, et al.		
Serial No.	09/861,229	Art Unit:	2152
Filed:	May 17, 2001		
Title:	MALICIOUS MOBILE CODE RUNTIME MONITORING SYSTEM AND METHODS		

Commissioner for Patents
Washington, D.C. 20231

RESPONSE TO NOTICE TO FILE MISSING PARTS OF NONPROVISIONAL APPLICATION

Dear Sir:

In response to the Notice to File Missing Parts of Nonprovisional Papers mailed on July 19, 2001, in the above-identified application, enclosed herewith are the following:

1) Copy of Notice to File Missing Parts of Nonprovisional Application

2) Combined Power of Attorney and Declaration for Patent Application

3) Ten (10) sheets of informal drawings consisting of Figures 1a, 1b, 1c, 2, 3, 4, 5, 6a, 6b, 7a, 7b, 8, 9, 10a, 10b, 11, 12a, and 12b

4) Letter to the Official Draftsperson (Request to Substitute Drawings) (in duplicate)

5) Transmittal Form

6) Fee Transmittal (in duplicate)

7) Acknowledgment Postcard

If the Examiner has any questions or needs additional information, the Examiner is invited to telephone the undersigned attorney at (650) 856-6500.

If for any reason an insufficient fee has been paid, please charge the insufficiency to Deposit Account No. **05-0150**. A duplicate of this communication is enclosed.

Date: 9/10/01

Respectfully submitted,

SQUIRE, SANDERS & DEMPSEY L.L.P.
600 Hansen Way
Palo Alto, California 94304-1043
Telephone: (650) 856-6500
Facsimile: (650) 843-8777

Daryl C. Josephson
Attorney for Applicants
Registration No.: 37,365

UNITED STATES PATENT AND TRADEMARK OFFICE

COMMISSIONER FOR PATENTS
UNITED STATES PATENT AND TRADEMARK OFFICE
WASHINGTON, D.C. 20231
www.uspto.gov

APPLICATION NUMBER	FILING/RECEIPT DATE	FIRST NAMED APPLICANT	ATTORNEY DOCKET NUMBER
09/861,229	05/17/2001	Yigal Edery	43426.00014

CONFIRMATION NO. 5421

FORMALITIES LETTER

OC000000006314695

Intellectual Property Department
Squire, Sanders & Dempsey L.L.P.
600 Hansen Way
Palo Alto, CA 94304-1043

Date Mailed: 07/19/2001

NOTICE TO FILE MISSING PARTS OF NONPROVISIONAL APPLICATION

FILED UNDER 37 CFR 1.53(b)

Filing Date Granted

An application number and filing date have been accorded to this application. The item(s) indicated below, however, are missing. Applicant is given **TWO MONTHS** from the date of this Notice within which to file all required items and pay any fees required below to avoid abandonment. Extensions of time may be obtained by filing a petition accompanied by the extension fee under the provisions of 37 CFR 1.136(a).

- The oath or declaration is missing.
  A properly signed oath or declaration in compliance with 37 CFR 1.63, identifying the application by the above Application Number and Filing Date, is required.
- To avoid abandonment, a late filing fee or oath or declaration surcharge as set forth in 37 CFR 1.16(e) of $65 for a small entity in compliance with 37 CFR 1.27, must be submitted with the missing items identified in this letter.
- **The balance due by applicant is $ 65.**

The application is informal since it does not comply with the regulations for the reason(s) indicated below.

The required item(s) identified below must be timely submitted to avoid abandonment:

- Substitute drawings in compliance with 37 CFR 1.84 because:
  - drawing sheets do not have the appropriate margin(s) (see 37 CFR 1.84(g)). Each sheet must include a top margin of at least 2.5 cm. (1 inch), a left side margin of at least 2.5 cm. (1 inch), a right side margin of at least 1.5 cm. (5/8 inch), and a bottom margin of at least 1.0 cm. (3/8 inch);

A copy of this notice MUST be returned with the reply.

Customer Service Center
Initial Patent Examination Division (703) 308-1202
PART 2 - COPY TO BE RETURNED WITH RESPONSE

