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

Canadian Intellectual Property Office
An agency of Industry Canada

CA 2663232 C 2016/01/05

(11)(21) **2 663 232**

(12) BREVET CANADIEN / CANADIAN PATENT

(13) **C**

(86) **Date de dépôt PCT/PCT Filing Date:** 2007/08/03
(87) **Date publication PCT/PCT Publication Date:** 2008/02/21
(45) **Date de délivrance/Issue Date:** 2016/01/05
(85) **Entrée phase nationale/National Entry:** 2009/03/11
(86) **N° demande PCT/PCT Application No.:** EP 2007/058088
(87) **N° publication PCT/PCT Publication No.:** 2008/019961
(30) **Priorité/Priority:** 2006/08/15 (US11/464,581)

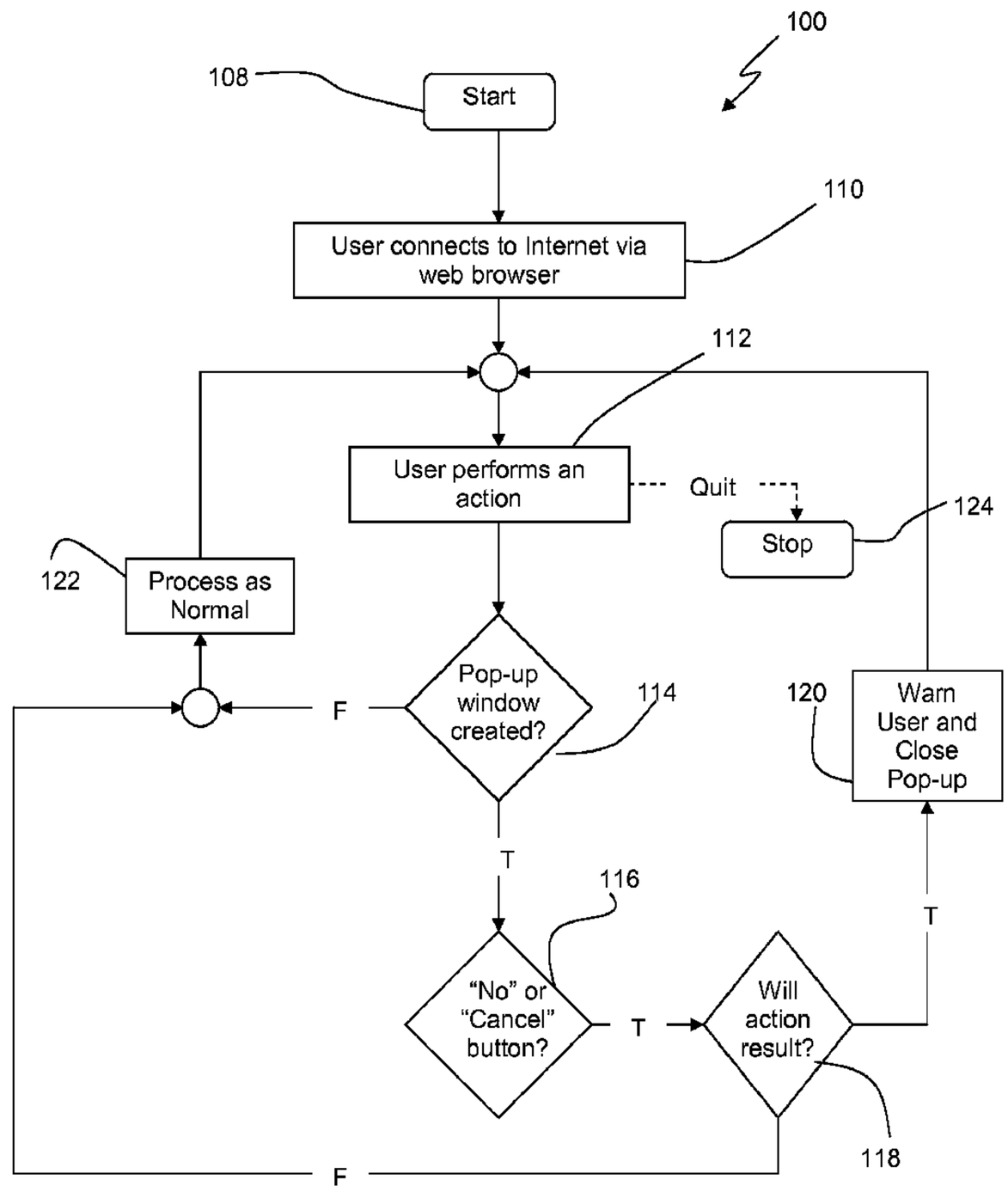
(51) **Cl.Int./Int.Cl.** ***G06F 21/56*** (2013.01)

