

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

<b>UTILITY                  PATENT APPLICATION                  TRANSMITTAL</b>  <i>(Only for new nonprovisional applications under 37 C.F.R. 1.53(b))</i>	Attorney Docket No.	FIN0008-DIV1
	First Inventor	David GRUZMAN, et al.
	Title	System and Method for Inspecting Dynamically Generated Executable code
	Express Mail Label No.	

<b>APPLICATION ELEMENTS</b> <i>See MPEP chapter 600 concerning utility patent application contents.</i>	<b>ADDRESS TO:</b> Commissioner for Patents P.O. Box 1450 Alexandria VA 22313-1450
--	--

1.  **Fee Transmittal Form (e.g., PTO/SB/17)**  
*(Submit an original and a duplicate for fee processing)*
2.  **Applicant claims small entity status.**  
 See 37 CFR 1.27.
3.  **Specification** [Total Pages 37 ]  
 Both the claims and abstract must start on a new page  
*(For information on the preferred arrangement, see MPEP 608.01(a))*
4.  **Drawing(s)** (35 U.S.C. 113) [Total Sheets 5 ]
5. **Oath or Declaration** [Total Sheets 6 ]
  - a.  Newly executed (original or copy)
  - b.  A copy from a prior application (37 CFR 1.63 (d))  
*(for a continuation/divisional with Box 18 completed)*
  - i.  **DELETION OF INVENTOR(S)**  
 Signed statement attached deleting inventor(s)  
 named in the prior application, see 37 CFR  
 1.63(d)(2) and 1.33(b).
6.  **Application Data Sheet.** See 37 CFR 1.76
7.  **CD-ROM or CD-R** in duplicate, large table or  
 Computer Program (*Appendix*)  
 Landscape Table on CD
8. **Nucleotide and/or Amino Acid Sequence Submission**  
*(if applicable, items a.-c. are required)*
  - a.  Computer Readable Form (CRF)
  - b.  Specification Sequence Listing on:
    - i.  CD-ROM or CD-R (2 copies); or
    - ii.  Paper
  - c.  Statements verifying identity of above copies

<b>ACCOMPANYING APPLICATIONS PARTS</b>	
9. <input checked="" type="checkbox"/> <b>Assignment Papers</b> (cover sheet & document(s)) Name of Assignee <u>Finjan, Inc.</u>	
10. <input checked="" type="checkbox"/> <b>37 C.F.R. 3.73(b) Statement</b> <input checked="" type="checkbox"/> <b>Power of Attorney</b> <i>(when there is an assignee)</i>	
11. <input type="checkbox"/> <b>English Translation Document</b> <i>(if applicable)</i>	
12. <input checked="" type="checkbox"/> <b>Information Disclosure Statement</b> (PTO/SB/08 or PTO-1449) <input type="checkbox"/> Copies of citations attached	
13. <input type="checkbox"/> <b>Preliminary Amendment</b>	
14. <input type="checkbox"/> <b>Return Receipt Postcard</b> (MPEP 503) <i>(Should be specifically itemized)</i>	
15. <input type="checkbox"/> <b>Certified Copy of Priority Document(s)</b> <i>(if foreign priority is claimed)</i>	
16. <input type="checkbox"/> <b>Nonpublication Request</b> under 35 U.S.C. 122(b)(2)(B)(i). Applicant must attach form PTO/SB/35 or equivalent.	
17. <input checked="" type="checkbox"/> <b>Other: Filed Electronically</b>	

18. If a CONTINUING APPLICATION, check appropriate box, and supply the requisite information below and in the first sentence of the specification following the title, or in an Application Data Sheet under 37 CFR 1.76:

<input type="checkbox"/> Continuation	<input checked="" type="checkbox"/> Divisional	<input type="checkbox"/> Continuation-in-part (CIP)	of prior application No: <u>11 / 298,475</u>
Prior application information: <u>Examiner Ponnoreay Pich</u>		Art Unit: <u>2435</u>	

**19. CORRESPONDENCE ADDRESS**

The address associated with Customer Number 74877 OR  Correspondence address below

Name			
Address			
City	State	Zip Code	
Country	Telephone	Email	

Signature	/Dawn-Marie Bey/	Date	June 9, 2010
Name (Print/Type)	Dawn-Marie Bey	Registration No. (Attorney/Agent)	44,442

This collection of information is required by 37 CFR 1.53(b). 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 1.11 and 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: Mail Stop Patent Application, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

*If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.*

## SYSTEM AND METHOD FOR INSPECTING DYNAMICALLY GENERATED EXECUTABLE CODE

### FIELD OF THE INVENTION

[0001] The present invention relates to computer security, and more particularly to protection against malicious code such as computer viruses.

### BACKGROUND OF THE INVENTION

[0002] Computer viruses have been rampant for over two decades now. Computer viruses generally come in the form of executable code that performs adverse operations, such as modifying a computer's operating system or file system, damaging a computer's hardware or hardware interfaces, or automatically transmitting data from one computer to another. Generally, computer viruses are generated by hackers willfully, in order to exploit computer vulnerabilities. However, viruses can also arise by accident due to bugs in software applications.

[0003] Originally computer viruses were transmitted as executable code inserted into files. As each new viruses was discovered, a signature of the virus was collected by anti-virus companies and used from then on to detect the virus and protect computers against it. Users began routinely scanning their file systems using anti-virus software, which regularly updated its signature database as each new virus was discovered.

[0004] Such anti-virus protection is referred to as "reactive", since it can only protect in reaction to viruses that have already been discovered.

[0005] With the advent of the Internet and the ability to run executable code such as scripts within Internet browsers, a new type of virus formed; namely, a virus that enters a computer over the Internet and not through the computer's file system. Such Internet viruses can be embedded within web pages and other web content, and begin executing within an Internet browser as soon as they enter a computer. Routine file scans are not able to detect such viruses, and as a result more sophisticated anti-virus tools had to be developed.

[0006] Two generic types of anti-virus applications that are currently available to protect against such Internet viruses are (i) gateway security applications, and (ii) desktop security applications. Gateway security applications shield web content before the content is delivered to its intended destination computer. Gateway security applications scan web content, and block the content from reaching the destination computer if the content is deemed by the security application to be potentially malicious. In distinction, desktop security applications shield against web content after the content reaches its intended destination computer.

[0007] Moreover, in addition to reactive anti-virus applications, that are based on databases of known virus signatures, recently "proactive" antivirus applications have been developed. Proactive anti-virus protection uses a methodology known as "behavioral analysis" to analyze computer content for the presence of viruses. Behavior analysis is used to automatically scan and parse executable content, in order to detect which computer operations the content may perform. As such, behavioral analysis can block viruses that have not been previously detected and which do not have a signature on record, hence the name "proactive".

[0008] Assignee's US Patent No. 6,092,194 entitled SYSTEM AND METHOD FOR PROTECTING A COMPUTER AND A NETWORK FROM HOSTILE DOWNLOADABLES, the contents of which are hereby incorporated by reference, describes gateway level behavioral analysis. Such behavioral analysis scans and parses content received at a gateway and generates a security profile for the content. A security profile is a general list or delineation of suspicious, or potentially malicious, operations that executable content may perform. The derived security profile is then compared with a security policy for the computer being protected, to determine whether or not the content's security profile violates the computer's security policy. A security policy is a general set of simple or complex rules, that may be applied logically in series or in parallel, which determine whether or not a specific operation is permitted or forbidden to be performed by the content on the computer being protected. Security policies are generally configurable, and set by an administrator of the computer that are being protected.

[0009] Assignee's US Patent No. 6,167,520 entitled SYSTEM AND METHOD FOR PROTECTING A CLIENT DURING RUNTIME FROM HOSTILE DOWNLOADABLES, the contents of which are hereby incorporated by reference, describes desktop level behavioral analysis. Desktop level behavioral analysis is generally implemented during runtime, while a computer's web browser is processing web content received over the Internet. As the content is being processed, desktop security applications monitor calls made to critical systems of the computer, such as the operating system, the file system and the network system. Desktop security applications use hooks to intercept calls made to operating system functions, and allow or block the calls as appropriate, based on the computer's security policy.

[00010] Each of the various anti-virus technologies, gateway vs. desktop, reactive vs. proactive, has its pros and cons. Reactive anti-virus protection is computationally simple and fast; proactive virus protection is computationally intensive and slower. Reactive anti-virus protection cannot protect against new "first-time" viruses, and cannot protect a user if his signature file is out of date; proactive anti-virus protection can protect against new "first-time" viruses and do not require regular downloading of updated signature files. Gateway level protection keeps computer viruses at a greater distance from a local network of computers; desktop level protection is more accurate. Desktop level protection is generally available in the consumer market for hackers to obtain, and is susceptible to reverse engineering; gateway level protection is not generally available to hackers.

[00011] Reference is now made to FIG. 1, which is a simplified block diagram of prior art systems for blocking malicious content, as described hereinabove. The topmost system shown in FIG. 1 illustrates a gateway level security application. The middle system shown in FIG. 1 illustrates a desktop level security application, and the bottom system shown in FIG. 1 illustrates a combined gateway + desktop level security application.

[00012] The topmost system shown in FIG. 1 includes a gateway computer 105 that receives content from the Internet, the content intended for delivery to a client computer 110. Gateway computer 105 receives the content over a communication channel 120, and gateway

computer communicates with client computer 110 over a communication channel 125. Gateway computer 105 includes a gateway receiver 135 and a gateway transmitter 140. Client computer 110 includes a client receiver 145. Client computer generally also has a client transmitter, which is not shown.

[00013] Client computer **110** includes a content processor **170**, such as a conventional web browser, which processes Internet content and renders it for interactive viewing on a display monitor. Such Internet content may be in the form of executable code, JavaScript, VBScript, Java applets, ActiveX controls, which are supported by web browsers.

[00014] Gateway computer **105** includes a content inspector **174** which may be reactive or proactive, or a combination of reactive and proactive. Incoming content is analyzed by content inspector **174** before being transmitted to client computer **110**. If incoming content is deemed to be malicious, then gateway computer **105** preferably prevents the content from reaching client computer **110**. Alternatively, gateway computer **105** may modify the content so as to render it harmless, and subsequently transmit the modified content to client computer **110**.

[00015] Content inspector **174** can be used to inspect incoming content, on its way to client computer **110** as its destination, and also to inspect outgoing content, being sent from client computer **110** as its origin.

[00016] The middle system shown in **FIG. 1** includes a gateway computer **105** and a client computer **110**, the client computer **110** including a content inspector **176**. Content inspector **176** may be a conventional Signature-based anti-virus application, or a run-time behavioral based application that monitors run-time calls invoked by content processor **170** to operating system, file system and network system functions.

[00017] The bottom system shown in **FIG. 1** includes both a content inspector **174** at gateway computer **105**, and a content inspector **176** at client computer **110**. Such a system can support conventional gateway level protection, desktop level protection, reactive anti-virus protection and proactive anti-virus protection.

[00018] As the hacker vs. anti-virus protection battle continues to wage, a newer type of virus has sprung forward; namely, dynamically generated viruses. These viruses are themselves generated only at run-time, thus thwarting conventional reactive analysis and conventional gateway level proactive behavioral analysis. These viruses take advantage of features of dynamic HTML generation, such as executable code or scripts that are embedded within HTML pages, to generate themselves on the fly at runtime.

