

105 of user-mode SCR. A plurality of SCR stacks are generally contained within the library **105**, wherein each stack contains a plurality of SCRs organized in a chain-like form. In the case of Windows, such SCRs are DLL files. The SCR stacks are generally available to service any process in the system for carrying out specific tasks. For example, when a first process **101** activates a stack **103** of SCRs from the library **105**, the operating system creates a virtual mapping of stack **103**, so that the first process actually sees in its memory space **120** a copy **113** of stack **103**, and activates the same. Generally, each stack contains a plurality of SCRs, and the process selectively uses only one or few SCRs from each stack, but the whole stack is activated. If a second process **102**, needs a service from the same stack **103**, the same procedure repeats, and a virtual mapping **123** of stack **103** is produced in the memory space **121** of the second process. More particularly, the first process sees a virtual copy **113** of the stack **103** within its memory space **120**, and the second process sees a virtual copy **123** of stack **103** within its memory space **121**, however, both actually operates the same stack **103**. Security considerations require a total separation between the virtual memory **120** of the first process **101**, and the virtual memory **121** the second process **102**. However, as will be shown, this is not always the case. The fact that the same library stacks, in this example stack **103**, are shared by more than one process, enables a user of a second process to breach the memory space of a first process, by altering an SCR stack that is used by said first process. This is generally done by creating an SCR, and implanting it within a stack that supposed to be used by said first process. Whenever said altered stack is activated by said first process, the implanted SCR is also activated. The implanted SCR can perform essentially any activity within the memory space of said first process. For example, if a user of the second process implants an SCR within stack **103**, said SCR will appear in the memory space of any process that will use stack **103** in the future, for example as SCR **116** within the memory space **120**.

[0061] FIG. 1B illustrates, in block diagram form, a typical architecture of an extensible multi-layered system **10**. FIG. 1B illustrates the relevant portions of system **10**. Typically, system **10** makes a distinction between kernel-mode and user-mode, regarding the memory allocation. Such a distinction is indicated by dotted line **20**. When any of the processes **51**, **52**, or **53** wishes to get a service from an SCR stack, it calls the OS (not shown), which in turn activates the stack. Each process maps the stack to its isolated memory space, however only one copy of the stack exists within the physical memory. The SCR stack manager **3** manages the operation of the stack, and the Call Interface **4** interfaces between the stack **56** and each process, for example **51**, **52** or **53**, that needs a stack service. Call Interface **4** implies that a specific software module (not shown) is available in the system to activate a stack and map it to each process. The Call Interface **4** is sometimes called an Application Program Interface (API).

[0062] As said, essentially all the existing operating systems allow a user to add one or more SCRs to any shared stack, for enhancing the services he receives from the stack. A person, who, unfortunately, can be a possible offender, can add a new SCR to an extensible stack, for example a Winsock stack. In that way, the added SCR within the relevant stack would be available to any process requiring the service of said stack in user-mode. After activating the

stack by a specific user, or by the offender in a manner of inserting the stack into the memory space of that specific user, the added SCR can perform any task as designated by the offender. For example, if a specific application process is directed to store in memory any character typed on a keyboard, and the added SCR is programmed to read from that memory storage and transmit the content to the process of the offender, then the offender will be informed on whatever typed by the user on his keyboard.

[0063] According to an embodiment of the present invention, at least one sensor (or array of sensors) **140** is provided for detecting breach activities by means of illegally using SCRs, and for preventing such breach activities. FIG. 1C illustrates how such sensor activation is provided. Initially, a knowledge base **150** is prepared, which contains authentication information regarding each stack that supposed to be checked. For example, for stack **130** the knowledge base **150** contains at least the list of all the SCRs within the stack, and the last date of their updating. Within each stack **130-132** of library **139**, an authenticating SCR **136** is implanted by the valid user. Authenticating SCR **136** hereinafter also referred to also as "probe". This SCR is implanted in such a manner as to be activated any time when the stack is called, so preferably it should be implanted as close as possible to the beginning of the stack chain. Then, whenever a stack is activated, for example stack **130**, the authenticating SCR **136** activates the sensor **140**, which is a piece of code, the purpose of which is to check the authenticity of the stacks of library **139**, and to detect any unauthorized action within the stacks of it. When the stack is activated, the sensor **140** checks the authenticity of the stack operation, and its structure. The sensor **140** performs this operation by means of comparing the stack activity and its structure with the expected parameters as stored in the knowledge base **150** for the same stack. For example, whenever stack **130** is activated, sensor **140** compares the SCR found in said stack **130** with the list of SCRs expected to be included within this stack. Whenever a new SCR is found within the stack, and absolutely if such a new SCR is activated, an alert is sent, and optionally its activation is frozen or inhibited as defined, until a further decision is made. It should be noted that knowledge base **150** should preferably be dynamically updated with new information regarding the structure and authenticity of the stacks, and regarding suspected parameters or signs that should be particularly checked. In one embodiment of the invention, the sensor **140** is a "public" sensor. A public sensor is a unit, which is common to all the stacks within library **139**. In another embodiment of the invention, a plurality of dedicated "private" sensors are provided, one for each stack in the library. In such a case, each authenticating SCR **136** includes the sensor **140**, and the authenticating parameters (as stored in knowledge base **150**) relevant to the one stack supposed to be monitored.

[0064] It is important to note that:

[0065] (i) A sensor may encounter different scenarios, depending mainly upon the behavior of the offender, whether it is an eavesdropper, a manipulator, or another;

[0066] (ii) Each scenario may comprise many states, either consequently or concurrently; and

[0067] (iii) There may be multiple operation-modes of a sensor: it may be implemented as an independent