(72) **Inventeurs/Inventors:**
BOSS, GREGORY JENSEN, US;
CHEN, GANG, US;
HAMILTON, RICK ALLEN, II, US;
LANGFORD, JOHN STEVEN, US

(73) **Propriétaire/Owner:**
INTERNATIONAL BUSINESS MACHINES CORPORATION, US

(74) **Agent:** WANG, PETER

(54) **Titre : PROTECTION DES UTILISATEURS DES FENETRES PUBLICITAIRES D'ENTREES HOSTILES**
(54) **Title: PROTECTING USERS FROM MALICIOUS POP-UP ADVERTISEMENTS**



(57) **Abrégé/Abstract:**
The present invention is a solution for detecting a spoofed command button in a pop-up window. The solution tracks the creation process of a pop-up window, detects the presence of command buttons in the pop-up window, verifies the value labeled on each command button in the pop-up window and determines a follow-up action generated from selecting a command button on the pop-up window.

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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
21 February 2008 (21.02.2008)

PCT

(10) International Publication Number
WO 2008/019961 A1

(51) International Patent Classification:
G06F 21/00 (2006.01)

(21) International Application Number:
PCT/EP2007/058088

(22) International Filing Date: 3 August 2007 (03.08.2007)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
11/464,581 15 August 2006 (15.08.2006) US

(71) Applicant *(for all designated States except US)*: **INTERNATIONAL BUSINESS MACHINES CORPORATION** [US/US]; New Orchard Road, Armonk, New York 10504 (US).

(71) Applicant *(for MG only)*: **IBM UNITED KINGDOM LIMITED** [GB/GB]; PO Box 41, North Harbour, Portsmouth Hampshire PO6 3AU (GB).