[00019] For example, consider the following portion of a standard HTML page:

---

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML>
<SCRIPT LANGUAGE="JavaScript">

document.write("<hl>text that is generated at run-time</hl>");

</SCRIPT>
<BODY>

</BODY>
</HTML>
```

---

The text within the `<SCRIPT>` tags is JavaScript, and includes a call to the standard function `document.write()`, which generates dynamic HTML. In the example above, the function `document.write()` is used to generate HTML header text, with a text string that is generated at run-time. If the text string generated at run-time is of the form

```
<SCRIPT>malicious JavaScript</SCRIPT>
```

then the `document.write()` function will insert malicious JavaScript into the HTML page that is currently being rendered by a web browser. In turn, when the web browser processes the inserted text, it will perform malicious operations to the client computer.

[0020] Such dynamically generated malicious code cannot be detected by conventional reactive content inspection and conventional gateway level behavioral analysis content inspection, since the malicious JavaScript is not present in the content prior to run-time. A content inspector will only detect the presence of a call to *Document.write()* with input text that is yet unknown. If such a content inspector were to block all calls to *Document.write()* indiscriminately, then many harmless scripts will be blocked, since most of the time calls to *Document.write()* are made for dynamic display purposes only.

[0021] US Patent Nos. 5,983,348 and 6,272,641, both to Ji, describe reactive client level content inspection, that modifies downloaded executable code within a desktop level anti-virus application. However, such inspection can only protect against static malicious content, and cannot protect against dynamically generated malicious content.

[0022] Desktop level run-time behavioral analysis has a chance of shielding a client computer against dynamically generated malicious code, since such code will ultimately make a call to an operating system function. However, desktop anti-virus protection has a disadvantage of being widely available to the hacker community, which is always eager to find vulnerabilities. In addition, desktop anti-virus protection has a disadvantage of requiring installation of client software.

[0023] As such, there is a need for a new form of behavioral analysis, which can shield computers from dynamically generated malicious code without running on the computer itself that is being shielded.

#### SUMMARY OF THE DESCRIPTION

[0024] The present invention concerns systems and methods for implementing new behavioral analysis technology. The new behavioral analysis technology affords protection against dynamically generated malicious code, in addition to conventional computer viruses that are statically generated.

[0025] The present invention operates through a security computer that is preferably remote from a client computer that is being shielded while processing network content.

During run-time, while processing the network content, but before the client computer invokes a function call that may potentially dynamically generate malicious code, the client computer passes the input to the function to the security computer for inspection, and suspends processing the network content pending a reply back from the security computer. Since the input to the function is being passed at run-time, it has already been dynamically generated and is thus readily inspected by a content inspector. Referring to the example above, were the input to be passed to the security computer prior to run-time, it would take the form of indeterminate text; whereas the input passed during run-time takes the determinate form

<SCRIPT>malicious JavaScript</SCRIPT>,

which can readily be inspected. Upon receipt of a reply from the security computer, the client computer resumes processing the network content, and knows whether to by-pass the function call invocation.

[0026] To enable the client computer to pass function inputs to the security computer and suspend processing of content pending replies from the security computer, the present invention operates by replacing original function calls with substitute function calls within the content, at a gateway computer, prior to the content being received at the client computer.

[0027] The present invention also provides protection against arbitrarily many recursive levels of dynamic generation of malicious code, whereby such code is generated via a series of successive function calls, one within the next.

[0028] By operating through the medium of a security computer, the present invention overcomes the disadvantages of desktop anti-virus applications, which are available to the hacker community for exploit. Security applications embodying the present invention are concealed securely within managed computers.

[0029] There is thus provided in accordance with a preferred embodiment of the present invention a method for protecting a client computer from dynamically generated malicious content, including receiving at a gateway computer content being sent to a client computer



for processing, the content including a call to an original function, and the call including an input, modifying the content at the gateway computer, including replacing the call to the original function with a corresponding call to a substitute function, the substitute function being operational to send the input to a security computer for inspection, transmitting the modified content from the gateway computer to the client computer, processing the modified content at the client computer, transmitting the input to the security computer for inspection when the substitute function is invoked, determining at the security computer whether it is safe for the client computer to invoke the original function with the input, transmitting an indicator of whether it is safe for the client computer to invoke the original function with the input, from the security computer to the client computer, and invoking the original function at the client computer with the input, only if the indicator received from the security computer indicates that such invocation is safe.

[0030] There is further provided in accordance with a preferred embodiment of the present invention a system for protecting a client computer from dynamically generated malicious content, including a gateway computer, including a gateway receiver for receiving content being sent to a client computer for processing, the content including a call to an original function, and the call including an input, a content modifier for modifying the received content by replacing the call to the original function with a corresponding call to a substitute function, the substitute function being operational to send the input to a security computer for inspection, and a gateway transmitter for transmitting the modified content from the gateway computer to the client computer, a security computer, including a security receiver for receiving the input from the client computer, an input inspector for determining whether it is safe for the client computer to invoke the original function with the input, and a security transmitter for transmitting an indicator of the determining to the client computer, and a client computer communicating with the gateway computer and with the security computer, including a client receiver for receiving the modified content from the gateway computer, and for receiving the indicator from the security computer, a content processor for processing the modified content, and for invoking the original function only if the indicator indicates

that such invocation is safe; and a client transmitter for transmitting the input to the security computer for inspection, when the substitute function is invoked.

[0031] There is yet further provided in accordance with a preferred embodiment of the present invention a computer-readable storage medium storing program code for causing at least one computing device to receive content including a call to an original function, and the call including an input, replace the call to the original function with a corresponding call to a substitute function, the substitute function being operational to send the input for inspection, thereby generating modified content, process the modified content, transmit the input for inspection, when the substitute function is invoked while processing the modified content, and suspend processing of the modified content, determine whether it is safe to invoke the original function with the input, transmit an indicator of whether it is safe for a computer to invoke the original function with the input, and resume processing of the modified content after receiving the indicator, and invoke the original function with the input only if the indicator indicates that such invocation is safe.

[0032] There is additionally provided in accordance with a preferred embodiment of the present invention a method for protecting a client computer from dynamically generated malicious content, including receiving content being sent to a client computer for processing, the content including a call to an original function, and the call including an input, modifying the content, including replacing the call to the original function with a corresponding call to a substitute function, the substitute function being operational to send the input to a security computer for inspection, and transmitting the modified content to the client computer for processing.

[0033] There is moreover provided in accordance with a preferred embodiment of the present invention a system for protecting a client computer from dynamically generated malicious content, including a receiver for receiving content being sent to a client computer for processing, the content including a call to an original function, and the call including an input, a content modifier for modifying the received content by replacing the call to the original function with a corresponding call to a substitute function, the substitute function

being operational to send the input to a security computer for inspection, and a transmitter for transmitting the modified content to the client computer.

[0034] There is further provided in accordance with a preferred embodiment of the present invention a computer-readable storage medium storing program code for causing a computing device to receive content including a call to an original function, and the call including an input, and replace the call to the original function with a corresponding call to a substitute function, the substitute function being operational to send the input for inspection.

[0035] There is yet further provided in accordance with a preferred embodiment of the present invention a method for protecting a client computer from dynamically generated malicious content, including receiving content being sent to a client computer for processing, the content including a call to an original function, and the call including an input, modifying the content, including replacing the call to the original function with a corresponding call to a substitute function, the substitute function being operational to send the input for inspection, transmitting the modified content to the client computer for processing, receiving the input from the client computer, determining whether it is safe for the client computer to invoke the original function with the input, and transmitting to the client computer an indicator of whether it is safe for the client computer to invoke the original function with the input.

[0036] There is additionally provided in accordance with a preferred embodiment of the present invention a system for protecting a client computer from dynamically generated malicious content, including a receiver (i) for receiving content being sent to a client computer for processing, the content including a call to an original function, and the call including an input, and (ii) for receiving the input from the client computer, a content modifier for modifying the received content by replacing the call to the original function with a corresponding call to a substitute function, the substitute function being operational to send the input for inspection, an input inspector for determining whether it is safe for the client computer to invoke the original function with the input, and a transmitter (i) for transmitting the modified content to the client computer, and (ii) for transmitting an indicator of the determining to the client computer.

[0037] There is moreover provided in accordance with a preferred embodiment of the present invention a computer-readable storage medium storing program code for causing a computing device to receive content including a call to an original function, and the call including an input, replace the call to the original function with a corresponding call to a substitute function, the substitute function being operational to send the input for inspection, and determine whether it is safe for a computer to invoke the original function with the input.

[0038] There is further provided in accordance with a preferred embodiment of the present invention a method for protecting a computer from dynamically generated malicious content, including processing content received over a network, the content including a call to a first function, and the call including an input, transmitting the input to a security computer for inspection, when the first function is invoked, receiving from the security computer an indicator of whether it is safe to invoke a second function with the input, and invoking the second function with the input, only if the indicator indicates that such invocation is safe.

[0039] There is yet further provided in accordance with a preferred embodiment of the present invention a system for protecting a computer from dynamically generated malicious content, including a content processor (i) for processing content received over a network, the content including a call to a first function, and the call including an input, and (ii) for invoking a second function with the input, only if a security computer indicates that such invocation is safe, a transmitter for transmitting the input to the security computer for inspection, when the first function is invoked, and a receiver for receiving an indicator from the security computer whether it is safe to invoke the second function with the input.

[0040] There is additionally provided in accordance with a preferred embodiment of the present invention a computer-readable storage medium storing program code for causing a computing device to process content received over a network, the content including a call to a first function, and the call including an input, transmit the input for inspection, when the first function is invoked, and suspend processing of the content, receive an indicator of whether it is safe to invoke a second function with the input, and resume processing of the

content after receiving the indicator, and invoke the second function with the input only if the indicator indicates that such invocation is safe.

[0041] There is moreover provided in accordance with a preferred embodiment of the present invention a method for protecting a client computer from dynamically generated malicious content, including receiving an input from a client computer, determining whether it is safe for the client computer to invoke a function with the input, and transmitting an indicator of the determining to the client computer.

[0042] There is further provided in accordance with a preferred embodiment of the present invention a system for protecting a client computer from dynamically generated malicious content, including a receiver for receiving an input from a client computer, an input inspector for determining whether it is safe for the client computer to invoke a function with the input, and a transmitter for transmitting an indicator of the determining to the client computer.

[0043] There is further provided in accordance with a preferred embodiment of the present invention a computer-readable storage medium storing program code for causing a computing device to receive an input from a computer, determine whether it is safe for the computer to invoke a function with the input, and transmit an indicator of the determination to the computer.

[0044] The following definitions are employed throughout the specification and claims.  
SECURITY POLICY - a set of one or more rules that determine whether or not a requested operation is permitted. A security policy may be explicitly configurable by a computer system administrator, or may be implicitly determined by application defaults.  
SECURITY PROFILE - information describing one or more suspicious operations performed by executable software.

#### BRIEF DESCRIPTION OF THE DRAWINGS

[0045] The present invention will be more fully understood and appreciated from the following detailed description, taken in conjunction with the drawings in which:

[0046] **FIG. 1** is a simplified block diagram of prior art systems for blocking malicious content;

[0047] **FIG. 2** is a simplified block diagram of a system for protecting a computer from dynamically generated malicious executable code, in accordance with a preferred embodiment of the present invention;

[0048] **FIG. 3** is a simplified flowchart of a method for protecting a computer from dynamically generated malicious executable code, in accordance with a preferred embodiment of the present invention;

[0049] **FIG. 4** is a simplified block diagram of a system for protecting a computer from dynamically generated malicious executable code, in which the gateway computer itself performs the code inspection, in accordance with a preferred embodiment of the present invention; and

[0050] **FIG. 5** is a simplified flowchart of a method for protecting a computer from dynamically generated malicious executable code, whereby the gateway computer itself performs the code inspection, in accordance with a preferred embodiment of the present invention.

## DETAILED DESCRIPTION

[0051] The present invention concerns systems and methods for protecting computers against dynamically generated malicious code.

[0052] Reference is now made to **FIG. 2**, which is a simplified block diagram of a system for protecting a computer from dynamically generated malicious executable code, in accordance with a preferred embodiment of the present invention. Three major components of the system are a gateway computer **205**, a client computer **210**, and a security computer **215**. Gateway computer **220** receives content from a network, such as the Internet, over a communication channel **220**. Such content may be in the form of HTML pages, XML documents, Java applets and other such web content that is generally rendered by a web

browser. Client computer **210** communicates with gateway computer **205** over a communication channel **225**, and communicates with security computer **215** over a communication channel **230**. Gateway computer **205** receives data at gateway receiver **235**, and transmits data at gateway transmitter **240**. Similarly, client computer **210** receives data at client receiver **245**, and transmits data at client transmitter **250**; and security computer **215** receives data at security receiver **260** and transmits data at security transmitter **265**.

[0053] It will be appreciated by those skilled in the art that the network topology of **FIG. 2** is shown as a simple topology, for purposes of clarity of exposition. However, the present invention applies to general architectures including a plurality of client computers **210** that are serviced by one or more gateway computers **205**, and by one or more security computers **215**. Similarly, communication channels **220**, **225** and **230** may each be multiple channels using standard communication protocols such as TCP/IP.

[0054] Moreover, the functionality of security computer **215** may be included within gateway computer **205**. Such a topology is illustrated in **FIG. 4**.

[0055] The computers shown in **FIG. 2** also include additional processing modules, each of which is described in detail hereinbelow. Gateway computer **205** includes a content modifier **265**, client computer **210** includes a content processor **270**, and security computer **215** includes an inspector **275**, a database of client security policies **280**, and an input modifier **285**.

[0056] Content modifier **265** preferably modifies original content received by gateway computer **205**, and produces modified content, which includes a layer of protection to combat dynamically generated malicious code. Specifically, content modifier **265** scans the original content and identifies function calls of the form

Function (input), (1)

Content modifier **265** further modifies selected ones of the function calls (1) to corresponding function calls

Substitute\_function (input, \*), (2)

whereby the call to *Function()* has been replaced with a call to *Substitute\_function()*. It is noted that the input intended for the original function is also passed to the substitute function, along with possible additional input denoted by "\*".

[0057] It will be appreciated by those skilled in the art that content modifier **265** may modify all detected function calls, or only a portion of the detected function calls. Functions that are known to be safe, regardless of their inputs, need not be modified by content modifier **265**. Similarly, functions that are not passed any inputs when invoked and are known to be safe, also need not be modified by content modifier **265**.

[0058] Preferably, when call **(2)** is made, the substitute function sends the input to security computer **215** for inspection. Preferably, content modifier **265** also inserts program code for the substitute function into the content, or a link to the substitute function. Such a substitute function may be of the following general form shown in **TABLE I**.

---

**TABLE I:** Generic substitute function

---