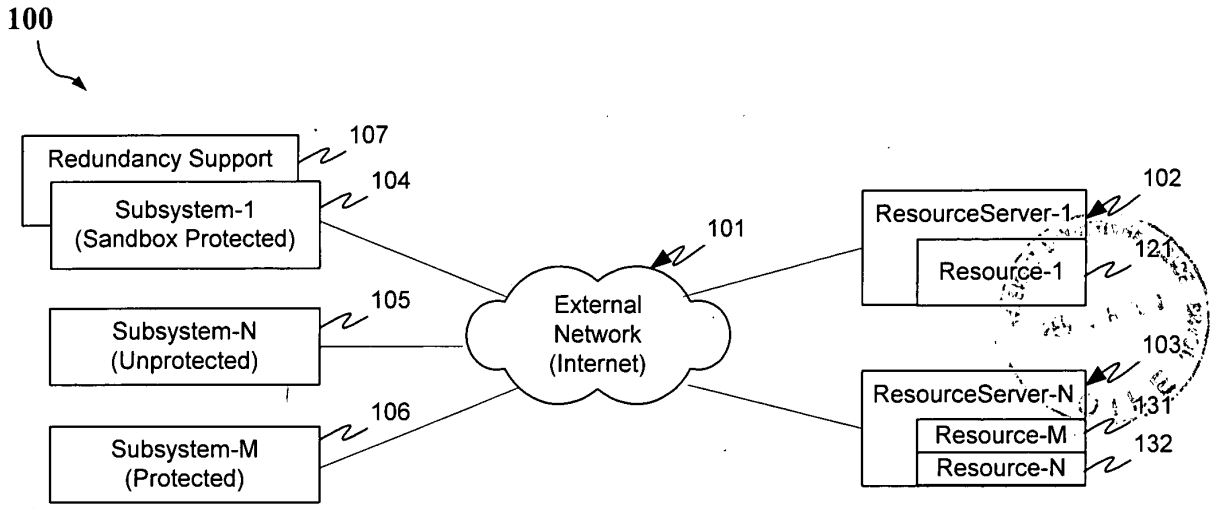

FIG. 1a

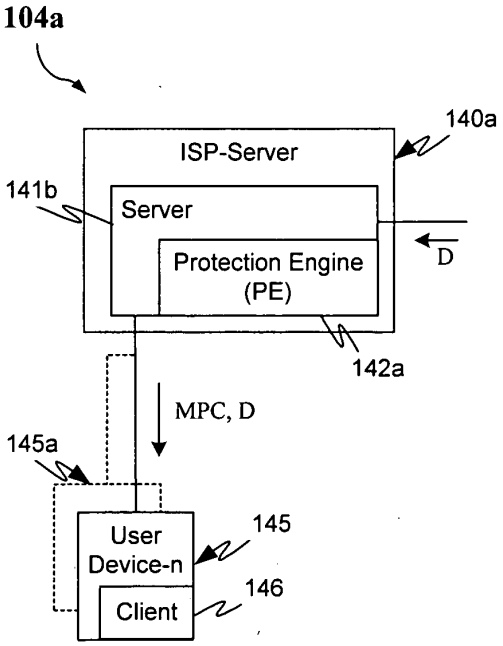

FIG. 1b

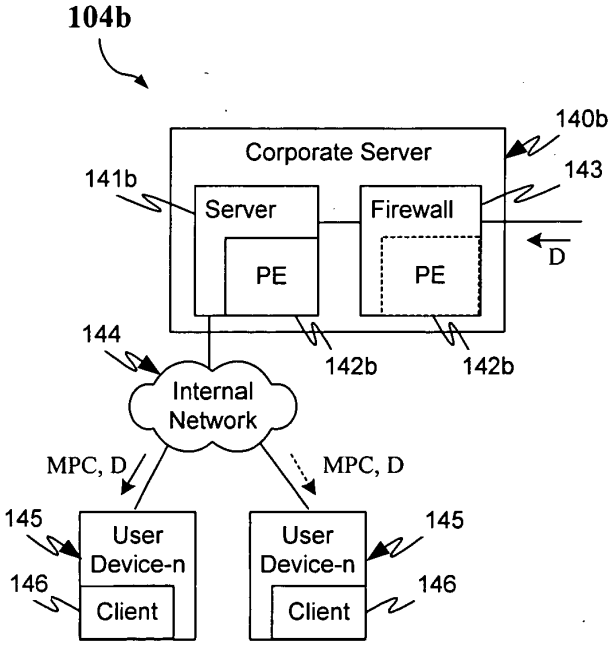

FIG. 1c

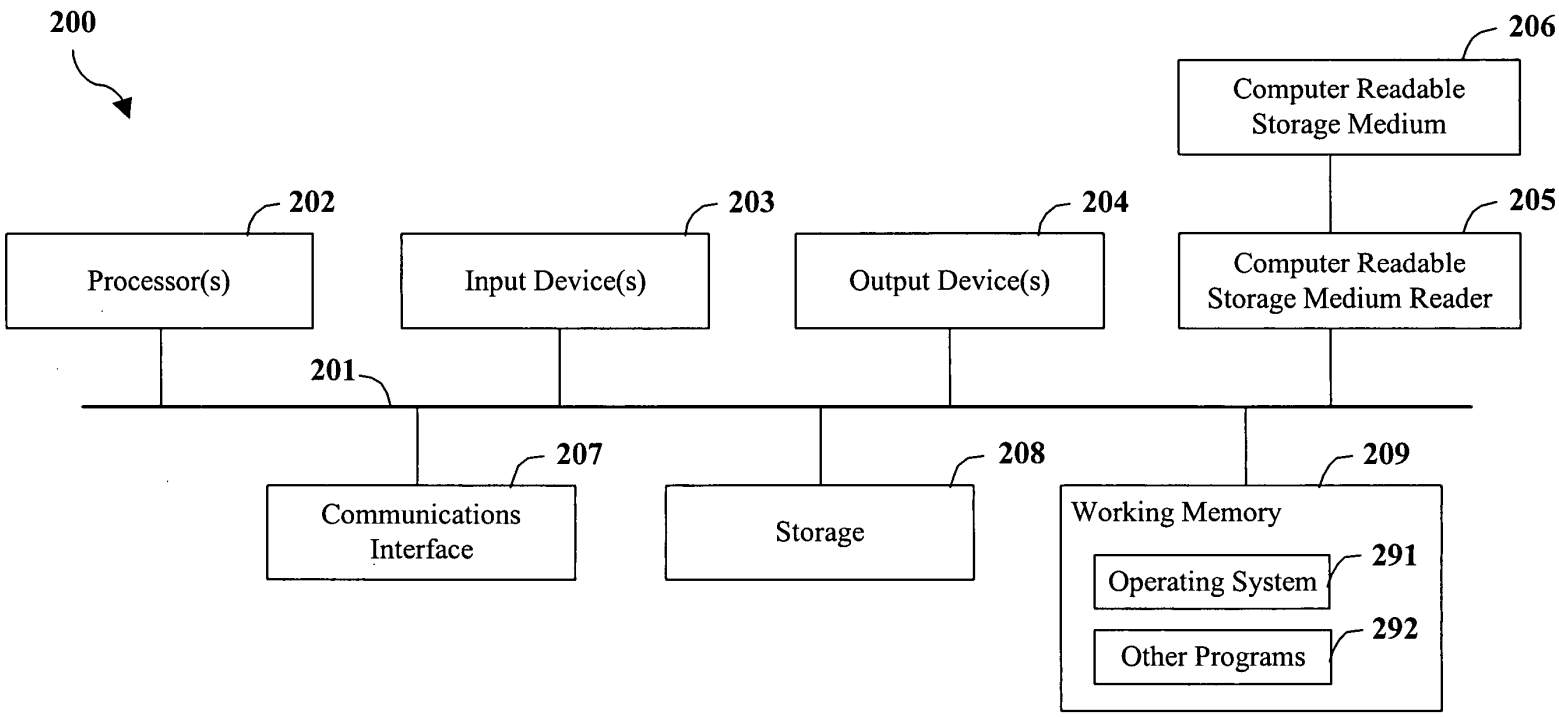
200
206
Computer Readable Storage Medium
202
Processor(s)
203
Input Device(s)
204
Output Device(s)
205
Computer Readable Storage Medium Reader
201
207
Communications Interface
208
Storage
209
Working Memory
Operating System
291
Other Programs
292

FIG. 2

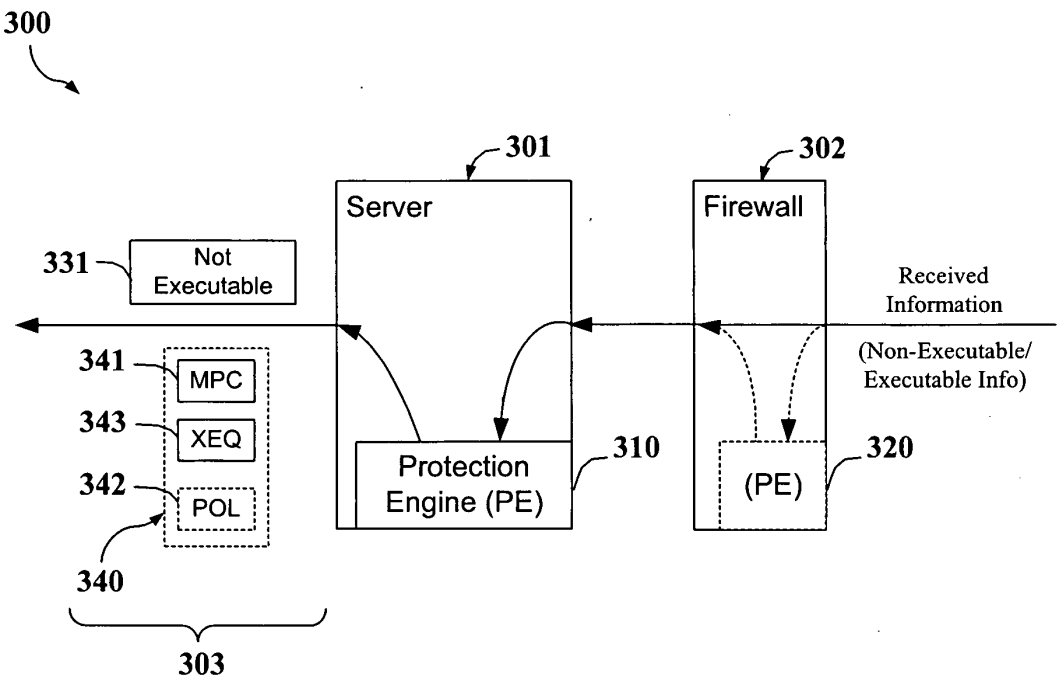
300
301
Server
302
Firewall
331
Not
Executable
Received
Information
(Non-Executable/
Executable Info)
341
MPC
343
XEQ
342
POL
340
303
Protection
Engine (PE)
310
(PE)
320

FIG. 3

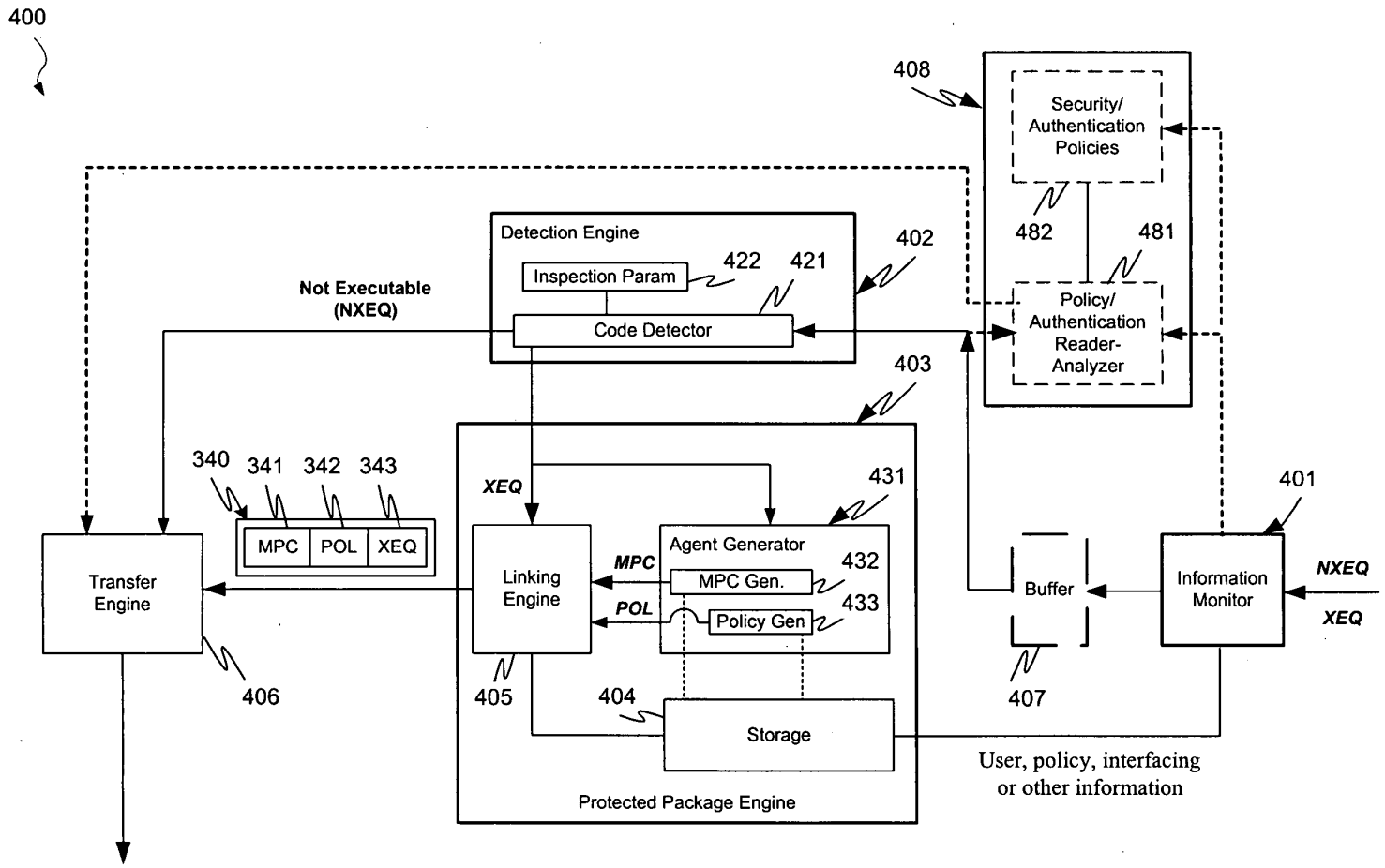
400
408
Security/ Authentication Policies
482
481
Policy/ Authentication Reader-Analyzer
Detection Engine
Inspection Param
422
421
402
Code Detector
Not Executable (NXEQ)
403
340
341
342
343
MPC
POL
XEQ
XEQ
431
Agent Generator
MPC
MPC Gen.
432
POL
Policy Gen
433
Linking Engine
Transfer Engine
406
405
404
Storage
Protected Package Engine
Buffer
407
Information Monitor
401
NXEQ
XEQ
User, policy, interfacing or other information