(72) Inventors; and

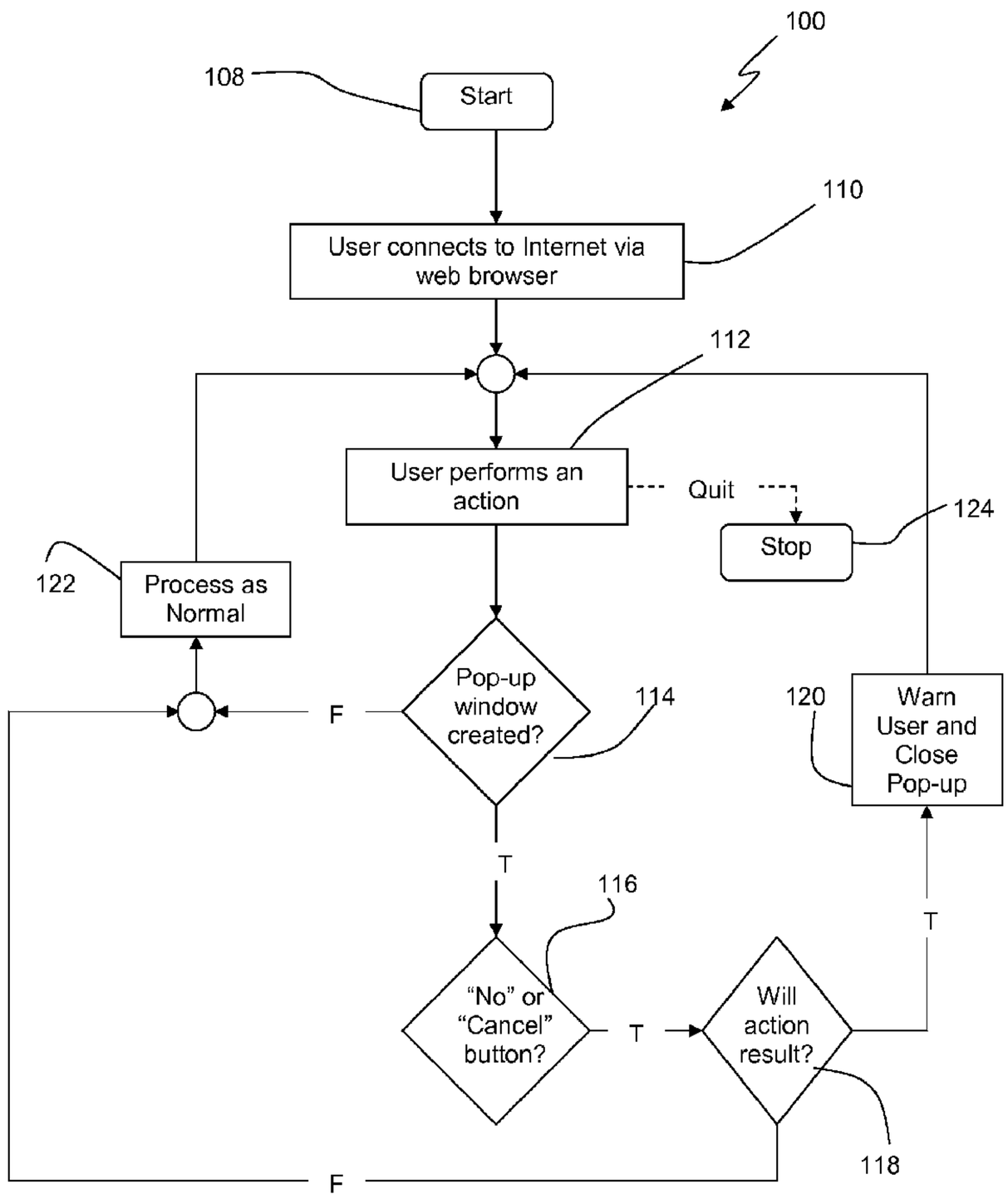
(75) Inventors/Applicants *(for US only)*: **BOSS, Gregory, Jensen** [US/US]; 559 West 860 North, American Fork, Utah 84003 (US). **CHEN, Gang** [CN/US]; 2893 Terrell Avenue, Oceanside, New York 11572 (US). **HAMILTON II, Rick, Allen** [US/US]; 1532 Dairy Road, Charlottesville, Virginia 22903 (US). **LANGFORD, John, Steven** [US/US]; 12320 Willow Bend Drive, Austin, Texas 78758 (US).

(74) Agent: **WILLIAMS, Julian, David**; IBM United Kingdom Limited, Intellectual Property Law, Hursley Park, Winchester Hampshire SO21 2JN (GB).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, SV, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

[Continued on next page]

(54) Title: PROTECTING USERS FROM MALICIOUS POP-UP ADVERTISEMENTS



(57) Abstract: The present invention is a solution for detecting a spoofed command button in a pop-up window. The solution tracks the creation process of a pop-up window, detects the presence of command buttons in the pop-up window, verifies the value labeled on each command button in the pop-up window and determines a follow-up action generated from selecting a command button on the pop-up window.