```
Function Substitute_function(input)
{
    inspection_result = Call_security_computer_to_inspect (
                        input, ID_of_client_computer);
    if (inspection_result)
        Original_function(input)
    else
        //do nothing
}
```

---



Preferably, the above function *call\_security\_computer\_to\_inspect()* passes the input intended for the original function to security computer **215** for inspection by inspector **275**. In addition, an ID of client computer **210** is also passed to security computer **215**. When security computer services many such client computers **210** at once, it uses such IDs to determine where to return its results. For example, the ID may correspond to a network address of client computer **210**. When security computer **215** services many such client computers **210** at once, it uses the IDs to determine where to return each of its many results.

[0059] Optionally, the substitute function may pass additional parameters to security computer **215**, such as the name of the original function, or security policy information as described hereinbelow with reference to database **280**.

[0060] The function *call\_security\_computer\_to\_inspect()* preferably returns an indicator, *inspection\_result*, of whether it is safe for client computer **210** to invoke the original function call **(1)**. The indicator may be a Boolean variable, or a variable with more than two settings that can carry additional safety inspection information. In addition, as described hereinbelow with reference to input modifier **285**, the function *call\_security\_computer\_to\_inspect()* may modify the input, and return to client computer **210** modified input to be used when invoking the original function call **(1)**, instead of the original input. Use of input modifier **285** protects client computer **210** against recursively generated malicious code whereby the input itself to a first function generates a call to a second function.

[0061] For example, suppose a portion of the original content is of the form shown in TABLE II.

---

TABLE II: Example original content

---

```
<!DOCTYPE HTML PUBLIC "-//w3c//DTD HTML 4.0 Transitional//EN">
<HTML>
<SCRIPT LANGUAGE="JavaScript"
<!
Document.write("<hl>hello</hl>");

</SCRIPT>
<BODY>

</BODY>
</HTML>
```

---

Preferably, content modifier **265** alters the original content in TABLE II to the modified form shown in TABLE III. Specifically, content modifier **265** substitutes the call to the standard function *Document.write()*, with a call to the substitute function *Substitute\_document.write()*, and inserts the function definition for the substitute function into the content. The standard function *Document.write()* generally writes lines of HTML and inserts them into the HTML page currently being processed by a client web browser.

---

**Table III:** Example modified content
 

---

```

<!DOCTYPE HTML PUBLIC "-//w3c//DTD HTML 4.0 Transitional//EN"u>
<HTML>
<SCRIPT LANGUAGE="JavaScript"
<!--
Function Substitute_document.write(text)
{
    inspection_result = Call_security_computer_to_inspect(text);
    if inspection_result
        Document.write(text)
    Else
        //do nothing
}
Substitute_document.write("<hl>hello</hl>");
</SCRIPT>
<BODY>

</BODY>

</HTML>

```

---

[0062] Content processor **270** processes the modified content generated by content modifier **265**. Content processor may be a web browser running on client computer **210**. When content processor invokes the substitute function call **(2)**, the input is passed to security computer **215** for inspection. Processing of the modified content is then suspended until security computer **215** returns its inspection results to client computer **210**. Upon receiving the inspection results, client computer **210** resumes processing the modified content. If `inspection_result` is true, then client computer **210** invokes the original function call **(1)**; otherwise, the client computer **210** does not invoke the original function call **(1)**.

[0063] Security computer **215** may also modify the input that is passed to it by the substitute function. In such case, client computer **210** invokes the original function with such modified input, instead of the original input, after receiving the inspection results.

[0064] Input inspector **275** analyzes the input passed to security computer **215** by client computer **210**; specifically, the input passed when client computer **210** invokes the function call **(2)**. Generally, input inspector **275** scans the input to determine the potentially malicious operations that it may perform, referred to as the input's "security profile". Such potentially malicious operations can include inter alia operating system level commands, file system level commands, network level commands, application level commands, certain URLs with hyperlinks, and applets already known to be malicious. Security profiles are described in assignee's US Patent No. 6,092,194 entitled SYSTEM AND METHOD FOR PROTECTING A COMPUTER AND A NETWORK FROM HOSTILE DOWNLOADABLES, the contents of which are hereby incorporated by reference. Security profiles encompass access control lists, trusted/un-trusted certificates, trusted/un-trusted URLs, and trusted/un-trusted content.

[0065] After determining a security profile for the input, inspector **275** preferably retrieves information about permission settings for client computer **210**, referred to as client computer's "security policy". Such permission settings are generally set by an administrator of client computer **210**, and determine which commands are permitted to be performed by content processor **270** while processing content, and which commands are not permitted. Security policies are also described in assignee's US Patent No. 6,092,194. Security policies are flexible, and are generally set by an administrator of client computer **210**. Preferably, security computer **215** has accesses to a database **280** of security profile information for a plurality of client computers. Database **280** may reside on security computer **215**, or on a different computer.

[0066] By comparing the input's security policy to client computer **210**'s security profile, input inspector **275** determines whether it is safe for client computer **210** to make the function call **(1)**. Security computer **215** sends back to client computer **210** an indicator, *inspection\_result*, of the inspector's determination. Comparison of a security policy to a

security profile is also described in assignee's US Patent No. 6,092,194. Security policies may include simple or complex logical tests for making a determination of whether or not an input is safe.

[0067] For example, suppose the content is an HTML page, and the function call (1) is the following JavaScript:

```
Document.write("<hl><SCRIPT>Some JavaScript</SCRIPT></hl>") (3)
```

Such a function call serves to instruct content processor 270 to insert the text between the <h1> header tags into the HTML pages; namely the text <SCRIPT>JavaScript</SCRIPT> which itself invokes the JavaScript between the <SCRIPT> tags. It is noted that the function call (1) uses a function *Document.write()* that is normally considered to be safe. Indeed, the function *Document.write()* does not access client computer 210's operating system or file system and does not send or receive data outside of client computer 210. Moreover, the input in the call (3) to *Document.write()* may itself be dynamically generated, and not available for inspection prior to processing the HTML page. That is, the call may be of the form

```
Document.write("content that is dynamically generated at run-time"),
```

where input to *Document.write()* may be in the form of a text string that itself is dynamically generated at run-time. Generally, such a function call cannot be analyzed successfully by behavioral based anti-virus software prior to run-time.

[0068] However, when input inspector 275 receives the input from client computer during run-time, after client computer has invoked the substitute call (2), the input has already been dynamically generated by content processor 270 and can thus be readily analyzed. Referring to the example above, when client computer 210 invokes the substitute call (2), it passes the input string

```
"<hl><SCRIPT>JavaScript</SCRIPT></hl>" (4)
```

to security computer 215. This string is then analyzed by input inspector 275, which recognizes the JavaScript and scans the JavaScript to determine any potentially malicious operations it includes. If potentially malicious operations are detected, and if they violate client computer 210's security policy, then inspector 275 preferably sets `inspection_result` to false. Otherwise, inspector 275 preferably sets `inspection_result` to true.

[0069] It may thus be appreciated by those skilled in the art that input inspector 275 is able to detect malicious code that is generated at runtime.

[0070] Malicious code may be generated within further recursive levels of function calls. For example, instead of the function call (3), which invokes a single function to dynamically generate JavaScript, two levels of function calls may be used. Consider, for example, the recursive function

call

```
Document.write("<hl>Docurnent.write(
    "<hl><SCRIPT>Some JavaScript</SCRIPT></hl>" </hl>") (5)
```

Such a function call first calls *Document.write()* to generate the function call (3), and then calls *Document.write()* again to generate the JavaScript. If the inputs to each of the *Document.write()* invocations in (5) are themselves dynamically generated at run-time, then one pass through input inspector may not detect the JavaScript.

[0071] To this end, input inspector 275 preferably passes inputs it receives to input modifier 285, prior to scanning the input. Input modifier preferably operates similar to content modifier 265, and replaces function calls detected in the input with corresponding substitute function calls. Referring to the example above, when client computer 210 invokes the outer call to *Document.write()* in (5), the input ext string