FIG. 4

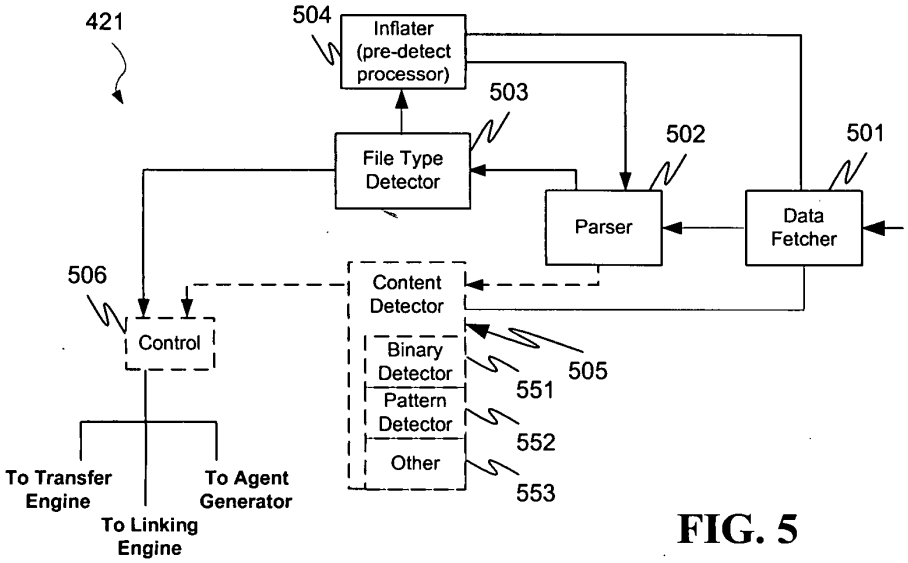

FIG. 5

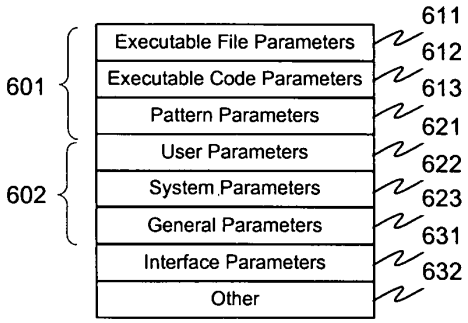

FIG. 6a

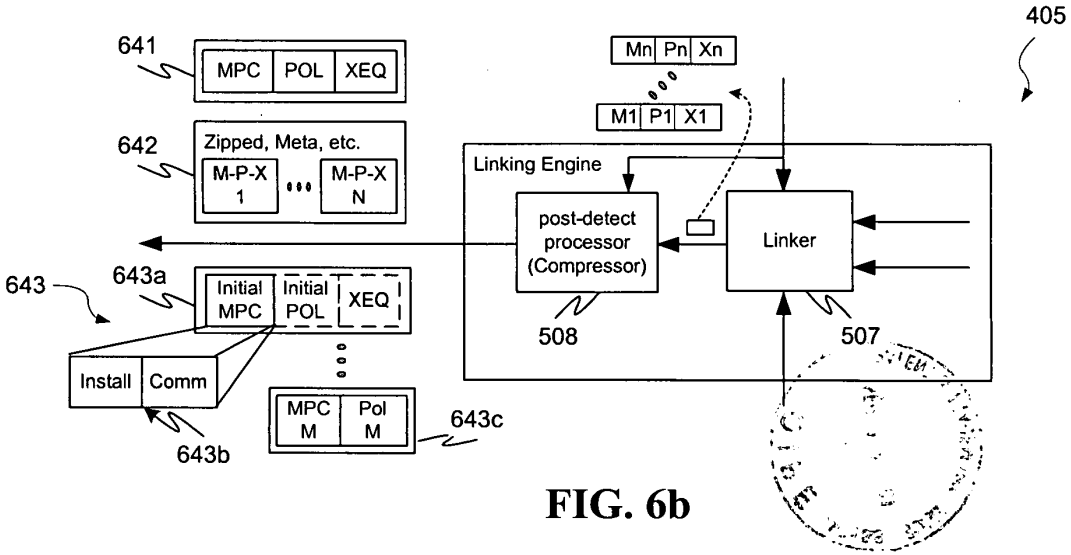

FIG. 6b

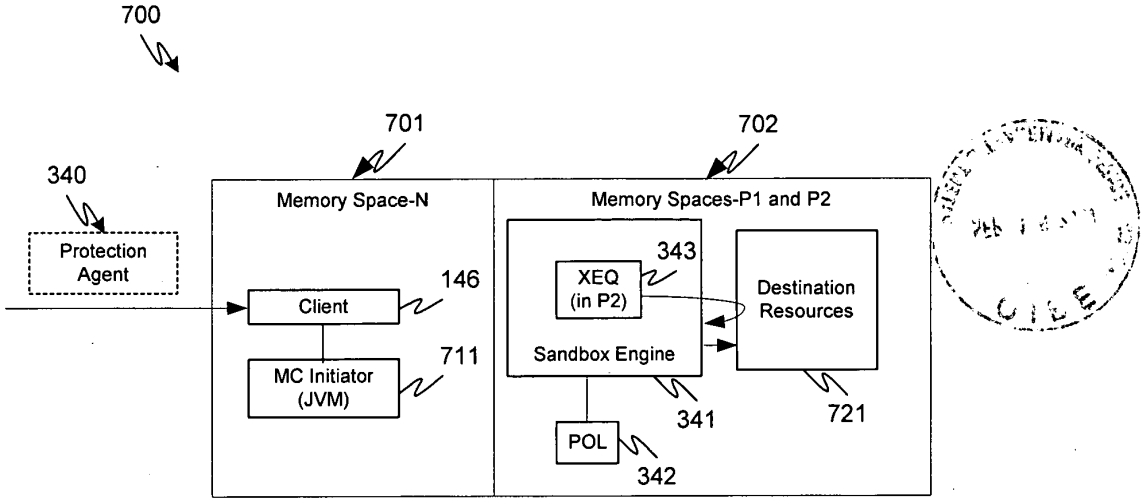

FIG. 7a

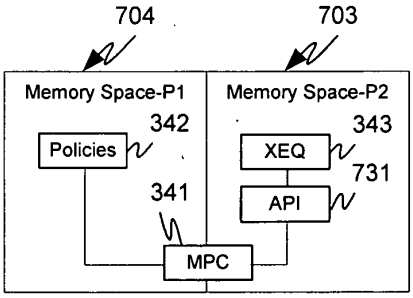

FIG. 7b

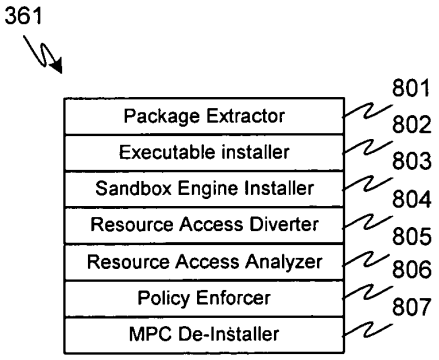

FIG. 8

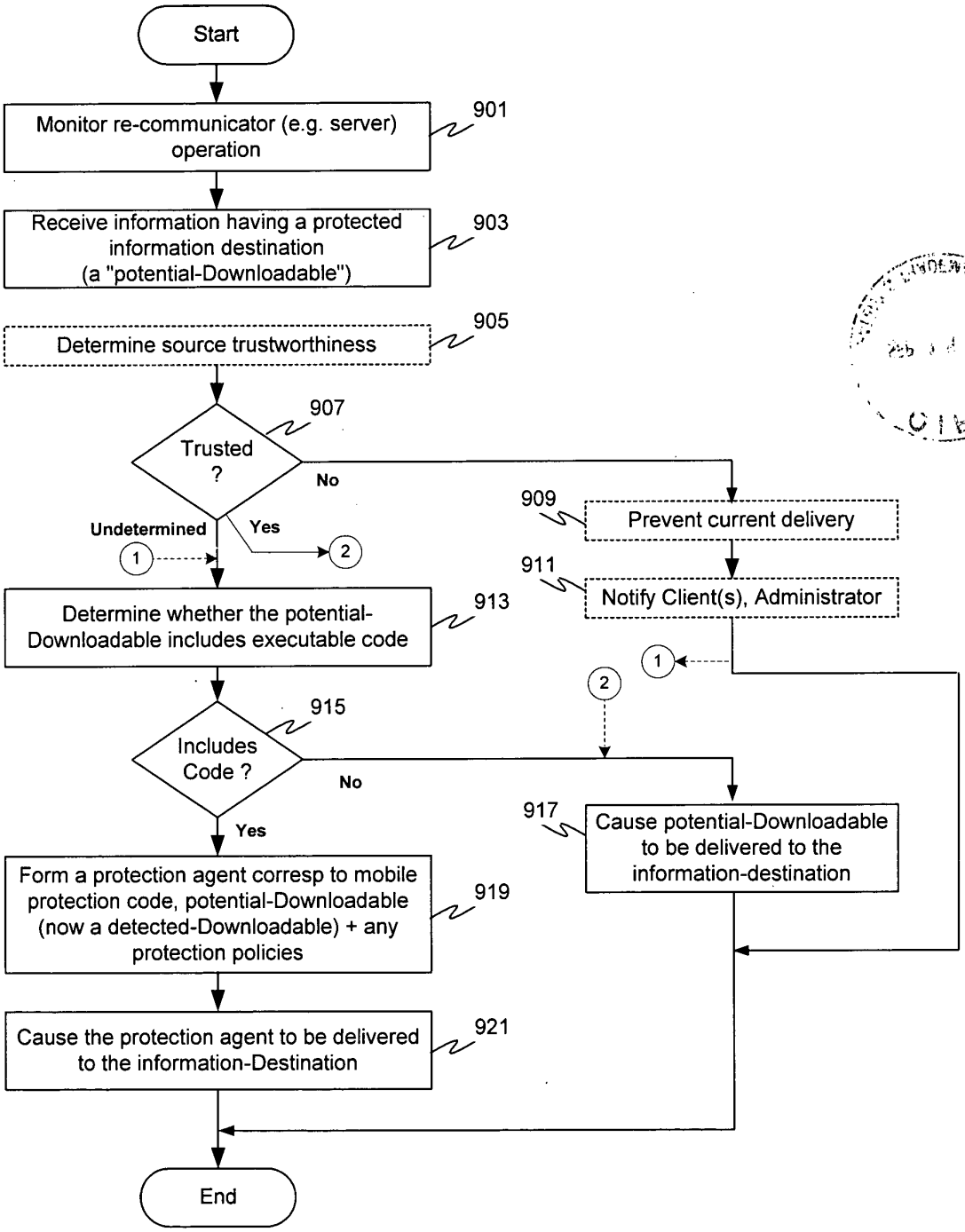
Start
Monitor re-communicator (e.g. server) operation
901
Receive information having a protected information destination (a "potential-Downloadable")
903
Determine source trustworthiness
905
Trusted ?
907
No
Undetermined
Yes
1
2
909
Prevent current delivery
911
Notify Client(s), Administrator
Determine whether the potential-Downloadable includes executable code
913
1
2
Includes Code ?
915
No
Yes
917
Cause potential-Downloadable to be delivered to the information-destination