WO 2008/019961 A1

WO 2008/019961 A1

(84) **Designated States** *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, LV, MC, MT, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report*

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

PROTECTING USERS FROM MALICIOUS POP-UP ADVERTISEMENTS

FIELD OF THE INVENTION

The present invention relates generally to preventing undesired effects of advertisements on the Internet. Particularly, the present invention relates to detecting any undesired follow-up action potentially triggered by selecting a command button on a pop-up advertisement window on a web browser during an online session. More particularly, the present invention relates to the prevention of an undesired follow-up action triggered by closing such a pop-up window.

BACKGROUND OF THE INVENTION

In a typical online session on the Internet, a user will encounter pop-up windows advertising all possible kinds of products and services. These advertisements usually "pop-up" in a separate window on top of the current web browser window. Some pop-up advertisements are harmless and can be easily closed with the click of a command button provided as part of the pop-up window. However, other pop-up windows may provide a command button that is spoofed. The spoofed command button may trigger a hidden action like installing "spyware", "adware", stealing computer cycles, sending spam via the user's computer or other undesirable applications. Figure 3 is an example of a pop-up window appearing to warn of adware or spyware, but may itself have spoofed buttons that would install a malicious program irrespective of whether the "Yes" or "No" button is selected. Although pop-up advertisements may have a system-supplied "Cancel" button that cannot be spoofed, namely the "X" button on the corner of a window, such a system-supplied button can be grayed-out, concealed or disabled. Some users may not know the difference between an explicit command button and a system-supplied "Cancel" button. There are also other pop-up windows that do not provide any command buttons to close the pop-up window other than affirmative command buttons like "Yes" or "Confirm" buttons. Figure 4 is an example of such a pop-up window. This restricts user options to close the pop-up window, which would inevitably trigger possible undesirable follow-up actions unknown to the user. In attempting to close such malicious pop-up windows, a computer user risks authorizing a

malicious action, which the computer would take to be an action consciously made by the user.

There are efforts to provide methods to avoid malicious pop-up windows like scanning malicious program code when a user has consciously given a command to download a program, where the scanning is conducted by comparing with a pre-existing set of programs prior to downloading the program code. Other efforts provide a method to escape from a display model dialog box, generated by an error Java applet, by diverting user input from the applet to the main browser loop, and receiving user key press command to execute a close window. This prior art is directed to model dialog boxes created in the Java programming language.

Thus, there exists a need to overcome at least one of the preceding deficiencies and/or limitations of the related art.

SUMMARY OF THE INVENTION

The present invention provides a method, a system and a computer program product for detecting a spoofed command button in a pop-up window and/or forewarning the user of potential harm in selecting the spoofed command button to close the pop-up window.

A first aspect of the present invention provides a method for detecting a spoofed command button, the method comprises the steps of: tracking a pop-up window creation process; detecting a command button created in the pop-up window; checking an assigned value of the command button; and determining a follow-up action generated on selection of the command button.

Preferably, the present invention provides a method wherein the step of detecting comprises performing a background source code validation.

Preferably, the present invention provides a method further comprises the step of verifying that the follow-up action generated closes the pop-up window.

Preferably, the present invention provides a method further comprising the step of alerting a user that the pop-up window poses a potential danger when the follow-up action from selecting the command button is an action other than closing the pop-up window.

Preferably, the present invention provides a method further comprising the step of alerting a user to proceed with caution when the assigned value of the command button restricts user selection.

A second aspect of the present invention provides a method for detecting a spoofed command button comprising: validating a closing action of pop-up window against a web browser Application Program Interface (API) for closing a window to detect an anomaly; and alerting a user on detecting that a follow-up action generated by the user on selecting the command button fails to communicate with the web browser API.

Preferably, the present invention provides a method further comprising the step of closing the pop-up window on alerting the user.

A third aspect of the present invention provides a system for detecting a spoofed command button comprising: a component for tracking a pop-up window creation process; a component for detecting creation of a command button in the pop-up window; a component for checking an assigned value of the command button; and a component for determining a follow-up action generated on selection of the command button. The system is a computer program that can be added to an existing computer program as a plug-in, extension, agent or any means that can be applied in conjunction with a web browser.