```
"<hl>Document.write(
    "<hl><SCRIPT>Some JavaScript</SCRIPT></hl>"</hl>" (6)
```

is passed to security computer 215. Input modifier 285 detects the inner function call to *Document.write()* and replaces it with a corresponding substitute function call of the form (2). Input inspector 275 then inspects the modified input. At this stage, if the input to the inner call to *Document.write()* has not yet been dynamically generated, input inspector may not detect the presence of the JavaScript, and thus may not set *inspection\_result* to false if the JavaScript is malicious. However, security computer 215 returns the modified input to client computer 210. As such, when content processor 270 resumes processing, it adds the modified input into the HTML page. This guarantees that when content processor 270 begins to process the modified input, it will again invoke the substitute function for *Document.write()*, which in turn passes the input of the inner *Document.write()* call of (5) to security computer 215 for inspection. This time around input inspector 275 is able to detect the presence of the JavaScript, and can analyze it accordingly.

[0072] It may thus be appreciated by those skilled in the art that when input modifier 285 supplements input inspector 275, inspector 275 has sufficient logic to be able to detect malicious code that is generated recursively at run-time.

[0073] In addition to inspecting inputs, security computer 215 preferably maintains an event log of potential security breaches. When input inspector 275 determines that an input is not safe, security computer 215 enters information about the input and client computer 210 into a log that is available for review by an administrator of client computer 210.

[0074] In accordance with a preferred embodiment of the present invention, it is anticipated that many client computers 210 use the same security computer 215 for protection. Each client computer may independently send inputs to security computer 215 for inspection. Security computer 215 may use cache memory to save results of inspection, so as to obviate the need to analyze the same input more than once. Use of cache memory when working with a plurality of security policies is described in assignee's US Patent No. 6,965,968 entitled POLICY-BASED CACHING.

[0075] Similarly, it is anticipated that gateway computer 205 services many client computers 210. Gateway computer may include its own content inspector, which is useful

for detecting malicious content that is not dynamically generated, as described in assignee's US Patent No. 6,092,194.

[0076] It may be appreciated that substitute functions as in TABLE I may also pass the name of the original function to the security computer. That is, the call to *Call\_security\_computer\_to\_inspect()* may also a variable, say *name\_of\_function*, so that input inspector **275** can determine whether it is safe to invoke the specific original function with the input. In this way, input inspector **275** can distinguish between different functions with the same input.

[0077] Reference is now made to **FIG. 3**, which is a simplified flowchart of a method for protecting a computer from dynamically generated malicious executable code, in accordance with a preferred embodiment of the present invention. The leftmost column of **FIG. 3** shows steps performed by a gateway computer, such as gateway computer **205**. The middle column of **FIG. 3** shows steps performed by a client computer, such as client computer **210**. The rightmost column of **FIG. 3** shows steps performed by a security computer, such as security computer **215**.

[0078] At step **304**, the gateway computer receives content from a network, the content on its way for delivery to the client computer. Such content may be in the form of an HTML web page, an XML document, a Java applet, an EXE file, JavaScript, VBScript, an ActiveX Control, or any such data container that can be rendered by a client web browser. At step **308**, the gateway computer scans the content it received, for the presence of function calls. At step **312**, the gateway computer branches, depending on whether or not function calls were detected at step **308**. If function calls were detected, then at step **318** the gateway computer replaces original function calls with substitute function calls within the content, thereby modifying the content. If function calls were not detected, then the gateway computer skips step **318**. At step **320**, the gateway computer sends the content, which may have been modified at step **318**, to the client computer.

[0079] At step **324** the client computer receives the content, as modified by the gateway computer. At step **328** the client computer begins to continuously process the modified



content; i.e., the client computer runs an application, such as a web browser or a Java virtual machine, that processes the modified content. At step **332**, which processing the modified content, the client computer encounters a call **(2)** to a substitute function, such as the substitute function listed in TABLE I. Client computer then transmits the input to the substitute function and an identity of the client computer, to the security computer for inspection, at step **336**. The identity of the client computer serves to inform the security computer where to return its inspection result. Since one security computer typically services many client computers, passing client computer identities is a way to direct the security computer where to send back its results. At this point, client computer suspends processing the modified content pending receipt of the inspection results from the security computer. As mentioned hereinabove, the client computer may also send the name of the original function to the security computer, for consideration in the inspection analysis.

[0080] At step **340** the security computer receives the input and client computer identifier. At step **344** the security computer scans the input for the presence of function calls. At step **348** the security computer branches, depending on whether or not function calls were detected at step **344**. If function calls were detected, then the security computer replaces original function calls with substitute function calls at step **352**, thereby modifying the input. The security computer may insert definitions of the substitute functions into the input, as indicated in TABLE III, or may insert links to such definitions. Otherwise, the security computer skips step **352**. Steps **344**, **348** and **352** are similar to respective steps **308**, **312** and **316** performed by the gateway computer.

[0081] At step **356** the security computer scans the input, which may have been modified at step **352**, for the presence of potentially malicious operations. Preferably, the security computer determines a security profile for the input, which corresponds to a list of the potentially malicious operations that are detected.

[0082] At step **360** the security computer retrieves a security policy that governs the client computer. The security policy may be retrieved from a database that stores a plurality of security policies, each policy configurable by an administrator of client computers. Security

policies may be set at a fine granularity of a policy for each client computer, or at a coarser granularity of a policy that applies to an entire department or workgroup.

[0083] At step **364** the security computer compares the security profile of the input under inspection with the security profile of the client computer, to determine if it is permissible for the client computer to invoke an original function with the input. Such determination may involve one or more simple or complex logical tests, structured in series or in parallel, or both, as described in assignee's US Patent No. 6,092,194.

[0084] At step **368** the security computer branches depending on the result of the comparison step **364**. If the comparison step determines that the input is safe; i.e., that the input's security profile does not violate the client computer's security policy, then at step **372** the security computer sets an indicator of inspection results to true. Otherwise, at step **376** the security computer sets the indicator to false. At step **380** the security computer returns the indicator to the client computer. In addition, if the security computer modified the input as step **352**, then it also returns the modified input to the client computer.

[0085] At step **384** the client computer receives the indicator and the modified input from the security computer and resumes processing the modified content, which had been suspended after step **336** as described hereinabove. At step **388** the client computer branches depending on the value of the indicator it received from the security computer. If the indicator is true, indicating that it is safe for the client computer to invoke the original function call **(1)**, then the client computer invokes the original function using the modified input it received from the security computer, at step **392**. Otherwise, the client computer does not invoke the original function, since the indicator indicates that such invocation may be malicious to the client computer. The client computer then loops back to step **328** to continue processing the modified content.

[0086] As described hereinabove, steps **344**, **348** and **352**, which modify the input, are useful in protecting against malicious code that is dynamically generated in a recursive manner, as in function call **(5)**. The security computer may require multiple passes to detect such malicious code, and steps **344**, **348** and **352** provide the mechanism for this to happen.

[0087] Reference is now made to **FIG. 4**, which is a simplified block diagram of a system for protecting a computer from dynamically generated malicious executable code, in which the gateway computer itself performs the code inspection, in accordance with a preferred embodiment of the present invention. The system illustrated in **FIG. 4** is similar to the system of **FIG. 2**, where the functionality of the security computer has been incorporated into the gateway computer. The elements in **FIG. 4** are thus similar in functionality to the elements in **FIG. 2**.

[0088] Two major components of the system, gateway computer **405** and client computer **410** communicate back and forth over communication channel **425**. Gateway computer **405** includes a gateway receiver **435** and a gateway transmitter **440**; and client computer **410** includes a client receiver **445** and a client transmitter **450**. Although **FIG. 4** includes only one client computer, this is solely for the purpose of clarity of exposition, and it is anticipated that gateway computer **405** serves many client computers **410**.