Form a protection agent corresp to mobile protection code, potential-Downloadable (now a detected-Downloadable) + any protection policies
919
Cause the protection agent to be delivered to the information-Destination
921
End

FIG. 9

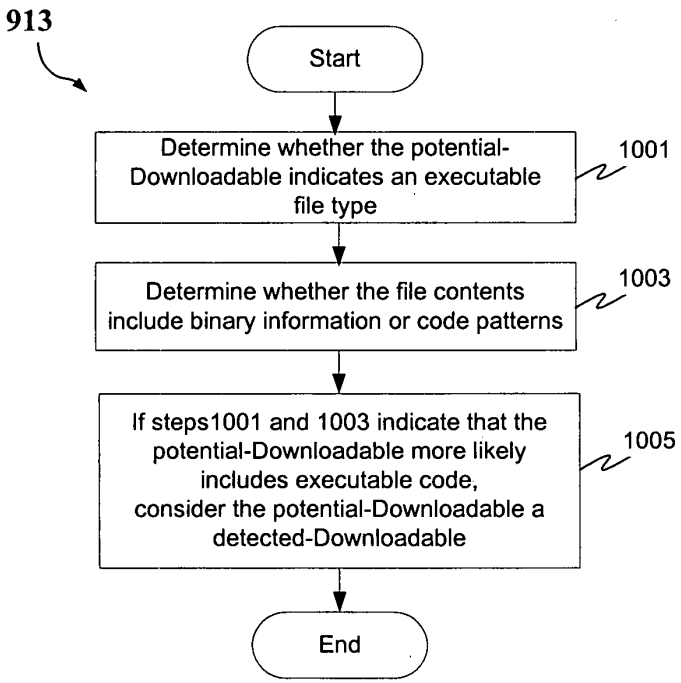

FIG. 10A

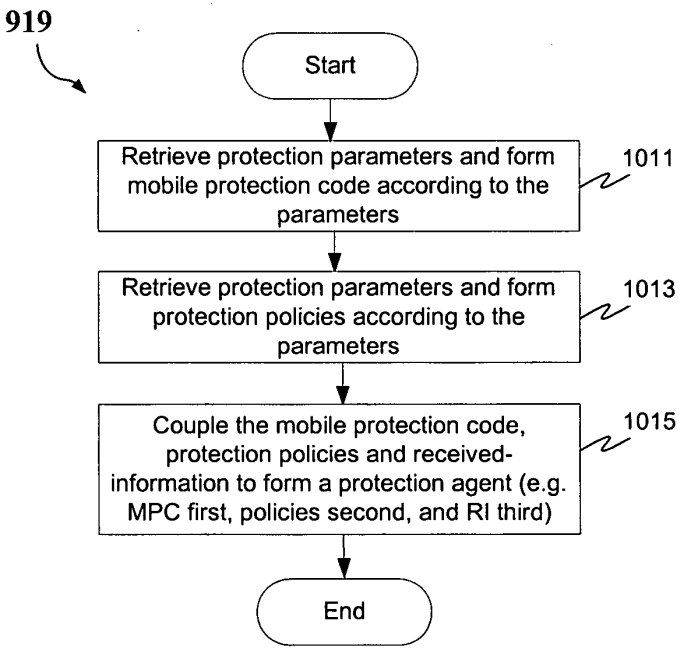

FIG. 10B

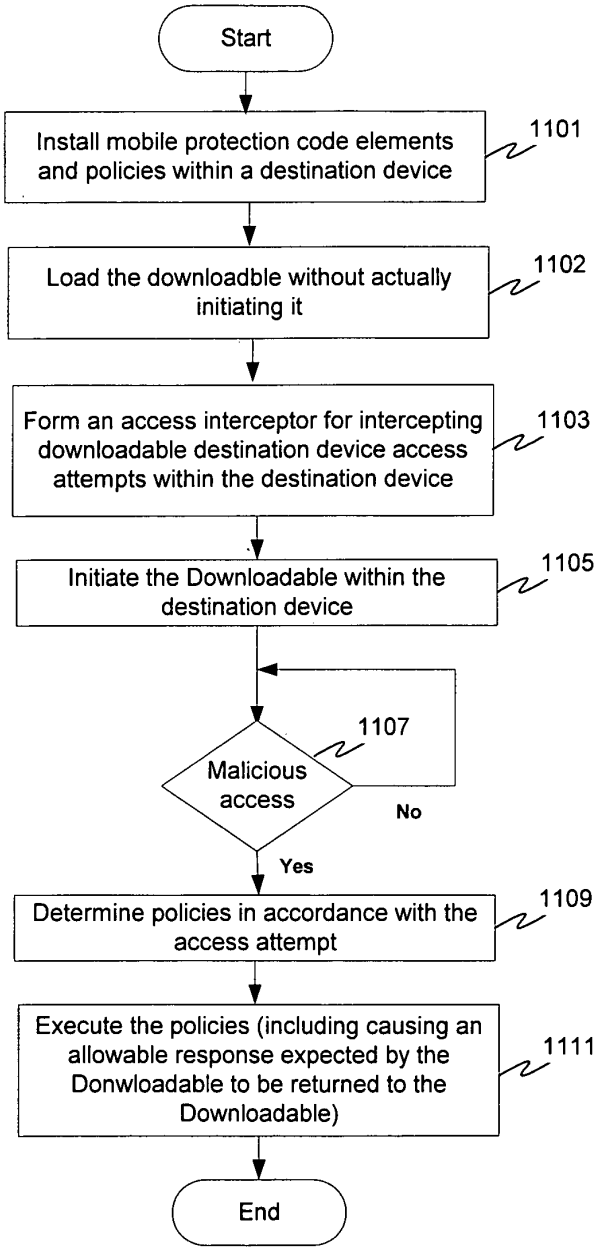
Start
Install mobile protection code elements and policies within a destination device
1101
Load the downloadble without actually initiating it
1102
Form an access interceptor for intercepting downloadable destination device access attempts within the destination device
1103
Initiate the Downloadable within the destination device
1105
Malicious access
1107
No
Yes
Determine policies in accordance with the access attempt
1109
Execute the policies (including causing an allowable response expected by the Donwloadable to be returned to the Downloadable)
1111
End