Preferably, the present invention provides a system wherein the component for detecting performs a background source code validation.

Preferably, the present invention provides a system further comprising a component for verifying that the follow up action generated closes the pop-up window.

Preferably, the present invention provides a system further comprising a component for alerting a user that the pop-up window poses a potential danger when the follow-up action performs an action other than closing the pop-up window.

Preferably, the present invention provides a system further comprising a component for alerting the user to proceed with caution when the assigned value of the command button restricts user selection.

Preferably, the present invention provides a system further comprising: a component for validating the pop-up window against a web browser Application Program Interface (API) for closing a window; a component for alerting the user on detecting the follow-up action generated by the user on selecting the command button omits communicating with the web browser API; and a component for closing the pop-up window.

A fourth aspect of the present invention provides a computer program stored on a machine-readable medium for detecting a spoofed command button, the computer readable program performing: tracking a pop-up window creation process; detecting a command button created in the pop-up window; checking an assigned value of the command button; and determining a follow-up action generated on selection of the command button.

A fifth aspect of the present invention provides a method for deploying an application for detecting a spoofed command button comprising: providing a computer infrastructure being operable to: track a pop-up window creation process; detect a command button created in the pop-up window; check an assigned value of the command button; and determine a follow-up action generated on selection of the command button.

BRIEF DESCRIPTION OF THE DRAWINGS

The accompanying drawings are schematic representations to illustrate a typical embodiment of the present invention and not intended to limit the principles of the invention. In the drawings, like numbering represents like elements between the drawings.

FIG. 1 is a flow chart illustrating an algorithm of one embodiment of the present invention.

FIG. 2 is a flow chart illustrating an algorithm of an alternative embodiment of the present invention.

FIG. 3 is a sample pop-up window illustrating a typical advertisement providing command buttons for the user to select to close the pop-up window.

FIG. 4 is another sample pop-up window illustrating a restriction of user selection to close the pop-up window.

FIG. 5 is a flow chart illustrating an algorithm of another embodiment of the present invention.

FIG. 6 depicts an illustrative computer system for implementing embodiment(s) of the present invention.

DETAILED DESCRIPTION OF THE INVENTION

Different approaches have been adopted to prevent malicious codes from executing a variety of actions, but there remains the need to provide a method to detect and alert the user of malicious code that would be installed into the computer should the user issue a command to close a pop-up window.

Turning to the Figures, FIG. 1 illustrates a process flow of an algorithm 100 for the detection of a pop-up window with a spoofed command button. A spoofed button is one where the result generated by clicking on the button does not return an expected result. The spoofed command button has the appearance of a typical command button, usually appearing in pairs (e.g. “Yes” and “No” as illustrated in Figure 3) on a pop-up window. A user would be led to believe that such a command button provides an avenue for issuing a desired command by simply clicking on the button. However, a different action may be triggered on

clicking on the command button in the pop-up window. The algorithm 100 is on stand-by 110 upon connection to the Internet via a web browser from the start 108. When a user performs an action 112 during the online session on the Internet, the algorithm 100 tracks a call to open a window (e.g. "window.open" in Java) to determine 114 if a pop-up window is being created. The algorithm 100 includes a step 116 to detect if a command button is created as part of the pop-up window by conducting a background page source code validation. The value of the command button is compared to the value of a "No" or "Cancel" button. Where the value matches, a further check 118 is conducted to determine if the follow-up action is simply "return false" (e.g. close the window without performing any other action) on clicking the command button. If so, the follow-up action is processed 122 without any interruption. If, however, the follow-up action is not "return false", the user will be alerted of the spoofed command button and the pop-up window will be closed 120 automatically by the system. This safeguards the user from clicking on the spoofed button and hence minimizes the risk of triggering malicious follow-up actions associated with clicking of the spoofed button. The algorithm 100 is then reset to be on stand-by again in anticipation of another user action that will trigger creation of a new pop-up window creation. If it is determined 114 that a pop-up window is not created, the algorithm 100 returns to the stand-by mode while the web browser processes the user command as desired 122. If the user elects to quit from the online session, the algorithm 100 registers the command and stops detecting the creation of pop-up windows 124.