[0089] Gateway computer **405** receives content, such as web content, from a network, over communications channel **420**. Client computer **410** includes a content processor **470**, such as a web browser, which processes content received from the network.

[0090] In accordance with a preferred embodiment of the present invention, gateway computer **405** includes an input inspector **475**, and a content modifier **465** which also serves as an input modifier. That is, content modifier **465** incorporates the functionalities of content modifier **265** and input modifier **285** from **FIG. 2**. In addition, gateway computer **405** includes a database **480** of security policies, or else has access to such a database. The operations of input inspector **475** and content/input modifier **465** are similar to the operations of the corresponding elements in **FIG. 2**, as described hereinabove.

[0091] Incoming content received at gateway computer **405** passes through content modifier **465**, which replaces function calls of the form **(1)** with substitute function calls of the form **(2)**, and the modified content is transmitted to client computer **410**. Content processor **470** processes the modified content and, while processing the modified content, if it encounters a substitute function call it sends the function's input to inspector **475** for inspection, and suspends processing of the modified content. The input passes through input modifier **465**, and input inspector **475** analyzes the modified input for the presence of potentially malicious operations. Gateway computer **405** returns the input inspection results to client computer **410**. Gateway computer **405** may also return the modified input to client computer **410**. After receiving the inspection results, client computer **410** resumes processing the modified content and invokes or does not invoke the original function call, based on the inspection results.

Reference is now made to **FIG. 5**, which is a simplified flowchart of a method for protecting a computer from dynamically generated malicious executable code, whereby the gateway computer itself performs the code inspection, in accordance with a preferred embodiment of the present invention. The leftmost column indicates steps performed by a gateway computer, such as gateway computer **405**; and the rightmost column indicates steps performed by a client computer, such as client computer **410**.

[0092] The method illustrated in **FIG. 5** is similar to that of **FIG. 3**, where steps **340 - 380** performed by the security computer in **FIG. 3** are performed by the gateway computer in **FIG. 5**. At step **500** the gateway computer receives content from a network, the content intended for delivery to the client computer. At step **505** the gateway computer scans the content for the presence of function calls. At step **510** the gateway computer branches. If function calls within the content were detected at step **505**, then at step **515** the gateway computer modifies the content by replacing original function calls of the form **(1)** with corresponding substitute function calls of the form **(2)**. Otherwise, if function calls were not detected at step **505**, then the gateway computer skips step **515**. At step **520** the gateway computer transmits the content, which may have been modified at step **515**, to the client computer.

[0093] At step **525** the client computer receives the content from the gateway computer, and at step **530** the client computer begins processing the content. While processing the content, the client computer invokes a substitute function call of the form **(2)** at step **535**. The substitute function, being of the form listed on TABLE I, instructs the client computer to transmit the function input and a client computer identifier to the gateway computer for inspection. At step **540** the client computer transmits the input and the identifier to the gateway computer, and suspends processing of the content pending a reply from the gateway computer.

[0094] At step **545** the gateway computer receives the input and the client identifier from the client computer, and loops back to step **505** to scan the input for the presence of function calls. At step **510** the gateway computer branches. If function calls within the Input were detected at step **505**, then the gateway computer modifies the input at step **515**, by replacing function calls of the form **(1)** with corresponding function calls of the form **(2)**. Otherwise, if function calls were not detected at step **505**, then the gateway computer skips step **515**.

[0095] The gateway computer then proceeds to step **550**, and scans the input, which may have been modified at step **515**, to identify potentially malicious operations within the input. The potentially malicious operations identified form a security profile for the input.

[0096] At step **555** the gateway computer retrieves a security policy for the client computer from a database of security policies. At step **560** the gateway computer compares the input's security profile with the client computer's security policy to determine whether or not the security profile violates the security policy. At step **565** the gateway computer branches. If the results of step **560** indicate that the input security profile does not violate the client computer security policy, then it is safe for the client to invoke the original function call, and an indicator of the inspection results is set to true at step **570**. Otherwise, the indicator is set to false at step **575**. At step **580** the gateway computer returns the indicator to the client computer. The gateway computer may also return the modified input, as modified at step **515**, to the client computer.

[0097] At step **585** the client computer receives the reply back from the gateway computer and resumes processing of the content, which processing had been suspended after step **540**. At step **590** the client computer branches. If the indicator was set to true by the gateway computer at step **570**, then the client computer invokes the original function call **(1)**. If the gateway computer had modified the input at step **515**, then preferably the client computer uses the modified input instead of the original input when invoking the original function call. Otherwise, if the indicator was set to false by the gateway computer at step **575**, then the client computer skips step **595**. The client computer then loops back to step **530** to continue processing of the content.

[0098] Having read the above disclosure, it will be appreciated by those skilled in the art that the present invention can be used to provide protection to computers against both statically and dynamically generated malicious code. Moreover, such protection may be afforded by a security computer that is remote from the computers being protected, thus adding another layer of security to methods and systems that embody the present invention.

[0099] In reading the above description, persons skilled in the art will realize that there are many apparent variations that can be applied to the methods and systems described. Thus it may be appreciated that the present invention applies to a variety of computing devices, including mobile devices with wireless Internet connections such as laptops, PDAs and cell phones.

[00100] In the foregoing specification, the invention has been described with reference to specific exemplary embodiments thereof. It will, however, be evident that various modifications and changes may be made to the specific exemplary embodiments without departing from the broader spirit and scope of the invention as set forth in the appended claims. Accordingly, the specification and drawings are to be regarded in an illustrative rather than a restrictive sense.

## CLAIMS

What is claimed is:

1. A system for protecting a computer from dynamically generated malicious content, comprising:

a content processor (i) for processing content received over a network, the content including a call to a first function, and the call including an input, and (ii) for invoking a second function with the input, only if a security computer indicates that such invocation is safe;

a transmitter for transmitting the input to the security computer for inspection, when the first function is invoked; and

a receiver for receiving an indicator from the security computer whether it is safe to invoke the second function with the input.

2. The system of claim 1 wherein said content processor (i) suspends processing of the content after said transmitter transmits the input to the security computer, and (ii) resumes processing of the modified content after said receiver receives the indicator from the security computer.

3. A computer-readable storage medium storing program code for causing a computing device to: process content received over a network, the content including a call to a first function, and the call including an input;

transmit the input for inspection, when the first function is invoked, and suspend processing of the content;

receive an indicator of whether it is safe to invoke a second function with the input; and

resume processing of the content after receiving the indicator, and invoke the second function with the input only if the indicator indicates that such invocation is safe.

## ABSTRACT OF THE DISCLOSURE

A method for protecting a client computer from dynamically generated malicious content, including receiving at a gateway computer content being sent to a client computer for processing, the content including a call to an original function, and the call including an input, modifying the content at the gateway computer, including replacing the call to the original function with a corresponding call to a substitute function, the substitute function being operational to send the input to a security computer for inspection, transmitting the modified content from the gateway computer to the client computer, processing the modified content at the client computer, transmitting the input to the security computer for inspection when the substitute function is invoked, determining at the security computer whether it is safe for the client computer to invoke the original function with the input, transmitting an indicator of whether it is safe for the client computer to invoke the original function with the input, from the security computer to the client computer, and invoking the original function at the client computer with the input, only if the indicator received from the security computer indicates that such invocation is safe. A system and a computer-readable storage medium are also described and claimed.



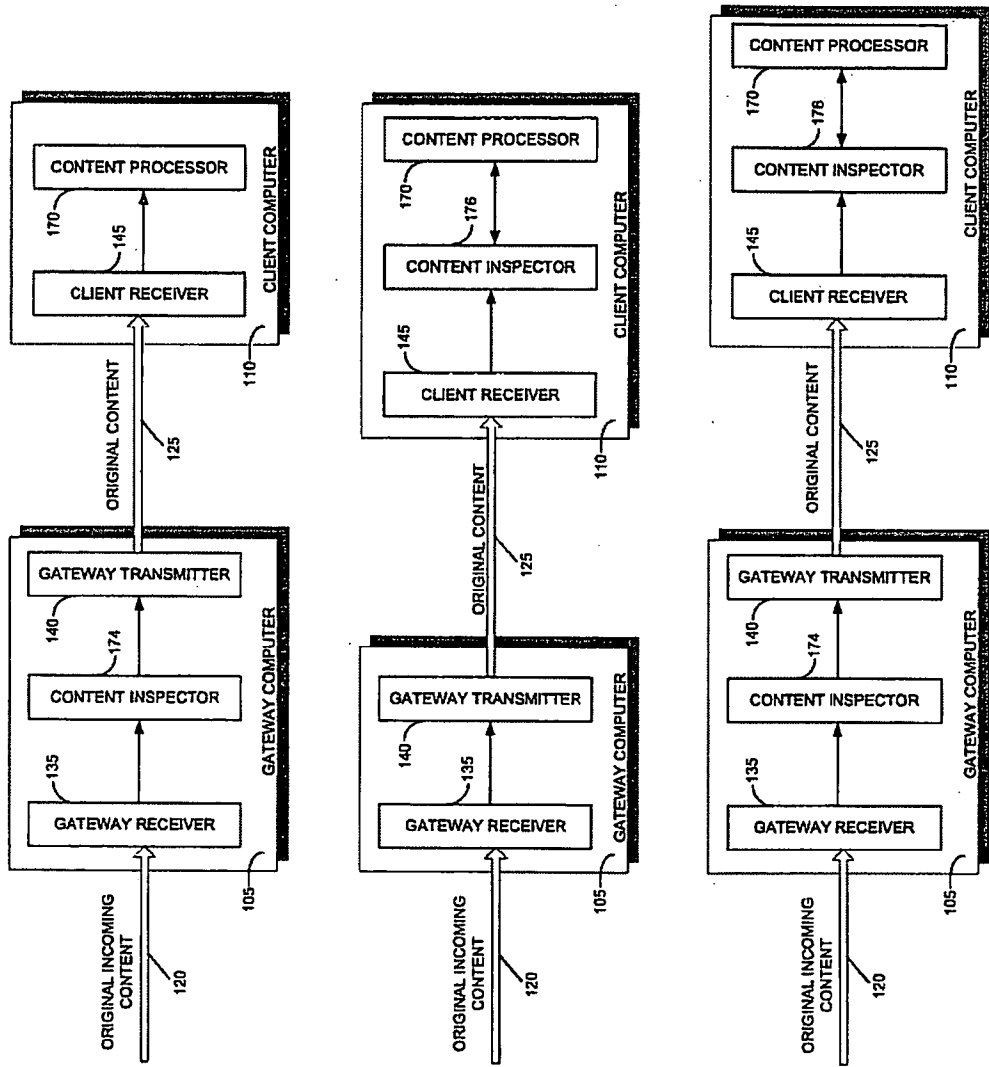


FIG. 1  
(PRIOR ART)

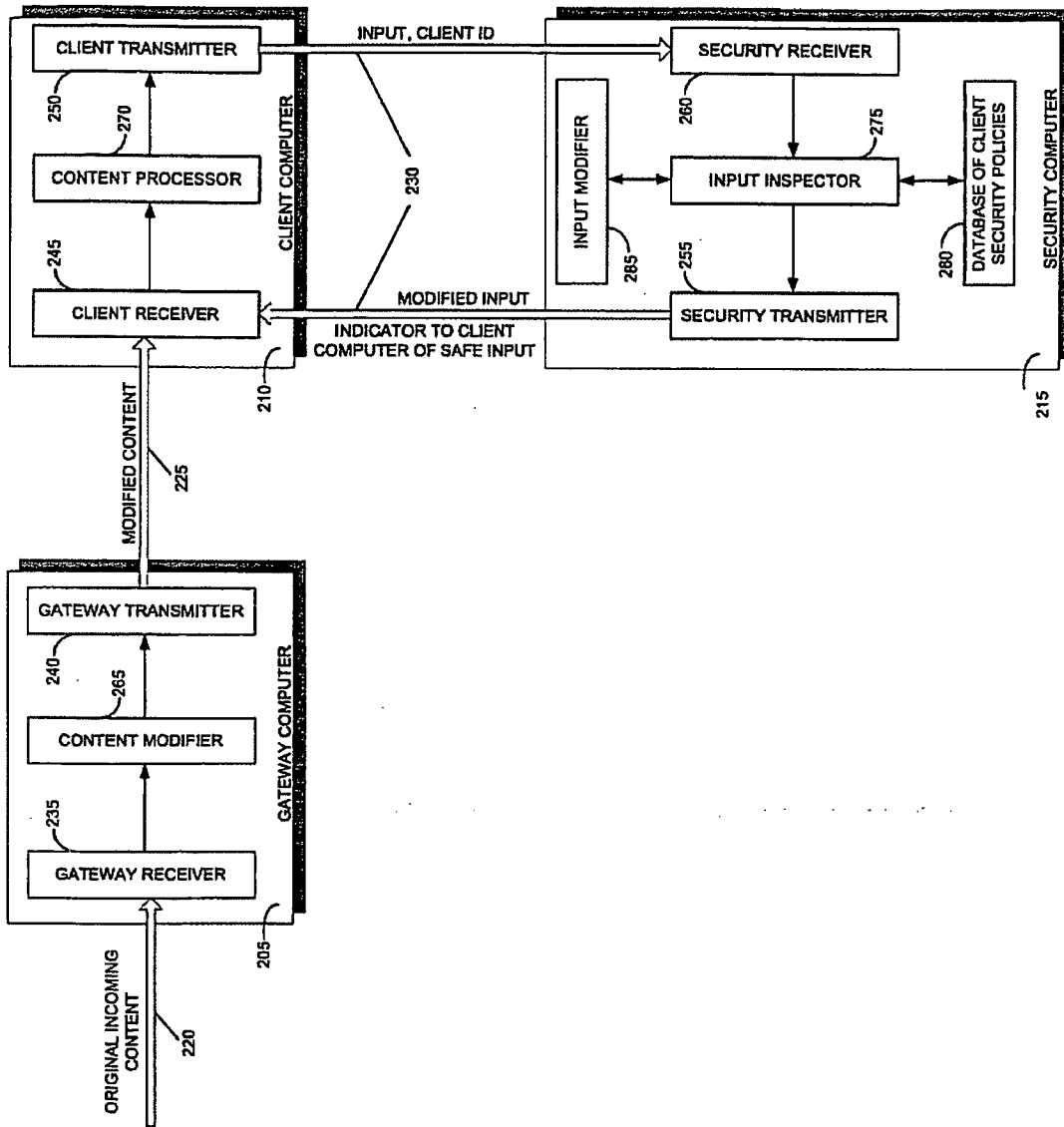


FIG. 2

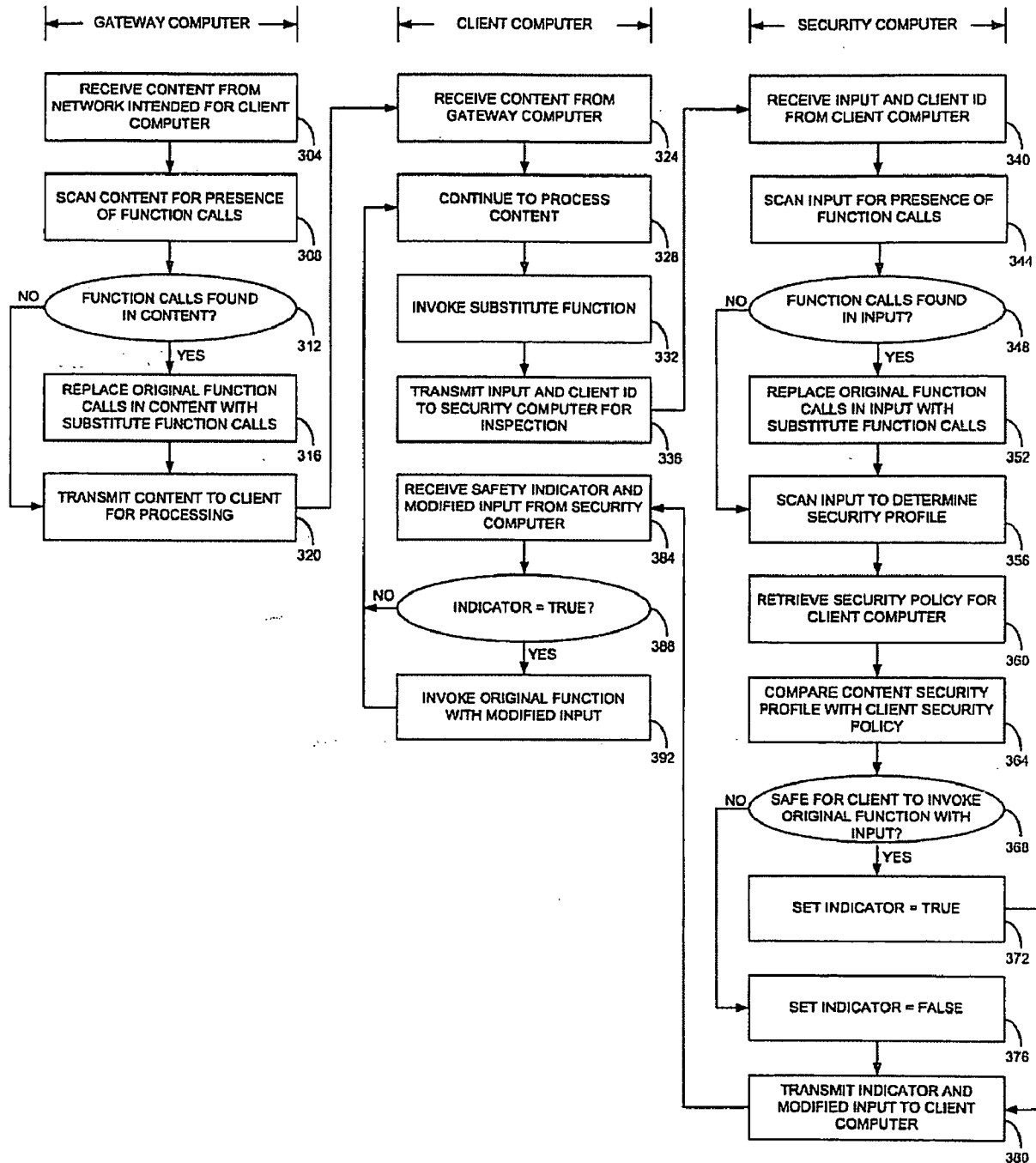


FIG. 3

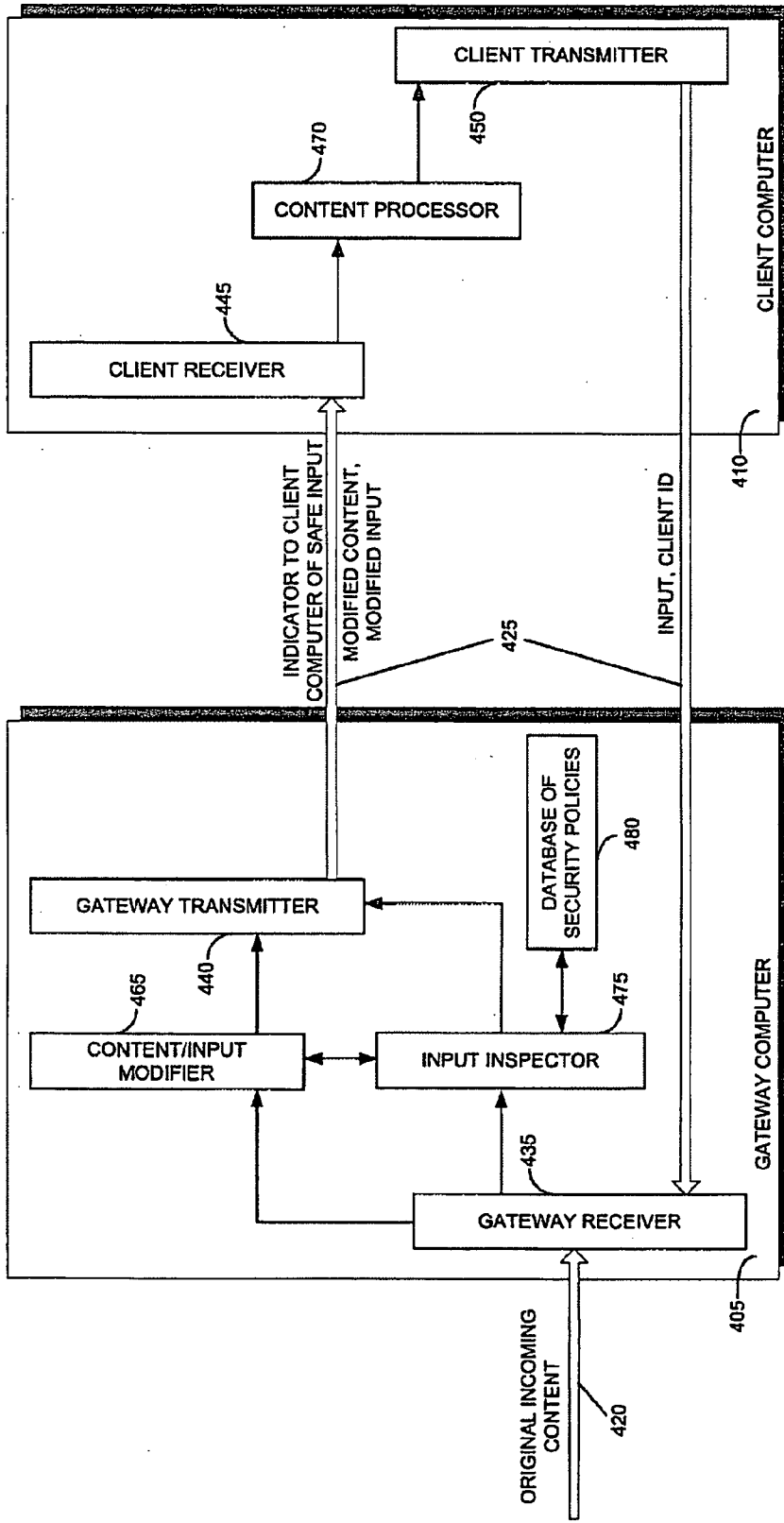


FIG. 4

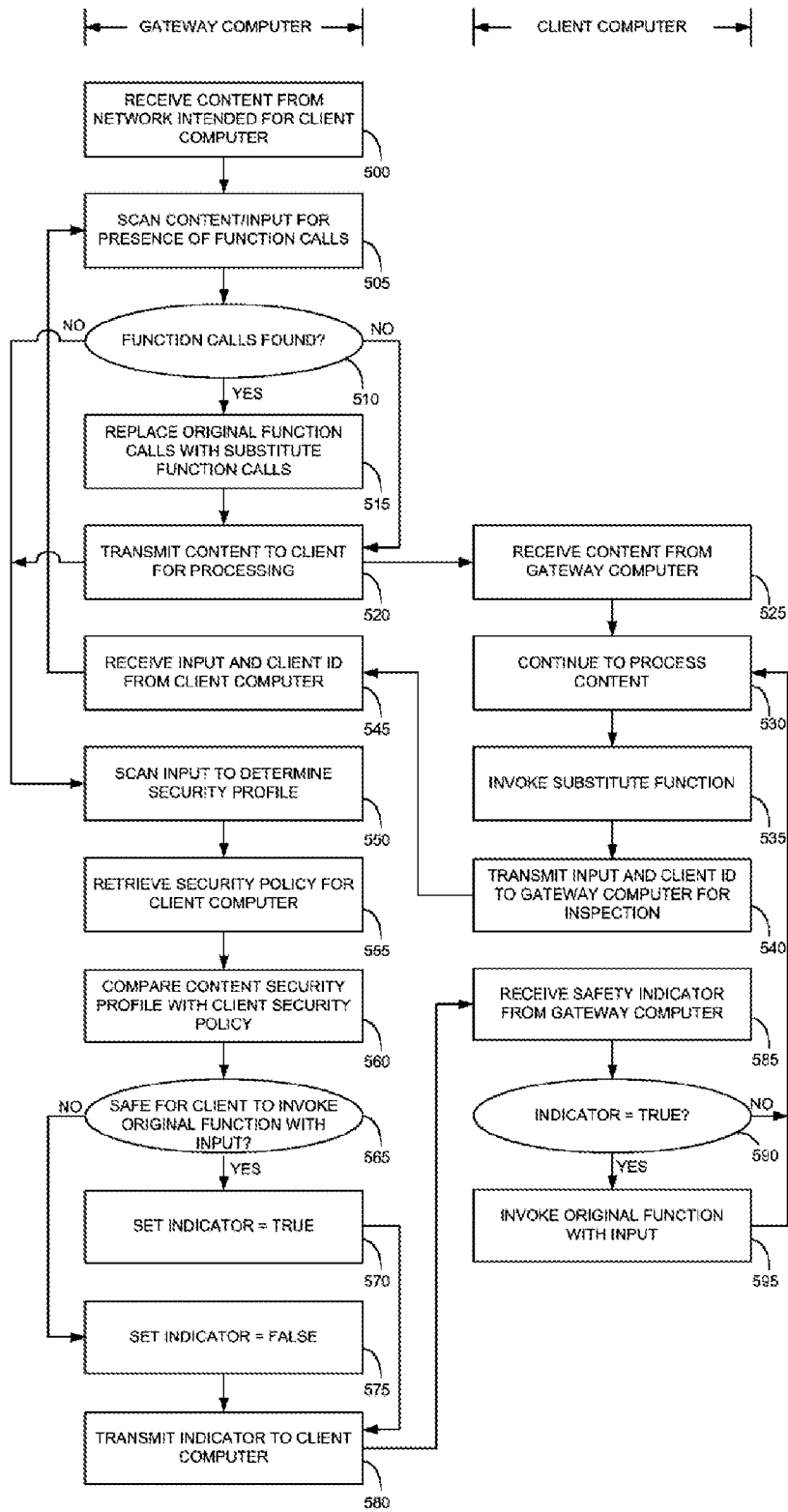


FIG. 5



**DECLARATION AND POWER OF ATTORNEY FOR PATENT APPLICATION**

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below under my name.