FIG. 11

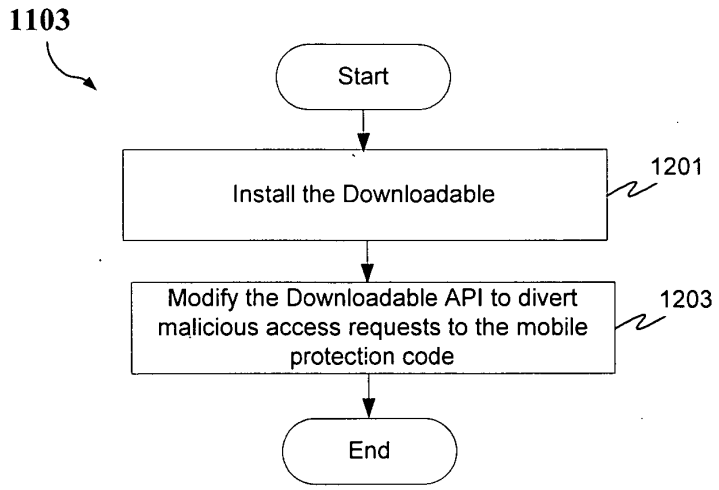

FIG. 12a

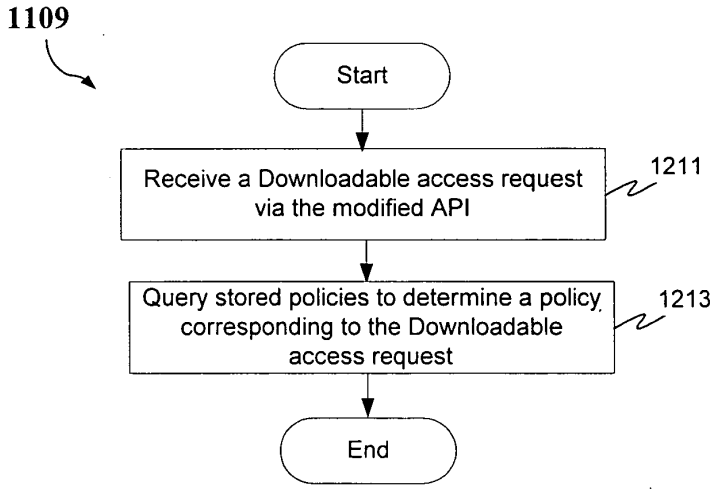

FIG. 12b

#3

PATENT
Attorney Docket No.: 43426.00014

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:

Yigal Edery, et al.

Serial No. 09/861,229

Filed: May 17, 2001

COMBINED POWER OF ATTORNEY AND DECLARATION FOR PATENT APPLICATION

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name,

I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter, which is claimed and for which a patent is sought on the invention entitled:

MALICIOUS MOBILE CODE RUNTIME MONITORING SYSTEM AND METHODS

the specification of which

☐ is attached hereto

OR

☒ was filed on May 17, 2001 as United States Application Number or PCT International Application Number 09/861,229.

☐ and was amended on _____
(if applicable)

I hereby state that I have reviewed and understand the contents of the above-identified specification, including the claims, as amended by any amendment referred to above.

I acknowledge the duty to disclose information which is material to the patentability of this application in accordance with Title 37, Code of Federal Regulations, §1.56.

I hereby claim the benefit under Title 35, United States, §119 (e) of any United States provisional application(s) listed below.

60/205,591	May 17, 2000
(Application Number)	(Filing Date)
(Application Number)	(Filing Date)

I hereby claim foreign priority benefits under Title 35, United States Code, §119 (a)-(d) or §365(b) of any foreign application(s) for patent or inventor's certificate, or §365 (a) of any PCT international application(s) which designated at least one country other than the United States of America, listed below and have also identified below any foreign application(s) for patent or inventor's certificate or any PCT international application(s) having a filing date before that of the application(s) of which priority is claimed:

(Application Number)	(Filing Date)
(Application Number)	(Filing Date)
(Application Number)	(Filing Date)

I hereby claim the benefit under Title 35, United States Code, Section 120 of any United States application(s) or PCT international application(s) designating the United States of America that is/are listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in that/those prior application(s) in the manner provided by the first paragraph of Title 35, United States Code, Section 112, I acknowledge the duty to disclose material information as defined in Title 37, Code of Federal Regulations, Section 1.56(a) which occurred between the filing date of the prior application(s) and the national or PCT international filing date of this application.

PRIOR U.S. APPLICATIONS OR PCT INTERNATIONAL APPLICATIONS DESIGNATING THE U.S. FOR BENEFIT UNDER 35 U.S.C. 120

U.S. APPLICATIONS			STATUS (Check one)		
U.S. APPLICATION NUMBER	U.S. FILING DATE		PATENTED	PENDING	ABANDONED
09/539,667	March 30, 2000			X	
09/551,302	April 18, 2000			X	
PCT APPLICATIONS DESIGNATING THE U.S.					
PCT APPLICATION NO.	PCT FILING DATE	U.S. SERIAL NUMBERS ASSIGNED (if any)			

POWER OF ATTORNEY: As a named inventor, I hereby appoint the following attorney(s) and/or Agent(s) to prosecute this application and transact all business in the Patent and Trademark Office connected therewith.

Marc A. Sockol, Reg. No. 40,823; Daryl C. Josephson, Reg. No. 37,365; Arnold de Guzman, Reg. No. 39,955, Cameron Kerrigan, Reg. No. 44,826; Patrick D. Benedicto, Reg. No. 40,909; David B. Abel, Reg. No. 32,394; Nathan Lane, Reg. No. 43,738; Lorinda Howland, Reg. No. 42,671; Michael Lechter, Reg. No. 27,350; David Koo, Reg. No. 46,839; David Rogers, Reg. No. 38,287; William Bachand, Reg. No. 34,980; Aaron Wininger, Reg. No. 45,229; Paul A. Durdik, Reg. No. 37,819; Paul J. Meyer 47,791; Victoria L. Nicholson, Reg. No. 47,823; and Fariba Sirjani, Reg. No. 47,947.

Please direct all correspondence to: Daryl C. Josephson
Squire, Sanders & Dempsey L.L.P.
600 Hansen Way
Palo Alto, CA 94304-1043

Direct Phone Calls To: Daryl C. Josephson, 650- 856-6500

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issuing thereon.

1. **First Inventor's Name** Yigal (**First**) Mordechai (**Middle**) Edery (**Last Name**)

Citizenship Israel

Residence Hashikma 11, POB 1115, Pardesia 42815

(State/Foreign Country) Israel

First Inventor's Signature _____ **Date** 3/9/01

2. **Second Inventor's Name** Nimrod (**First**) Itzhak (**Middle**) Vered (**Last Name**)

Citizenship Israel

Residence Moshav Mismeret #81, Goosh Tel-Mond 40695

(State/Foreign Country) Israel

Post Office Address _____ **(Zip Code)** _____

Second Inventor's Signature _____ **Date** 3/Sept/01

3. **Third Inventor's Name** David R. Kroll
First **Middle** **Last Name**

Citizenship United States

Residence 4856 Kingbrook Dr., San Jose, CA 95124

(State/Foreign Country) United States

Post Office Address N/A **(Zip Code)** _____

Third Inventor's Signature _____ **Date** 8/27/01

#3

PATENT
ATTORNEY DOCKET NO. 43426.00014

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF MAILING

I hereby certify that this paper (along with any paper referred to as being attached or enclosed) is being deposited with the United States Postal Service on the date shown below with sufficient postage as first class mail in an envelope addressed to Commissioner for Patents, Washington, D.C. 20231, on

Date: 9/10/01 By: Sandy Yi
Sandy Yi

In re Application of: Examiner: Unknown

Yigal Edery, et al.

Serial No. 09/861,229 Art Unit: 2152

Filed: May 17, 2001

Title: MALICIOUS MOBILE CODE RUNTIME MONITORING SYSTEM AND METHODS

Commissioner for Patents
Washington, D.C. 20231

LETTER TO THE OFFICIAL DRAFTSPERSON
(Request to Substitute Drawings)

Sir:

Subject to the approval of the Primary Examiner in the above-entitled patent application, please substitute the enclosed ten (10) sheets of drawings, containing Figures 1a, 1b, 1c, 2, 3, 4, 5, 6a, 6b, 7a, 7b, 8, 9, 10a, 10b, 11, 12a, and 12b, for the ten (10) sheets of informal drawings containing Figures 1a, 1b, 1c, 2, 3, 4, 5, 6a, 6b, 7a, 7b, 8, 9, 10a, 10b, 11, 12a, and 12b as previously filed on May 17, 2001.

REMARKS

Applicants respectfully submit that the requested drawing substitution is consistent with the corresponding material in the specification and does not add any new matter to the application.

Should the Examiner have any questions concerning this request, the Examiner is invited to call the undersigned at the number shown below.

The Commissioner is hereby authorized to charge payment for any deficiency of required fees associated with this communication to Deposit Account **05-0150**.