In FIG. 2, the algorithm 200 has, in addition to the process flow set out in the algorithm 100 in FIG. 1, an additional process flow 202 following from the logic step 116 to address the situation where a pop-up window does not provide a "Cancel" button. This essentially means that there is no command button created with a value that is comparable with the value of a "No" or "Cancel" button for closing the pop-up window. The pop-up window, however, may include a command button that restricts the user option leading to compliance and increasing the risks of triggering malicious or undesirable follow-up actions. Although the user is not given the option to close the pop-up window with a command button that would specifically perform such an action, the web browser would forewarn the user with a message like: *"The pop-up window you are browsing doesn't have an appropriate Cancel function, please proceed with caution."* However, no additional action

would be taken by the web browser to minimize the risks as the command, once issue by the user, will be processed. Alternatively, the command is processed without alerting the user.

FIG. 5 illustrates another embodiment of the present invention in which an algorithm 300 for validating the close action against the standard Operating System (OS) or web browser Application Program Interface (API) to close a window in the process flow is provided. The process flow from step 116 or step 118 to step 122 includes an additional step 302, which validates the close action issued by the command button against the OS of API in order to detect any anomaly. Where the command button to close the pop-up window does not call on the web browser API, this will be detected as an anomaly. The algorithm 300 will then initiate a warning message to the user and close the pop-up window automatically 120. This alternative step 302 can be applied independently as an alternative to the algorithms 100 or 200 set-out above or as a safeguard measure from any accidental selection of the spoofed command button. As a safeguard, this algorithm 300 can be implemented immediately before the logic step 122 to ensure that even where the command button does not appear to be spoofed, should the behavior of the follow-up actions not conform to an expected mode of operation, the user will be alerted and the pop-up window will be closed by the web browser immediately.

Therefore, what is needed is a method that enables web browsers to thwart button spoofing in pop-up advertisement windows. What is also needed is an executable system that can be added on to an existing internet web browser program to detect malicious code that would be executed when a user selects a spoofed command button provided as part of the pop-up window. What is further needed is an executable system that can forewarn a user of possible unintentional actions for selecting a spoofed command button on a pop-up window.

FIG. 6 shows an illustrative system 400 for detecting a spoofed command button in a pop-up window and forewarning a user of potential harm when selecting the spoofed command button in accordance with embodiment(s) of the present invention. To this extent, the system 400 includes a computer infrastructure 402 that can perform the various process steps described herein for detecting a spoofed command button. In particular, the computer

infrastructure 402 is shown including a computer system 404 that comprises a spoofed command button detecting system 430, which enables the computer system 404 to detect the creation of a spoofed command button when a pop-up window is created by performing the process steps of the invention.

The computer system 404 as shown includes a processing unit 408, a memory 410, at least one input/output (I/O) interface 414, and a bus 412. Further, the computer system 404 is shown in communication with at least one external device 416 and a storage system 418. In general, the processing unit 408 executes computer program code, such as spoofed command button detecting system 430, that is stored in memory 410 and/or storage system 418. While executing computer program code, the processing unit 408 can read and/or write data from/to the memory 410, storage system 418, and/or I/O interface(s) 414. Bus 412 provides a communication link between each of the components in the computer system 404. The at least one external device 416 can comprise any device (e.g., display 420) that enables a user (not shown) to interact with the computer system 404 or any device that enables the computer system 404 to communicate with one or more other computer systems.