I believe that I am the original and first sole inventor or an original and first joint inventor of the subject matter which is claimed and for which a patent is sought on the invention entitled:

**SYSTEM AND METHOD FOR INSPECTING DYNAMICALLY GENERATED EXECUTABLE CODE**

the Specification of which

- is attached hereto
- was filed on **December 12, 2005**  
as United States Application Number or PCT International Application No. **11/298,475**  
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 examination of this application in accordance with Title 37, Code of Federal Regulations, 1.56(a).

I hereby claim foreign priority benefits under Title 35, United States Code, §119 of any provisional application filed in the United States in accordance with 35 U.S.C. §1.119(e), or any application for patent that has been converted to a Provisional Application within one (1) year of its filing date, or any foreign application(s) for patent or inventor's certificate listed below and have also identified below any foreign application for patent or inventor's certificate having a filing date before that of the application on which priority is claimed.

PRIOR FILED APPLICATION(S)

<u>APPLICATION NUMBER</u>	<u>COUNTRY</u>	<u>(DAY/MONTH/YEAR FILED)</u>	<u>PRIORITY CLAIMED</u>
---------------------------	----------------	-------------------------------	-------------------------

I hereby claim the benefit under Title 35, United States Code, §120 of any United States application listed below, and, insofar as the subject matter of each of the claims of this application is not disclosed in any prior United States application in the manner provided by the first paragraph of Title 35, United States Code, §112, I acknowledge the duty to disclose material information as defined in Title 37, Code of Federal Regulations, §1.56(a), which

EPLC

Attorney Docket No.: P-9216-US

occurred between the filing date of the prior application and the national or PCT international filing date of this application:

APPLICATION NO.	FILING DATE (DAY/MONTH/YEAR)	STATUS - PATENTED, PENDING, ABANDONED
-----------------	------------------------------	---------------------------------------

I hereby appoint as my attorney(s) and agent(s) Vladimir Sherman (Attorney, Registration No. 43,116) said attorney(s) and agent(s) with full power of substitution and revocation to prosecute this application and transact all business in the Patent and Trademark Office connected therewith.

Please address all correspondence regarding this application to:

EITAN LAW GROUP  
C/O Landon IP Inc.  
1700 Diagonal Road  
Suite 450  
Alexandria, VA 22314

Direct all telephone calls to (703) 486-1150 and all facsimiles at (703) 892-4510.

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 issued thereon.

FULL NAME OF INVENTOR: GRUZMAN, David

FULL RESIDENCE ADDRESS: Zohar 7/5, Ramat Gan, Israel

COUNTRY OF CITIZENSHIP: Israeli

FULL POST OFFICE ADDRESS: same

SIGNATURE OF INVENTOR X [Signature]

DATE 07/03/2006  
(day / month / year)

Attorney Docket No.: P-9216-US

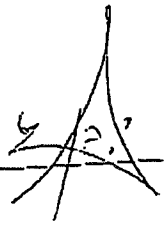
FULL NAME OF INVENTOR: BEN-ITZHAK, Yuval

FULL RESIDENCE ADDRESS: King David Boulevard 36/8, Tel Aviv, Israel

COUNTRY OF CITIZENSHIP: Israeli

FULL POST OFFICE ADDRESS: same

SIGNATURE OF INVENTOR \_\_\_\_\_



DATE \_\_\_\_\_

1 2 2006  
(day / month / year)



Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

**STATEMENT UNDER 37 CFR 3.73(b)**Applicant/Patent Owner: Finjan, Inc.Application No./Patent No.: To Be Assigned Filed/Issue Date: HerewithEntitled: System and Method For Inspecting Dynamically Generated Executable CodeFinjan, Inc., a corporation

(Name of Assignee)

(Type of Assignee, e.g., corporation, partnership, university, government agency, etc.)

states that it is:

1.  the assignee of the entire right, title, and interest; or
2.  an assignee of less than the entire right, title, and interest

The extent (by percentage) of its ownership interest is \_\_\_\_\_ %  
in the patent application/patent identified above by virtue of either:

- A.  An assignment from the inventor(s) of the patent application/patent identified above. The assignment was recorded in the United States Patent and Trademark Office at Reel \_\_\_\_\_, Frame \_\_\_\_\_, or for which a copy thereof is attached.

**OR**

- B.  A chain of title from the inventor(s), of the patent application/patent identified above, to the current assignee as shown below:

1. From: Yuval Ben-Itzhak To: Finjan Software, Ltd.  
The document was recorded in the United States Patent and Trademark Office at Reel \_\_\_\_\_, Frame \_\_\_\_\_, or for which a copy thereof is attached.
2. From: Finjan Software, Ltd. To: Finjan, Inc.  
The document was recorded in the United States Patent and Trademark Office at Reel \_\_\_\_\_, Frame \_\_\_\_\_, or for which a copy thereof is attached.
3. From: \_\_\_\_\_ To: \_\_\_\_\_  
The document was recorded in the United States Patent and Trademark Office at Reel \_\_\_\_\_, Frame \_\_\_\_\_, or for which a copy thereof is attached.

- Additional documents in the chain of title are listed on a supplemental sheet.

As required by 37 CFR 3.73(b)(1)(i), the documentary evidence of the chain of title from the original owner to the assignee was, or concurrently is being, submitted for recordation pursuant to 37 CFR 3.11.

**[NOTE:** A separate copy (i.e., a true copy of the original document(s)) must be submitted to Assignment Division in accordance with 37 CFR Part 3, if the assignment is to be recorded in the records of the USPTO. See MPEP 302.08]

The undersigned (whose title is supplied below) is authorized to act on behalf of the assignee.

/Dawn-Marie Bey/June 14, 2010

Signature

Date

Dawn-Marie Bey(202) 626-8978

Printed or Typed Name

Telephone Number

Partner, King & Spalding LLP

Title

This collection of information is required by 37 CFR 3.73(b). 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 1.11 and 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 the form, call 1-800-PTO-9199 and select option 2.

ASSIGNMENT

WHEREAS, **David GRUZMAN and Yuval BEN-ITZHAK** (referred to as “ASSIGNOR”) has invented certain new and useful improvements in an invention entitled **SYSTEM AND METHOD FOR INSPECTING DYNAMICALLY GENERATED EXECUTABLE CODE,**

- for which a utility application for a United States Patent was filed with the United States Patent and Trademark Office on **December 12, 2005**, Serial Number **11/298,475**.
- for which an application for a United States Patent is being submitted to the United States Patent and Trademark Office herewith; and

WHEREAS, **Finjan Software, Ltd.**, having an office at **Hamachshev St. 1, New Industrial Area, Netanya, 42504, Israel** (hereinafter referred to as the “ASSIGNEE”), is desirous of acquiring the entire right, title and interest in and to said invention, and in and to said application and any Letters Patent that may issue thereon;

NOW, THEREFORE, for and in consideration of One Dollar (\$1.00), and other good and valuable consideration, the receipt of which is hereby acknowledged, ASSIGNOR hereby sells and transfers to said ASSIGNEE, and to ASSIGNEE’S successors and assigns, ASSIGNOR’S entire right, title and interest in and to said invention in the United States and its territorial possessions and in all foreign countries and to all Letters Patent or similar legal protection in the United States and its territorial possessions and in any and all foreign countries to be obtained for said invention by said application or any patent application claiming priority to the application, or any continuation, division, continuation-in-part, reexamination, renewal, substitute, extension or reissue thereof or any legal equivalent thereof in a foreign country for the full term of terms for which the same may be granted; and authorize and request the Commissioner of Patents of the United States and any official of any foreign country whose duty it is to issue patents or legal equivalents thereto, to issue same for this invention to ASSIGNEE, its lawful successors and assigns.

ASSIGNOR further covenants that ASSIGNEE will, upon its request, be provided promptly with all pertinent facts and documents relating to said application, said invention and said Letters Patent and legal equivalents in foreign countries as may be known and accessible to ASSIGNOR and will testify as to the same in any interference or litigation related thereto and will promptly execute and deliver to ASSIGNEE or its legal representative any and all papers, instruments or affidavits required to apply for, obtain, maintain, issue and enforce said application, said invention and said Letters Patent and said equivalents thereof in any foreign country which may be necessary or desirable to carry out the purposes thereof.

IN WITNESS WHEREOF, I/We have hereunto set hand and signed on the date indicated below:

## SIGNATURE(S)

The signature(s) must correspond with the name(s) of the inventor(s) above.

INVENTOR(S)

DATE SIGNED

1) \_\_\_\_\_  
David GRUZMAN



2) \_\_\_\_\_  
Yuval BEN-ITZHAK

February 22, 2009

Employment Contract

Entered into and signed in Netanya this 22<sup>nd</sup> day of March, 2004

Between: Finjan Software Ltd.  
1 HaMachshev Street, Beit Shoham  
Industrial Zone, Netanya South  
(Hereinafter: "the Company" or "the Employer")

Party of the first part

And: David Grozman  
Identity No. 314052382  
7/5 Zohar Street, Ramat Gan  
(Hereinafter: "the Employee")

Party of the second part

Whereas: The Employee wishes to work for the Company in his areas of occupation, in accordance with that which has been set forth in this Contract;

And whereas: The Company wishes to employ the Employee in his areas of occupation, pursuant to the terms that have been set forth in this Contract;

And whereas: The Parties wish to govern their mutual rights and obligations within the framework of this Employment Contract;

[Initials]

It is accordingly declared, stipulated and agreed between the Parties as follows:

■ [REDACTED]

■ [REDACTED]

■ [REDACTED]

■ [REDACTED]

■ [REDACTED]

■ [REDACTED]

■ [REDACTED]

■ [REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]



[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

**10. Proprietary right in inventions**

- 10.1 The property rights in anything related to or deriving from the work of the Employee, including any invention that the Employee shall discover, develop, upgrade or invent or to the invention of which he shall be a party, the discovery or development of which was made in his term of work or consequent on his work for the Employee, whether or not such rights are statutorily able to be registered, shall belong to the Employer, and the Employee shall not be entitled, in respect thereof, to any consideration or royalties whatsoever in respect of the invention or the use thereof.