Date: 9/10/01

Respectfully submitted,

Squire, Sanders & Dempsey L.L.P.
600 Hansen Way
Palo Alto, CA 94304-1043
Telephone (650) 856-6500
Facsimile (650) 843-8777

By: Daryl C. Josephson

Daryl C. Josephson
Attorney for Applicants
Registration No. 37,365

GAU21

#4

PTO/SB/21 (08-00)

Please type a plus sign (+) inside this box → [+]

Approved for use through 10/31/2002. OMB 0651-0031
U.S. Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE

TRANSMITTAL FORM

(to be used for all correspondence after initial filing)

Total Number of Pages in This Submission

Application Number	09/861,229
Filing Date	May 17, 2001
First Named Inventor	Yigal Edery, et al.
Group Art Unit	2152
Examiner Name	Unknown
Attorney Docket Number	43426.00014

ENCLOSURES *(check all that apply)*

- [x] Fee Transmittal Form
  - [ ] Fee Attached
  - [x] Deposit Account Authorization on Fee Transmittal Form
- [ ] Amendment / Response
  - [ ] After Final
- [ ] Signed Oath/Declaration
- [ ] Extension of Time Request
- [x] Return Postcard
- [x] Information Disclosure Statement (2 pages) & PTO Form 1449 (2 pages)
- [ ] Certified Copy of Priority Document(s)
- [ ] Response to Missing Parts
- [ ] Response to Incomplete Application
- [ ] Assignment & Cover Sheet *(for an Application)*
- [ ] Drawing(s) _____ sheets
- [ ] Licensing-related Papers
- [ ] Petition
- [ ] Petition to Convert to a Provisional Application
- [ ] Power of Attorney, Revocation Change of Correspondence Address
- [ ] Terminal Disclaimer
- [ ] Request for Refund
- [ ] CD, Number of CD(s) _____
- [ ] After Allowance Communication to Group
- [ ] Appeal Communication to Board of Appeals and Interferences
- [ ] Appeal Communication to Group *(Appeal Notice, Brief, Reply Brief)*
- [ ] Proprietary Information
- [ ] Status Letter
- [x] Other Enclosure(s) *(please identify below)*:

47 References

Remarks

RECEIVED
SEP 27 2001
Technology Center 2100

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT

Firm *or* Individual name	Marc A. Sockol, Reg. No. 40,823 Squire, Sanders & Dempsey, L.L.P. 600 Hansen Way Palo Alto, CA 94304-1043
Signature	M A Sockol
Date	September 17, 2001

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Assistant Commissioner for Patents, Washington, D.C. 20231 on this date: September 17, 2001

Typed or printed name	Sandy Yi		
Signature	Sandy Yi	Date	September 17, 2001

PTO/SB/17 (11-00)
Approved for use through 10/31/2002. OMB 0651-0032
U.S. Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE

FEE TRANSMITTAL for FY 2001

Patent fees are subject to annual revision.

#4

Complete if Known	
Application Number	09/861,229
Filing Date	May 17, 2001
First Named Inventor	Yigal Edery, et al.
Examiner Name	Unknown
Group / Art Unit	2152
Attorney Docket No.	43426.00014

TOTAL AMOUNT OF PAYMENT ($) 0

METHOD OF PAYMENT (check one)

1. ☒ The Commissioner is hereby authorized to charge indicated fees and credit any over payments to:

Deposit Account Number: 05-0150

Deposit Account Name: Squire, Sanders & Dempsey, L.L.P.

☒ Charge Any Additional Fee Required Under 37 CFR 1.16 and 1.17

☒ Applicant claims small entity status. See 37 CFR 1.27

2. ☐ Payment Enclosed:

☐ Check ☐ Credit card ☐ Money Order ☐ Other

FEE CALCULATION

1. BASIC FILING FEE

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description	Fee Paid
101	710	201	355	Utility filing fee	
106	320	206	160	Design filing fee	
107	490	207	245	Plant filing fee	
108	710	208	355	Reissue filing fee	
114	150	214	75	Provisional filing fee	

SUBTOTAL (1) ($) 0

2. EXTRA CLAIM FEES

			Extra Claims		Fee from below		Fee Paid
Total Claims		-20	= 0	X		=	0
Independent Claims		-3	= 0	X		=	0
Multiple Dependent				X		=	0

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description
103	18	203	9	Claims in excess of 20
102	80	202	40	Independent claims in excess of 3
104	270	204	135	Multiple dependent claim, if not paid
109	80	209	40	** Reissue independent claims over original patent
110	18	210	9	** Reissue claims in excess of 20 and over original patent

SUBTOTAL (2) ($) 0

**or number previously paid, if greater; For Reissues, see above

FEE CALCULATION (continued)

3. ADDITIONAL FEES

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description	Fee Paid
105	130	205	65	Surcharge - late filing fee or oath	
127	50	227	25	Surcharge - late provisional filing fee or cover sheet.	
139	130	139	130	Non-English specification	
147	2,520	147	2,520	For filing a request for reexamination	
112	920*	112	920*	Requesting publication of SIR prior to Examiner action	
113	1,840*	113	1,840*	Requesting publication of SIR after Examiner action	
115	110	215	55	Extension for reply within first month	
116	390	216	195	Extension for reply within second month	
117	890	217	445	Extension for reply within third month	
118	1,390	218	695	Extension for reply within fourth month	
128	1,890	228	945	Extension for reply within fifth month	
119	310	219	155	Notice of Appeal	
120	310	220	155	Filing a brief in support of an appeal	
121	270	221	135	Request for oral hearing	
138	1,510	138	1,510	Petition to institute a public use proceeding	
140	110	240	55	Petition to revive – unavoidable	
141	1,240	241	620	Petition to revive – unintentional	
142	1,240	242	620	Utility issue fee (or reissue)	
143	440	243	220	Design issue fee	
144	600	244	300	Plant issue fee	
122	130	122	130	Petitions to the Commissioner	
123	130	123	130	Petitions related to provisional applications	
126	180	126	180	Submission of Information Disclosure Stmt	
581	40	581	40	Recording each patent assignment per property (times number of properties)	
146	710	246	355	Filing a submission after final rejection (37 CFR § 1.129(a))	
149	710	249	355	For each additional invention to be examined (37 CFR § 1.129(b))	
179	710	279	355	Request for Continued Examination (RCE)	
169	900	169	900	Request for expedited examination of a design application	
				Other fee (specify) ____	

*Reduced by Basic Filing Fee Paid SUBTOTAL (3) ($) 0

SUBMITTED BY				Complete (if applicable)	
Name (Print/Type)	Marc A. Sockol	Registration No. Attorney/Agent)	40,823	Telephone	650.856.6500
Signature	M A Sockol			Date	September 17, 2001

WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.

Burden Hour Statement: This form is estimated to take 0.2 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Assistant Commissioner for Patents, Washington, DC 20231.

GAU21

#4

PTO/SB/21 (08-00)

Please type a plus sign (+) inside this box → [+]

Approved for use through 10/31/2002. OMB 0651-0031
U.S. Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE

TRANSMITTAL FORM

(to be used for all correspondence after initial filing)

Total Number of Pages in This Submission

Application Number	09/861,229
Filing Date	May 17, 2001
First Named Inventor	Yigal Edery, et al.
Group Art Unit	2152
Examiner Name	Unknown
Attorney Docket Number	43426.00014

ENCLOSURES *(check all that apply)*

- [x] Fee Transmittal Form
  - [ ] Fee Attached
  - [x] Deposit Account Authorization on Fee Transmittal Form
- [ ] Amendment / Response
  - [ ] After Final
- [ ] Signed Oath/Declaration
- [ ] Extension of Time Request
- [x] Return Postcard
- [x] Information Disclosure Statement (2 pages) & PTO Form 1449 (2 pages)
- [ ] Certified Copy of Priority Document(s)
- [ ] Response to Missing Parts
- [ ] Response to Incomplete Application
- [ ] Assignment & Cover Sheet *(for an Application)*
- [ ] Drawing(s) _____ sheets
- [ ] Licensing-related Papers
- [ ] Petition
- [ ] Petition to Convert to a Provisional Application
- [ ] Power of Attorney, Revocation Change of Correspondence Address
- [ ] Terminal Disclaimer
- [ ] Request for Refund
- [ ] CD, Number of CD(s) _____
- [ ] After Allowance Communication to Group
- [ ] Appeal Communication to Board of Appeals and Interferences
- [ ] Appeal Communication to Group *(Appeal Notice, Brief, Reply Brief)*
- [ ] Proprietary Information
- [ ] Status Letter
- [x] Other Enclosure(s) *(please identify below)*:

47 References

Remarks

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT

Firm *or* Individual name	Marc A. Sockol, Reg. No. 40,823 Squire, Sanders & Dempsey, L.L.P. 600 Hansen Way Palo Alto, CA 94304-1043
Signature	M A Sockol
Date	September 17, 2001

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Assistant Commissioner for Patents, Washington, D.C. 20231 on this date: September 17, 2001

Typed or printed name	Sandy Yi		
Signature	Sandy Yi	Date	September 17, 2001

PTO/SB/17 (11-00)
Approved for use through 10/31/2002. OMB 0651-0032
U.S. Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE

FEE TRANSMITTAL for FY 2001

Patent fees are subject to annual revision.

TOTAL AMOUNT OF PAYMENT ($) 0

Complete if Known	
Application Number	09/861,229
Filing Date	May 17, 2001
First Named Inventor	Yigal Edery, et al.
Examiner Name	Unknown
Group / Art Unit	2152
Attorney Docket No.	43426.00014

METHOD OF PAYMENT (check one)

1. ☒ The Commissioner is hereby authorized to charge indicated fees and credit any over payments to:

Deposit Account Number: 05-0150

Deposit Account Name: Squire, Sanders & Dempsey, L.L.P.

