

service-program, it may be integrated into common OS components, or it may be a replacement of some standard input element.

[0068] The invention provides a protection against manipulation of SCRs by offenders. More particularly, the invention provides a sensor that examines the activity of the shared stacks and SCRs, and if a suspected activity within the said shared resources is detected, an alert to the legal user is provided. For example, in a first embodiment of the invention used for a first given range of services (which will be referred to hereinafter as the first setting, there is provided a sensor that can detect whether an extra SCR has been added to a stack, without determining the identity of the added SCR. The aforementioned sensor is implemented as an SCR (probe) that is added to a stack chain, in a similar manner to the tainted SCRs it is supposed to detect. The sensor uses the fact that it is inserted and positioned first, when possible, within the stack chain and then, when the stack is activated, it is inserted into the address space of the user's process and thus can monitor the activity of the stack. For example, under Microsoft Windows™, the first setting is typical with window messages and message-hooks, that deal mainly with defining an application's behavior and its reaction to standard OS messages. The sensor of the invention therefore searches for unauthorized breaches within the activity of these messages.

[0069] According to another embodiment of the invention, a sensor for detecting unauthorized activities within a second range of services, referred to herein as the second setting, is provided. The sensor in that case enumerates the chain within each stack, and detects unauthorized activities in these stacks. More particularly, the sensor for the second setting enables a user to obtain a list of the SCRs within each stack chain, and the SCRs relative position within the chain. Whenever a new SCR is detected within the chain, when the stack is activated, the sensor initiates an alert. For example, under Microsoft Windows™, second setting is typical with WOSA implementations, which deal mainly with information delivery. Although an operation system, such as the Windows NT, would normally prevent a user without appropriate permission from installing a new Winsock provider, the more ubiquitous versions of Microsoft Windows (e.g., Windows 98) would not prevent it. Furthermore, even under Windows NT, when File Allocation Table (FAT) is used instead of the New Technology File System (NTFS), a common user is free to manipulate the system and cause hostile applications and modules to activate when an administrator logs in; this obviously circumvents the prior limitation.

[0070] FIG. 2A schematically illustrates the overall architecture of a sensor for detecting breaches within shared resources. The sensor comprises the sensor main unit 25, a probe 26 for sampling the stack activity, wherein the probe is the SCR that is installed within the stack chain, a decision unit 29 and a communication unit 24. The sensor itself communicates and compares information with knowledge base 150, that as said includes authentication information relating to the expected structure and activity of each stack.

[0071] The overall architecture may further comprise one or more fellow agents 28 that can be used, generally, for notifying other systems about the detection of a suspected SCR, or notifying the system about suspected signs. The

sensor's main unit 25 may communicate with external fellow agents 28 via the communication unit 25, for: (i) posting each state-transition, from a currently activated SCR to the next SCR in the chain, to form a log queue. (ii) Alerting the decision unit 29, or a human user when a predefined threshold condition is met, i.e., a suspected SCR is detected. (iii) Receiving instructions from the decision unit 29. (iv) Accepting new weights from an agent 28 (or the human user). (v) Receiving load/unload commands from a risk-assessor (not shown) and a load-balancing agent (not shown). This is useful for (I) eliminating false alarms, and (II) economize the usage of limited system resources. Fellow agents 28 may reside either within the same machine, or somewhere within the network.

[0072] As will be further discussed hereinafter, probe 26 has a different implementation for each setting (i.e., first or second settings), scenario or operation-mode.

[0073] The sensor's main unit 25 creates and activates probe 26 as said, which is an SCR inserted into the stack chain. After the activation of probe 26, the sensor main unit 25 enters into a waiting state in which it waits for notifications from probe 26 on suspicious SCRs, when detected. Probe 26 operate as follows:

[0074] it waits for signals, in this context, 'signals' are indications given by the OS, concerning some state transitions. Since each sensor of the invention is responsible for checking just a limited range of state transitions, a reference is made herein to a 'range of signals' that a given sensor should handle.

[0075] upon detecting a signal transition, probe 26 evaluates the specific signal transition in order to detect whether the transition is suspicious or not. The evaluation process is described hereinafter.

[0076] if the signal is found to be suspicious, then the probe 26 notifies the main unit 25, which in turn performs one or more of the following procedures: it freezes the suspected signal stack, it continuous monitoring of the stack, or it initiates an alert.

[0077] Probe 26 receives updates from the main unit 25 on its desired mode of operation. The mode of operation may either be user defined, or dominated by self learned rules. It uses relatively fast heuristics to determine if a monitored signal should be treated as suspected. If the heuristics indicate a state transition, probe 26 flushes a dedicated cache of state-records, which it keeps, for a more persistent storage, available also to sensor 25 and/or to fellow agent 28. If the heuristics indicate that the current state requires intervention, probe 26 freezes the suspected SCR 27 and, possibly, the whole offended process as well, and it notifies the sensor's main unit 25.

[0078] FIG. 2B schematically illustrates the implementation of sensors within a system 10, according to an embodiment of the invention. Block 1 indicates the calling of a stack by the kernel of the OS. Following this call, the stack is activated, including its chain 179 of SCRs. SCR 21 is a valid SCR. The SCR probe 26, which is a part of the sensor of the invention, is indicated as numeral 26. Numeral 27 indicates a tainted SCR, which is implanted by an offender. The call interface 4 interfaces between the active chain 179 and the application via sensor 41, which is capable of freezing the