- 10.2 If the Employer should decide to protect the invention by means of registration of a patent in Israel or abroad, that Employee must cooperate with the Employer, and all including the execution of any document and the delivery of any material or information as may be required for the submission of the application for making the registration.
- 10.3 Subsection 10.1 above shall also apply to an invention that the Employee discovers, develops or invents during the period of one year from the date the labor relations between him and the Employer reached a conclusion for any reason whatsoever, if the Employee uses and resorts to the information and/or material that reached him or came to his knowledge pursuant to his work [with the Company].

10.4 The Employee hereby confirms that he neither has nor shall have any rights, demands or claims in connection with inventions and/or developments as aforesaid, including rights to payments and/or royalties, and that All of the rights including the rights to payments for the inventions and/or the developments belong to the Employer.

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

In witness whereof the Parties have set their hand at the place and time that have been set forth in the preface to this Contract:

[Signature]

Finjan Software Ltd.  
("the Employer")

[Signature]

("the Employee")

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

## SCHEDULE C2

### ASSIGNMENT OF PATENT RIGHTS

WHEREAS, Finjan Software, Ltd., an Israeli corporation (the "Company"), is either the sole and exclusive owner or has an ownership interest in the Patents/Applications in Exhibit A; and

WHEREAS, Finjan, Inc. ("Finjan"), doing business at 2025 Gateway Place, Suite 180, San Jose, California, 95110, is desirous of acquiring, and the Company is desirous of assigning to Finjan, all of the right, title, and interest of the Company into said Patents/Applications, and the inventions disclosed therein and covered thereby.

NOW THEREFORE, for good and valuable consideration, the receipt and sufficiency of which is hereby acknowledged, the Company and Finjan agree as follows:

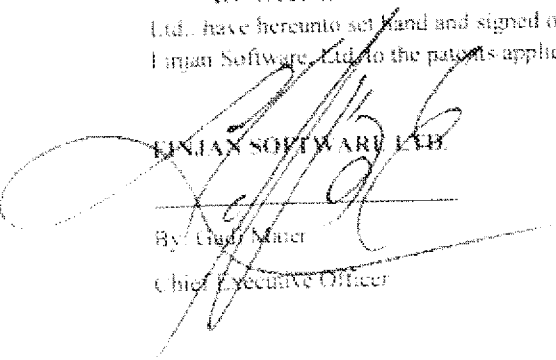
1. The Company is the sole and exclusive owner of all right, title, and interest in and to the Patents and does hereby sell, assign, transfer and set over to Finjan, all of the Company's right, title and interest to the Patents, and to any and all inventions described in the Patents/Applications, in the United States, its territorial possessions and all foreign countries, and in any and all continuations-in-part, continuations, divisions, substitutes, reissues, extensions thereof, and all other applications for letters patent relating thereto that have been or shall be filed in the United States, its territorial possessions and/or any foreign countries, and all rights, together with all priority rights, under any of the international conventions, unions, agreements, acts, and treaties, including all future conventions, unions, agreements, acts, and treaties, the same to be held and enjoyed by Finjan for its own use and enjoyment, and for the use and enjoyment of its successors, assigns or other legal representatives, to the end of the term or terms for which letters patent are or may be granted or reissued as fully and entirely to the same extent as the same would have been held and enjoyed by the Company, if this assignment and sale had not been made; together with all claims for damages or injunctive relief by reason of infringements of such letters patent resulting from the Patent, with the right to sue for past infringement, and collect the same for its own use and behalf and for the use and behalf of its successors, assigns or other legal representatives.
2. The Company hereby authorizes and requests the Commissioner of Patents and Trademarks to issue any and all letters patents of the United States on such inventions or resulting from the Patent, or any continuations-in-part, continuations, divisions, substitutes, reissues or extensions thereof, to Finjan, as assignee of the Company's entire interest, and hereby covenants that the Company has full right to convey the interests herein assigned, and that it has not executed, and will not execute, any agreement in conflict herewith.

The Company agrees that upon request by Finjan, or its successors, assigns or other legal representatives that the Company or its successors, assigns or other legal representatives shall do all other legal acts reasonably necessary to carry out the intent of this assignment at the assignee's expense and request as well as provide such other material, information, or assistance as assignee or its successors, assigns or other legal representatives may consider necessary.

*[Remainder of the page intentionally left blank]*

IN WITNESS WHEREOF, I, Gadi Maier, Chief Executive Officer of Finjan Software, Ltd., have hereunto set hand and signed on the 2<sup>nd</sup> day of November, 2009, assigning the rights of Finjan Software, Ltd. to the patents applications listed in Schedule A to Finjan, Inc.

FINJAN SOFTWARE LTD.

By:   
Chief Executive Officer



ANNEX A

Country/ Pat./App. No.	Title
U.S. / 7,058,822	Malicious Mobile Code Runtime Monitoring System and Methods
U.S. / 12/471,942	Malicious Mobile Code Runtime Monitoring System and Methods
U.S. / 6,804,780	System and Method For Protecting a Computer and a Network From Hostile Downloadables
U.S. / 6,092,194	System and Method For Protecting a Computer and a Network From Hostile Downloadables
U.S. / 90/009,175	System and Method For Protecting a Computer and a Network From Hostile Downloadables
U.S. / 6,154,844	System and Method for Attaching a Downloadable Security Profile to a Downloadable
U.S. / 6,480,962	System and Method For Protecting a Client During Runtime From Hostile Downloadables
U.S. / 90/008,678	System and Method For Protecting a Client During Runtime From Hostile Downloadables
U.S. / 6,167,520	System and Method For Protecting a Client From Hostile Downloadables
U.S. / 90/008,684	System and Method For Protecting a Client During Runtime From Hostile Downloadables
U.S. / 6,298,446	Method and System For Copyright Protection of Digital Images Transmitted Over Networks
U.S. / 6,922,693	Method and System For Copy Protection of Images Displayed on a Computer Monitor
U.S. / 6,993,662	Method and System For Copy Protection of Displayed Data Content
U.S. / 6,353,892	Copy Protection of Digital Images Transmitted Over Networks
U.S. / 6,944,822	Method and Apparatus For Preventing Reuse of Text, Images, and Software Transmitted via Networks
U.S. / 6,209,103	Methods and Apparatus for Preventing Reuse of Text, Images and Software Transmitted Via Networks
U.S. / 6,965,968	Policy-Based Caching
U.S. / 10/680,962	Methods and Systems For Auto-Marking, Watermarking, Auditing,

Country/ Pat./App. No.	Title
	Reporting, Tracing and Policy Enforcement Via E-Mail and Networking Systems
U.S. / 11/606,663	System and Method For Appending Security Information to Search Engine Results CA / 2,275,771 System and Method For Protecting a Computer and a Network From Hostile Downloadables
U.S. / 11/159,455	Malicious Mobile Code Runtime Monitoring System and Methods
U.S. / 11/370,114	Method and System for Protecting a Computer and a Network From Hostile Downloadables
IL / 190518	Malicious Mobile Code Runtime Monitoring System and Methods
IL / 147712	Malicious Mobile Code Runtime Monitoring System and Methods
U.S. / 7,418,731	Method and System For Caching at Secure Gateways
EP / 05775457.4	Method and System For Adaptive Rule-Based Content Scanners
CA / 2578792	Method and System For Adaptive Rule-Based Content Scanners
U.S. / 11/009,437	Method and System For Adaptive Rule-Based Content Scanners For Desktop Computers
IL / 181611	Method and System For Adaptive Rule-Based Content Scanners
U.S. / 10/930,884	Method and System For Adaptive Rule-Based Content Scanners
DE / 0965094	System and Method For Protecting a Computer and a Network From Hostile Downloadables
EP / 0965094	System and Method For Protecting a Computer and a Network From Hostile Downloadables
FR / 0965094	System and Method For Protecting a Computer and a Network From Hostile Downloadables
IL / 129729	System and Method For Protecting a Computer and a Network From Hostile Downloadables
IT / 0965094	System and Method For Protecting a Computer and a Network From Hostile Downloadables
JP / 3952315	System and Method For Protecting a Computer and a Network From Hostile Downloadables
NL / 0965094	System and Method For Protecting a Computer and a Network From Hostile Downloadables

Country/ Pat./App. No.	Title
UK / 0965094	System and Method For Protecting a Computer and a Network From Hostile Downloadables
EP / 99122069.0	Method and System For Copyright Protection of Digital Images Transmitted Over Networks
U.S. / 7,076,469	Copyright Protection of Digital Images Transmitted Over Networks
U.S. / 7,155,743	Method and System For Controlling Use of a Dynamically Linked Software Library
U.S. / 7,155,744	Copyright Protection of Digital Images Transmitted Over Networks
U.S. / 7,281,272	Method and System For Copyright Protection of Digital Images
U.S. / 11/169,823	Method and System For Copy Protection of Displayed Data Content
U.S. / 10/141,308	Method and System For Real-Time Control of Document Printing
U.S. / 7,185,358	Method and Apparatus For Preventing Reuse of Text, Images, and Software Transmitted Via Networks
IL / 127093	Copy Protection
IL / 127869	Network File Copy Protection
U.S. / 10/768,920	Method and System For Embedding Messages Within HTTP
U.S. / 11/298,475	System and Method For Inspecting Dynamically Generated Executable Code
U.S. / 11/354,893	System and Method For Enforcing a Security Context on a Downloadable
U.S. / 12/174,592	Computer Security Method and System With Input Parameter Validation
U.S. / 11/606,707	System and Method For Appending Security Information to Search Engine Results
EP / 06821605.0	System and Method For Appending Security Information to Search Engine Results
U.S. / 11/797,539	Byte-Distribution Analysis of File Security
U.S. / 12/178,558	Splitting an SSL Connection Between Gateways

**Docket No. FIN0008-DIV1**

**PATENT**

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

In re Patent Application of

**David GRUZMAN, et al.**

Serial No.: **To Be Assigned**                      Group Art Unit: **To Be Assigned**

Filed: **Herewith**                                      Examiner: **To Be Assigned**

For: **SYSTEM AND METHOD FOR INSPECTING DYNAMICALLY  
GENERATED EXECUTABLE CODE**

**INFORMATION DISCLOSURE STATEMENT  
UNDER 37 C.F.R. §§ 1.97 AND 1.98**

U.S. Patent and Trademark Office  
Customer Window, Mail Stop Amendment  
Randolph Building  
401 Dulany Street  
Alexandria , VA 22314

Sir:

In accordance with the requirements of 37 C.F.R. §§ 1.56, 1.97-1.98 and MPEP § 609, the references noted on the attached Form PTO-1449 are hereby brought to the attention of the Examiner.

No fees are believed to be necessary since the references cited in this statement are being submitted before the First Office Action. However, the Commissioner is hereby authorized to charge any additional fees which may be required, or to credit any overpayment, to Deposit Account No. 50-4402.