☒ Charge Any Additional Fee Required Under 37 CFR 1.16 and 1.17

☒ Applicant claims small entity status. See 37 CFR 1.27

2. ☐ Payment Enclosed:

☐ Check ☐ Credit card ☐ Money Order ☐ Other

FEE CALCULATION

1. BASIC FILING FEE

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description	Fee Paid
101	710	201	355	Utility filing fee	
106	320	206	160	Design filing fee	
107	490	207	245	Plant filing fee	
108	710	208	355	Reissue filing fee	
114	150	214	75	Provisional filing fee	

SUBTOTAL (1) ($) 0

2. EXTRA CLAIM FEES

			Extra Claims		Fee from below		Fee Paid
Total Claims		-20	= 0	X		=	0
Independent Claims		-3	= 0	X		=	0
Multiple Dependent				X		=	0

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description
103	18	203	9	Claims in excess of 20
102	80	202	40	Independent claims in excess of 3
104	270	204	135	Multiple dependent claim, if not paid
109	80	209	40	** Reissue independent claims over original patent
110	18	210	9	** Reissue claims in excess of 20 and over original patent

SUBTOTAL (2) ($) 0

**or number previously paid, if greater; For Reissues, see above

FEE CALCULATION (continued)

3. ADDITIONAL FEES

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description	Fee Paid
105	130	205	65	Surcharge - late filing fee or oath	
127	50	227	25	Surcharge - late provisional filing fee or cover sheet.	
139	130	139	130	Non-English specification	
147	2,520	147	2,520	For filing a request for reexamination	
112	920*	112	920*	Requesting publication of SIR prior to Examiner action	
113	1,840*	113	1,840*	Requesting publication of SIR after Examiner action	
115	110	215	55	Extension for reply within first month	
116	390	216	195	Extension for reply within second month	
117	890	217	445	Extension for reply within third month	
118	1,390	218	695	Extension for reply within fourth month	
128	1,890	228	945	Extension for reply within fifth month	
119	310	219	155	Notice of Appeal	
120	310	220	155	Filing a brief in support of an appeal	
121	270	221	135	Request for oral hearing	
138	1,510	138	1,510	Petition to institute a public use proceeding	
140	110	240	55	Petition to revive – unavoidable	
141	1,240	241	620	Petition to revive – unintentional	
142	1,240	242	620	Utility issue fee (or reissue)	
143	440	243	220	Design issue fee	
144	600	244	300	Plant issue fee	
122	130	122	130	Petitions to the Commissioner	
123	130	123	130	Petitions related to provisional applications	
126	180	126	180	Submission of Information Disclosure Stmt	
581	40	581	40	Recording each patent assignment per property (times number of properties)	
146	710	246	355	Filing a submission after final rejection (37 CFR § 1.129(a))	
149	710	249	355	For each additional invention to be examined (37 CFR § 1.129(b))	
179	710	279	355	Request for Continued Examination (RCE)	
169	900	169	900	Request for expedited examination of a design application	
				Other fee (specify) ____	

*Reduced by Basic Filing Fee Paid SUBTOTAL (3) ($) 0

SUBMITTED BY				Complete *(if applicable)*	
Name (Print/Type)	Marc A. Sockol	*Registration No. Attorney/Agent)*	40,823	*Telephone*	650.856.6500
Signature	M A Sockol			*Date*	September 17, 2001

WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.

Burden Hour Statement: This form is estimated to take 0.2 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Assistant Commissioner for Patents, Washington, DC 20231.

PATENT
Attorney Docket No.: 43426.00014

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF MAILING

I hereby certify that this paper (along with any paper referred to as being attached or enclosed) is being deposited with the United States Postal Service on the date shown below with sufficient postage as first class mail in an envelope addressed to Commissioner for Patents, Washington, D.C. 20231, on

Date: 9/17/01 By: Sandy Yi
Sandy Yi

In re Application of:	
Yigal Edery, et al.	Examiner: Unknown
Serial No.: 09/861,229	Art Unit: 2152
Filed: May 17, 2001	
Title: MALICIOUS MOBILE CODE RUNTIME MONITORING SYSTEM AND METHODS	

Commissioner for Patents
Washington, D.C. 20231

INFORMATION DISCLOSURE STATEMENT PURSUANT TO 37 C.F.R. §§1.97-1.98

Sir:

In accordance with the duty of disclosure under 37 C.F.R. §1.56 and pursuant to 37 C.F.R. §§1.97-1.98, Applicants hereby notify the U.S. Patent and Trademark Office of the references listed on the attached Form PTO-1449. One copy of each cited reference is submitted herewith.

The submission of the listed documents is not intended as an admission that any such document constitutes prior art against the claims of the present application. Applicants reserve the right to dispute any of the listed documents as prior art during examination. Furthermore, Applicants do not waive any right to take any action that would be appropriate to antedate or otherwise remove any listed document as a competent reference against the claims of the present application. The submission of this Information Disclosure Statement is not to be construed as a representation that a search has been made or that no other material information may exist.

The Examiner is requested to initial the enclosed Form PTO-1449 and return a copy thereof to the undersigned.

The present Information Disclosure Statement is being filed before receiving the first Office Action. Therefore, no certification under 37 C.F.R. §1.97(e) or fee under 37 C.F.R. §1.17(p) is required.

However, if for any reason an insufficient fee has been paid, please charge the insufficiency to Deposit Account No. **05-0150**.

Date: September 17, 2001

Respectfully submitted,

Squire, Sanders & Dempsey L.L.P.
600 Hansen Way
Palo Alto, CA 94304-1043
Telephone (650) 856-6500
Facsimile (650) 843-8777

Marc A. Sockol
Attorney for Applicant
Reg. No. 40,823

Please type a plus si[illegible] inside this box ⟶ [+]

PTO/SB/21 (08-00)
Approved for use through 10/31/2002. OMB 0651-0031
U.S. Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE
Under the Paperwo[illegible]uction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

2/5-2 #5 3-5-03

OIPE JC10 FEB 2 4 2003 PATENT & TRADEMARK OFFICE

TRANSMITTAL FORM

(to be used for all correspondence after initial filing)

Application Number	09/861,229
Filing Date	May 17, 2001
First Named Inventor	Yigal Edery
Group Art Unit	2152
Examiner Name	Unknown
Total Number of Pages in This Submission: 3	Attorney Docket Number: 43426.00014

ENCLOSURES *(check all that apply)*

- [ ] Fee Transmittal Form (in duplicate)
- [ ] Amendment / Response
  - [ ] With RCE
  - [ ] After Final
  - [ ] Affidavits/declaration(s)
- [ ] Extension of Time Request (in duplicate)
- [ ] ____ Reference(s)
- [ ] IDS and Form 1449
- [ ] Certified Copy of Priority Document(s)
- [ ] Response to Missing Parts/ Incomplete Application
- [ ] Declaration/Oath
- [ ] Assignment and Recordation Cover Sheet *(for an Application)*
- [ ] Drawing(s) ____ Sheets
- [ ] Licensing-related Papers
- [ ] Petition
- [ ] Request for Continued Examination
- [x] Associate Power of Attorney
- [ ] Terminal Disclaimer
- [ ] Request for Refund
- [ ] CD, Number of CD(s) _____
- [ ] Request to Correct Filing Receipt
- [ ] Appeal Communication to Board of Appeals and Interferences
- [ ] Appeal Communication to Group *(Appeal Notice, Brief, Reply Brief)*
- [x] Return Postcard
- [ ] Status Letter
- [ ] Other Enclosure(s) *(please identify below)*:

Remarks

RECEIVED FEB 2 7 2003 Technology Center 2100

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT

Firm *or* Individual name	Marc A. Sockol, Reg. No. 40,823 Squire, Sanders & Dempsey, L.L.P. 600 Hansen Way Palo Alto, CA 94304-1043
Signature	M. A. Sockol
Date	February 13, 2003

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Assistant Commissioner for Patents, Washington, D.C. 20231 on this date: February 13, 2003

Typed or printed name	Sandy Yi		
Signature	Sandy Yi	Date	February 13, 2003

Burden Hour Statement: This form is estimated to take 0.2 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be send to the Chief Information Officer, U.S. Patent and Trademark Office, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Assistant Commissioner for Patents, Washington, DC 20231.

PATENT
Attorney Docket No.: 43426.00014

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF MAILING

I hereby certify that this paper (along with any paper referred to as being attached or enclosed) is being deposited with the United States Postal Service on the date shown below with sufficient postage as first class mail in an envelope addressed to Commissioner for Patents, Washington, D.C. 20231, on

Date: 2-13-03 By: Sandy Yi
Sandy Yi

In re Application of:			
	Yigal Edery, et al.	Examiner:	Unknown
Serial No.:	09/861,229	Art Unit:	2152
Filed:	May 17, 2001		
Title:	MALICIOUS MOBILE CODE RUNTIME MONITORING SYSTEM AND METHODS		

Commissioner of Patents
Washington, DC 20231

ASSOCIATE POWER OF ATTORNEY

Sir:

Please recognize the following attorney as an associate attorney in the above-referenced application:

Marc A. Berger, Reg. No. 44,029.

Please continue to address all correspondence and communications to:

Marc A. Sockol
Customer No. 30256
Squire, Sanders & Dempsey L.L.P.
600 Hansen Way
Palo Alto, CA 94304-1043
650-856-6500

Dated: 2-13-03

Respectfully submitted

Squire, Sanders & Dempsey L.L.P
600 Hansen Way
Palo Alto, CA 94304-1043
Tel (650) 856-6500
Fax (650) 843-8777

By: M. A. Sockol

Marc A. Sockol
Attorney for Applicants
Registration No. 40,823

PaloAlto Doc #: 49232v1

2152 #6 SP 7-22-03

PTO/SB/21 (05-03)
Approved for use through 04/30/2003. OMB 0651-0031
U.S. Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE

TRANSMITTAL FORM

(to be used for all correspondence after initial filing)

Application Number	09/861,229
Filing Date	May 17, 2001
First Named Inventor	Yigal Edery
Art Unit	2152
Examiner Name	Unknown
Total Number of Pages in This Submission: N/A	Attorney Docket Number: 43426.00014

ENCLOSURES *(check all that apply)*

- ☒ Fee Transmittal Form (in duplicate)
- ☐ Request for Corrected Filing Receipt
- ☐ Amendment / Response
  - ☐ After Final
  - ☐ With RCE
- ☐ Extension of Time Request
- ☒ Return Postcard
- ☒ Supplemental Information Disclosure Statement (2 pages)
- ☒ PTO Form 1449 (2 pages)
- ☒ 9 References
- ☐ Declaration/Oath
- ☐ Assignment and Recordation Cover Sheet *(for an Application)*
- ☐ Drawing(s) ____ Sheets
- ☐ Licensing-related Papers
- ☐ Petition
- ☐ RCE
- ☐ Power of Attorney, Revocation Change of Correspondence Address
- ☐ Terminal Disclaimer
- ☐ Request for Refund
- ☐ CD, Number of CD(s) ____
- ☐ After Allowance Communication to Group
- ☐ Appeal Communication to Board of Appeals and Interferences
- ☐ Appeal Communication to Group *(Appeal Notice, Brief, Reply Brief)*
- ☐ Proprietary Information
- ☐ Status Request
- ☐ Other Enclosure(s) *(please identify below)*:

Remarks

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT

Firm *or* Individual name	Marc A. Sockol, Reg. No. 40,823 Squire, Sanders & Dempsey L.L.P. 600 Hansen Way Palo Alto, CA 94304-1043
Signature	
Date	July 11, 2003

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being facsimile transmitted to the USPTO or deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on the date shown below.

Typed or printed name	Sandy Yi		
Signature	Sandy Yi	Date	July 11, 2003

This collection of information is required by 37 CFR 1.5. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

PTO/SB/17 (05-03)
Approved for use through 04/30/2003. OMB 0651-0032
U.S. Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE

FEE TRANSMITTAL for FY 2003

Effective 01/01/2003. Patent fees are subject to annual revision.

☐ Applicant claims small entity status. See 37 CFR 1.27

TOTAL AMOUNT OF PAYMENT ($) 0

Complete if Known	
Application Number	09/861,229
Filing Date	May 17, 2001
First Named Inventor	Yigal Edery
Examiner Name	Unknown
Art Unit	2152
Attorney Docket No.	43426.00014

METHOD OF PAYMENT (check all that apply)

☐ Check ☐ Credit card ☐ Money Order ☐ Other ☐ None

☒ Deposit Account:

Deposit Account Number: 05-0150

Deposit Account Name: Squire, Sanders & Dempsey L.L.P.

The Director is authorized to: *(check all that apply)*
☐ Charge fee(s) indicated below ☒ Credit any overpayments
☒ Charge any additional fee(s) during the pendency of this application
☐ Charge fee(s) indicated below, **except for the filing fee** to the above-identified deposit account.

FEE CALCULATION

1. BASIC FILING FEE

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description	Fee Paid
1001	750	2001	375	Utility filing fee	
1002	330	2002	165	Design filing fee	
1003	520	2003	260	Plant filing fee	
1004	750	2004	375	Reissue filing fee	
1005	160	2005	80	Provisional filling fee	

SUBTOTAL (1) ($) 0

2. EXTRA CLAIM FEES

			Extra Claims		Fee from below		Fee Paid
Total Claims		-20 **	= 0	X		=	0
Independent Claims		-3 **	= 0	X		=	0
Multiple Dependent				X		=	0

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description
1202	18	2202	9	Claims in excess of 20
1201	84	2201	42	Independent claims in excess of 3
1203	280	2203	140	Multiple dependent claim, if not paid
1204	84	2204	42	** Reissue independent claims over original patent
1205	18	2205	9	** Reissue claims in excess of 20 and over original patent

SUBTOTAL (2) ($) 0

***or number previously paid, if greater; For Reissues, see above*

FEE CALCULATION (continued)

3. ADDITIONAL FEES

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description	Fee Paid
1051	130	2051	65	Surcharge - late filing fee or oath	
1052	50	2052	25	Surcharge - late provisional filing fee or cover sheet.	
1053	130	1053	130	Non-English specification	
1812	2,520	1812	2,520	For filing a request for reexamination	
1804	920*	1804	920*	Requesting publication of SIR prior to Examiner action	
1805	1,840*	1805	1,840*	Requesting publication of SIR after Examiner action	
1251	110	2251	55	Extension for reply within first month	
1252	410	2252	205	Extension for reply within second month	
1253	930	2253	465	Extension for reply within third month	
1254	1,450	2254	725	Extension for reply within fourth month	
1255	1,970	2255	985	Extension for reply within fifth month	
1401	320	2401	160	Notice of Appeal	
1402	320	2402	160	Filing a brief in support of an appeal	
1403	280	2403	140	Request for oral hearing	
1451	1,510	1451	1,510	Petition to institute a public use proceeding	
1452	110	2452	55	Petition to revive – unavoidable	
1453	1,300	2453	650	Petition to revive – unintentional	
1501	1,300	2501	650	Utility issue fee (or reissue)	
1502	470	2502	235	Design issue fee	
1503	630	2503	315	Plant issue fee	
1460	130	1460	130	Petitions to the Commissioner	
1807	50	1807	50	Processing fee under 37 CFR 1.17 (q)	
1806	180	1806	180	Submission of Information Disclosure Stmt	
8021	40	8021	40	Recording each patent assignment per property (times number of properties)	
1809	750	2809	375	Filing a submission after final rejection (37 CFR § 1.129(a))	
1810	750	2810	375	For each additional invention to be examined (37 CFR § 1.129(b))	
1801	750	2801	375	Request for Continued Examination (RCE)	
1802	900	1802	900	Request for expedited examination of a design application	
Other fee (specify) ____					

*Reduced by Basic Filing Fee Paid

SUBTOTAL (3) ($) 0

SUBMITTED BY				Complete *(if applicable)*	
Name (Print/Type)	Marc A. Sockol	*Registration No. Attorney/Agent)*	40,823	*Telephone*	650.856.6500
Signature				*Date*	July 11, 2003

WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.

This collection of information is required by 37 CFR 1.17 and 1.27. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing this form, call 1-800-PTO-9199 (1-800-786-9199) and select option 2.

PATENT
Attorney Docket No.: 43426.00014

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:			
	Yigal Edery, et al.	Examiner:	Unknown
Serial No.:	09/861,229	Art Unit:	2152
Filed:	May 17, 2001		
Title:	MALICIOUS MOBILE CODE RUNTIME MONITORING SYSTEM AND METHODS		

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

SUPPLEMENTAL INFORMATION DISCLOSURE STATEMENT PURSUANT TO 37 C.F.R. §§1.97(b)

Sir:

In accordance with the duty of disclosure under 37 CFR §1.56 and pursuant to 37 CFR §§1.97-1.98, Applicants hereby notify the U.S. Patent and Trademark Office of the references listed on the enclosed Form PTO-1449. One copy of each reference cited is submitted herewith.

The present Supplemental Information Disclosure Statement is being filed more than three months after the filing date but before receiving the first Office Action. Accordingly, no fee or certification is needed.

The submission of the listed documents is not intended as an admission that any such document constitutes prior art against the claims of the present application. Applicants reserve the right to dispute any of the listed documents as prior art during examination. Furthermore, Applicants do not waive any right to take any action that would be appropriate to antedate or otherwise remove any listed document as a competent reference against the claims of the present application. The submission of this Supplemental Information Disclosure Statement is not to be

construed as a representation that a search has been made or that no other material information may exist.

The Examiner is requested to initial the enclosed Form PTO-1449 and return a copy thereof to the undersigned.

If for any reason an insufficient fee has been paid, please charge the insufficiency to Deposit Account No. **05-0150**.

Date: July 11, 2003

Respectfully submitted,

Squire, Sanders & Dempsey L.L.P.
600 Hansen Way
Palo Alto, CA 94304-1043
Telephone (650) 856-6500
Facsimile (650) 843-8777

By: _____
Marc A. Sockol
Attorney for Applicants
Reg. No. 40,823

CERTIFICATE OF MAILING

I hereby certify that this paper (along with any paper referred to as being attached or enclosed) is being deposited with the United States Postal Service on the date shown below with sufficient postage as first class mail in an envelope addressed to the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on

Date: July 11, 2003 By: Sandy Yi
Sandy Yi

	Type	L #	Hits	Search Text	DBs	Time Stamp
1	BRS	L1	135368	(code or executable or download$5 or applet or java or script or activex)near10(determin$5 or ascertain$3 or monitor$3 or analy$4 or inspect$3 or examin$5)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	2004/12/04 13:02
2	BRS	L2	7046	1 same(secure or environment or shell or sandbox or protect$3)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	2004/12/04 13:00
3	BRS	L3	199947	(transmi$5 or send$3 or sent or communicat$3 or forward$3)near10(secure or environment or shell or sandbox or protect$3)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	2004/12/04 13:01
4	BRS	L4	820	2 same 3	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	2004/12/04 13:01
5	BRS	L5	172333	(code or executable or download$5 or applet or java or script or activex)near10(append$3 or attach$5 or indicat$3 or profile or character$5)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	2004/12/04 13:03