The above information is presented so that the United States Patent and Trademark Office may, in the first instance, determine any materiality thereof to the claimed invention. See

37 C.F.R. §§ 1.104(a) conferring the PTO duty to consider and use any such information. It is respectfully requested that the information be expressly considered during the prosecution of this application, and that the references be made of record therein and appear among the "References Cited" on any patent to issue therefrom.

Respectfully submitted,

Date: June 9, 2010

By: /Dawn-Marie Bey - 44,442/  
Dawn-Marie Bey  
Registration No. 44,442

KING & SPALDING LLP  
1700 Pennsylvania Avenue, N.W.  
Suite 200  
Washington, DC 20006  
(202) 737-0500

15157/105014  
Doc. No. 1496663

Form PTO-1449 (Rev. 2-32)	U.S. Department of Commerce Patent & Trademark Office	Atty. Docket No. <b>FIN0008-DIV1</b>	Serial No. <b>To Be Assigned</b>
<b>INFORMATION DISCLOSURE STATEMENT</b> <i>(Use several sheets if necessary)</i>		Applicant <b>David GRUZMAN, et al.</b>	
		Filing Date <b>Herewith</b>	Group <b>To Be Assigned</b>

**U.S. PATENT DOCUMENTS**

Examiner Initial		Document Number	Date	Name	Class	Sub-Class	Filing Date (if appropriate)
	*	7,313,822	12/25/07	Ben-Itzhak	726	24	3/16/01
	*	7/287,279	10/23/07	Bertman, et al.	726	23	10/1/04
	*	7,203,934	4/10/07	Souloglou, et al.	717	146	6/6/02
	*	2007/0016948	1/18/07	Dubrovsky, et al.	726	22	7/15/05
	*	2006/0161981	7/20/06	Sheth, et al.	726	22	1/19/05
	*	2006/0015940	1/19/06	Zamir, et al.	726	22	7/14/04
	*	6,965,968	11/15/05	Touboul	711	118	2/27/03
	*	6,934,857	8/23/05	Bartleson, et al.	726	5	11/27/00
	*	2005/0108562	5/19/05	Khazan, et al.	726	23	6/18/03
	*	2004/0158729	8/12/04	Szor	713	190	2/6/03
	*	2004/0153644	8/5/04	McCorkendale, et al.	713	156	2/5/03
	*	2004/0133796	7/8/04	Cohen, et al.	726	24	1/3/03
	*	2002/0116635	8/22/02	Sheymov	726	24	2/14/02
	*	6,272,641	8/7/01	Ji	726	24	11/9/99
	*	6,167,520	12/26/00	Touboul	726	23	1/29/97
	*	6,092,194	7/18/00	Touboul	726	24	11/6/97

<b>EXAMINER</b>	<b>DATE CONSIDERED</b>
-----------------	------------------------

EXAMINER: Initial if citation considered, whether or not citation is in conformance with MPEP 609; draw line through citation if not in conformance and not considered. Include copy of this form with next communication.

**U.S. PATENT DOCUMENTS CONT'D.**

\* Reference cited in parent (Application Serial No. 11/298,475), and not provided herewith.

**Serial No. To Be Assigned (Docket No. FIN0008-DIV1)  
Information Disclosure Statement**

	*	5,983,348	11/9/99	Ji	726	13	9/10/97
	*	5,974,549	10/26/99	Golan	726	23	3/27/97
	*	5,359,659	10/25/94	Rosenthal	726	24	6/19/92
<b>FOREIGN PATENT DOCUMENTS</b>							
<b>OTHER DOCUMENTS (Including Author, Title, Date, Pertinent Pages, Etc.)</b>							
	*	International Search Report and Written Opinion for Application No. PCT/IL06/01430, dated July 17, 2008, 9 pp.					
	*	Web printout from <a href="http://www.finjan.com/Content.aspx?id=1456">http://www.finjan.com/Content.aspx?id=1456</a> , printed on 9/10/09, 6 pp.					
	*	Web printout from <a href="http://www.finjan.com/secure_web_gateway.aspx">http://www.finjan.com/secure_web_gateway.aspx</a> , printed on 9/10/09, 7 pp.					
	*	Mark LaDue, "Hostile Applets Home Page," 6 pp., printed 9/10/09					
	*	Mark LaDue, "The Rube Goldberg Approach to Java Security," 1998, 9 pp.					
	*	Mark LaDue, "Drowning in the Surf: A Review of Finjan Software's SurfinShield 2.0," 1997, 6 pp.					
	*	Mark LaDue, "With Trousers Down and Duke Exposed: How Finjan Software Handles Criticism," 1997, 5 pp.					
	*	Huang, et al., "Web Application Security Assessment by Fault Injection and Behavior Monitoring," ACM, 2003, 12 pp.					
	*	"Vital Security Web Appliicance," unknown author, unknown date, 7 pp.					
<b>EXAMINER</b>				<b>DATE CONSIDERED</b>			
EXAMINER: Initial if citation considered, whether or not citation is in conformance with MPEP 609; draw line through citation if not in conformance and not considered. Include copy of this form with next communication.							

15157/105014  
Doc. No. 1496653

\* Reference cited in parent (Application Serial No. 11/298,475), and not provided herewith.

## Electronic Patent Application Fee Transmittal

<b>Application Number:</b>	
<b>Filing Date:</b>	
<b>Title of Invention:</b>	SYSTEM AND METHOD FOR INSPECTING DYNAMICALLY GENERATED EXECUTABLE CODE
<b>First Named Inventor/Applicant Name:</b>	David GRUZMAN, et al.
<b>Filer:</b>	Dawn-Marie Bey./Terry Goad
<b>Attorney Docket Number:</b>	FIN0008-DIV1

Filed as Large Entity

### Utility under 35 USC 111(a) Filing Fees

Description	Fee Code	Quantity	Amount	Sub-Total in USD(\$)
<b>Basic Filing:</b>				
Utility application filing	1011	1	330	330
Utility Search Fee	1111	1	540	540
Utility Examination Fee	1311	1	220	220

**Pages:**

**Claims:**

**Miscellaneous-Filing:**

**Petition:**

**Patent-Appeals-and-Interference:**

000064



Description	Fee Code	Quantity	Amount	Sub-Total in USD(\$)
<b>Post-Allowance-and-Post-Issuance:</b>				
<b>Extension-of-Time:</b>				
<b>Miscellaneous:</b>				
<b>Total in USD (\$)</b>				<b>1090</b>

## Electronic Acknowledgement Receipt

<b>EFS ID:</b>	7803861
<b>Application Number:</b>	12814584
<b>International Application Number:</b>	
<b>Confirmation Number:</b>	9667
<b>Title of Invention:</b>	SYSTEM AND METHOD FOR INSPECTING DYNAMICALLY GENERATED EXECUTABLE CODE
<b>First Named Inventor/Applicant Name:</b>	David GRUZMAN, et al.
<b>Customer Number:</b>	74877
<b>Filer:</b>	Dawn-Marie Bey./Terry Goad
<b>Filer Authorized By:</b>	Dawn-Marie Bey.
<b>Attorney Docket Number:</b>	FIN0008-DIV1
<b>Receipt Date:</b>	14-JUN-2010
<b>Filing Date:</b>	
<b>Time Stamp:</b>	11:52:43
<b>Application Type:</b>	Utility under 35 USC 111(a)

### Payment information:

Submitted with Payment	yes
Payment Type	Credit Card
Payment was successfully received in RAM	\$ 1090
RAM confirmation Number	8323
Deposit Account	504402
Authorized User	BEY,DAWNMARIE

The Director of the USPTO is hereby authorized to charge indicated fees and credit any overpayment as follows:

Charge any Additional Fees required under 37 C.F.R. Section 1.17 (Patent application and reexamination processing fees)

Charge any Additional Fees required under 37 C.F.R. Section 1.19 (Document supply fees)

000000

Charge any Additional Fees required under 37 C.F.R. Section 1.20 (Post Issuance fees)

Charge any Additional Fees required under 37 C.F.R. Section 1.21 (Miscellaneous fees and charges)

**File Listing:**

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1	Transmittal of New Application	FIN0008-DIV_UtilityTransmittal.pdf	63083 fbc2fd41800d79617888f3a0856dfa28ca8e15e	no	1
<b>Warnings:</b>					
<b>Information:</b>					
2		FIN0008-DIV_Spec.pdf	194846 b3c74725b089079f5b1f1ff0f4ea9cabf50572ab	yes	31
	<b>Multipart Description/PDF files in .zip description</b>				
	<b>Document Description</b>	<b>Start</b>	<b>End</b>		
	Specification	1	29		
	Claims	30	30		
	Abstract	31	31		
<b>Warnings:</b>					
<b>Information:</b>					
3	Drawings-only black and white line drawings	FIN0008-DIV_Figures.pdf	339183 3c474b14c1533f68d78ac21949bc4f7d26b94b98	no	5
<b>Warnings:</b>					
<b>Information:</b>					
4	Oath or Declaration filed	FIN0008-DIV_ExecutedDeclaration.pdf	2356509 67b071ccd292e9b60723101b77079b76c51cb0b2	no	3
<b>Warnings:</b>					
<b>Information:</b>					
5	Power of Attorney	FIN0008-DIV_POA.pdf	494208 7d54b62c89ed7c57d1f58d1073f1f6f525bbe68c	no	1
<b>Warnings:</b>					
<b>Information:</b>					
6	Assignee showing of ownership per 37 CFR 3.73(b).	FIN0008-DIV_373b_Assignment.pdf	1321258 e66cfa0c4b2c263820c7719fc1cf6b6c63cdfcb5	no	19
<b>Warnings:</b>					
<b>Information:</b>					

7	Transmittal Letter	FIN0008-DIV_IDStransmittal.pdf	102880 99e5c385e3a6908ad9dd421e66ec17cf751b9ab7	no	2
<b>Warnings:</b>					
<b>Information:</b>					
8	Information Disclosure Statement (IDS) Filed (SB/08)	FIN0008-DIV_IDS1449.pdf	109342 70e477d1edd19086fb611d02bfeaa28bea10d15	no	2
<b>Warnings:</b>					
<b>Information:</b>					
This is not an USPTO supplied IDS fillable form					
9	Fee Worksheet (PTO-875)	fee-info.pdf	33275 005f53e219437c870c3a6dc0a4f8b22fcec32ae	no	2
<b>Warnings:</b>					
<b>Information:</b>					
<b>Total Files Size (in bytes):</b>			5014584		
<p><b>This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.</b></p> <p><b><u>New Applications Under 35 U.S.C. 111</u></b>  <b>If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.</b></p> <p><b><u>National Stage of an International Application under 35 U.S.C. 371</u></b>  <b>If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.</b></p> <p><b><u>New International Application Filed with the USPTO as a Receiving Office</u></b>  <b>If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.</b></p>					

Date: 06/14/10

Approved for use through 7/31/2006. OMB 0651-0032  
 U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

<b>PATENT APPLICATION FEE DETERMINATION RECORD</b> Substitute for Form PTO-875	Application or Docket Number <b>12/814,584</b>
---	---

APPLICATION AS FILED – PART I			SMALL ENTITY		OR		OTHER THAN SMALL ENTITY	
(Column 1) (Column 2)			RATE (\$)	FEE (\$)			RATE (\$)	FEE (\$)
FOR	NUMBER FILED	NUMBER EXTRA	N/A				N/A	<b>330</b>
BASIC FEE (37 CFR 1.16(a), (b), or (c))	N/A	N/A	N/A				N/A	<b>540</b>
SEARCH FEE (37 CFR 1.16(k), (i), or (m))	N/A	N/A	N/A				N/A	<b>220</b>
EXAMINATION FEE (37 CFR 1.16(o), (p), or (q))	N/A	N/A	x\$26				x\$52	
TOTAL CLAIMS (37 CFR 1.16(j))	<b>3</b>	minus 20 =	x\$110				x\$220	
INDEPENDENT CLAIMS (37 CFR 1.16(h))	<b>2</b>	minus 3 = *						
APPLICATION SIZE FEE (37 CFR 1.16(s))	If the specification and drawings exceed 100 sheets of paper, the application size fee due is \$260 (\$130 for small entity) for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR							
MULTIPLE DEPENDENT CLAIM PRESENT (37 CFR 1.16(j))								
			195				390	
			<b>TOTAL</b>				<b>TOTAL</b>	<b>1090</b>

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

APPLICATION AS AMENDED – PART II					SMALL ENTITY		OR		OTHER THAN SMALL ENTITY	
(Column 1) (Column 2) (Column 3)					RATE (\$)	ADDITIONAL FEE (\$)			RATE (\$)	ADDITIONAL FEE (\$)
AMENDMENT A	CLAIMS REMAINING AFTER AMENDMENT	HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA		X =				X =	
	Total (37 CFR 1.16(i))	*	Minus **	=	X =				X =	
	Independent (37 CFR 1.16(h))	*	Minus ***	=	N/A				N/A	
	Application Size Fee (37 CFR 1.16(s))					TOTAL	ADD'T FEE			TOTAL
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))										

AMENDMENT B	CLAIMS REMAINING AFTER AMENDMENT	HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA		X =				X =	
	Total (37 CFR 1.16(i))	*	Minus **	=	X =				X =	
	Independent (37 CFR 1.16(h))	*	Minus ***	=	N/A				N/A	
	Application Size Fee (37 CFR 1.16(s))					TOTAL	ADD'T FEE			TOTAL
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))										

\* 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.

This collection of information is required by 37 CFR 1.16. 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 the form, call 1-800-PTO-9199 and select option 2.