In any event, the computer system 404 can comprise any general purpose computing article of manufacture capable of executing computer program code installed by a user (e.g., a personal computer, server, handheld device, etc.). However, it is understood that the computer system 404 and spoofed command button detecting system 430 are only representative of various possible computer systems that may perform the various process steps of the invention. To this extent, in other embodiments, the computer system 404 can comprise any specific purpose computing article of manufacture comprising hardware and/or computer program code for performing specific functions, any computing article of manufacture that comprises a combination of specific purpose and general purpose hardware/software, or the like. In each case, the program code and hardware can be created using standard programming and engineering techniques, respectively.

Similarly, the computer infrastructure 402 is only illustrative of various types of computer infrastructures that can be used to implement the invention. For example, in one embodiment, the computer infrastructure 402 comprises two or more computer systems (e.g.,

a server cluster) that communicate over any type of wired and/or wireless communications link, such as a network, a shared memory, or the like, to perform the various process steps of the invention. When the communications link comprises a network, the network can comprise any combination of one or more types of networks (e.g., the Internet, a wide area network, a local area network, a virtual private network, etc.). Regardless, communications between the computer systems may utilize any combination of various types of transmission techniques.

As previously mentioned, the spoofed command button detecting system 430 enables the computer system 404 to perform a return false action check 432 when a command button of a pop-up window 434 is selected by the user. The pop-up window "popping-up" within a web browser 440. To this extent, the spoofed command button detecting system 430 is shown as including a pop-up window creation detecting system 436 for detecting the generation of a pop-up window 434 on the web browser 440, and a command button creation system 438 for performing a return false action check 432 for each command button in pop-up windows 434, based on the value assigned to the command button. Operation of each of these systems is discussed above. It is understood that some of the various systems shown in FIG. 6 can be implemented independently, combined, and/or stored in memory for one or more separate computer systems 404 that communicate over a network. Further, it is understood that some of the systems and/or functionality may not be implemented, or additional systems and/or functionality may be included as part of the system 400.

While shown and described herein as a method and system for detecting if command buttons in pop-up windows are spoofed, it is understood that the invention further provides various alternative embodiments. For example, in one embodiment, the invention provides a computer-readable medium that includes computer program code to enable a computer infrastructure to determine the follow-up action that will be triggered on selecting a command button of a pop-up window. To this extent, the computer-readable medium includes program code, such as the spoofed command button detecting system 430, which implements each of the various process steps of the invention. It is understood that the term "computer-readable medium" comprises one or more of any type of physical embodiment of the program code. In particular, the computer-readable medium can comprise program code

embodied on one or more portable storage articles of manufacture (e.g., a compact disc, a magnetic disk, a tape, etc.), on one or more data storage portions of a computer system, such as the memory 410 and/or storage system 418 (e.g., a fixed disk, a read-only memory, a random access memory, a cache memory, etc.), and/or as a data signal traveling over a network (e.g., during a wired/wireless electronic distribution of the program code).

In another embodiment, the invention provides a business method that performs the process steps of the invention on a subscription, advertising, and/or fee basis. That is, a service provider could offer to determine if a pop-up window carries with it spoofed command buttons that may trigger malicious actions if a user selects the command button. In this case, the service provider can create, maintain, support, etc., a computer infrastructure, such as the computer infrastructure 402, that performs the process steps of the invention for one or more customers. In return, the service provider can receive payment from the customer(s) under a subscription and/or fee agreement and/or the service provider can receive payment from the sale of advertising space to one or more third parties.

In still another embodiment, the invention provides a method of detecting a spoofed command button of a pop-up window. In this case, a computer infrastructure, such as the computer infrastructure 402, can be obtained (e.g., created, maintained, having made available to, etc.) and one or more systems for performing the process steps of the invention can be obtained (e.g., created, purchased, used, modified, etc.) and deployed to the computer infrastructure. To this extent, the deployment of each system can comprise one or more of (1) installing program code on a computer system, such as the computer system 404, from a computer-readable medium; (2) adding one or more computer systems to the computer infrastructure; and (3) incorporating and/or modifying one or more existing systems of the computer infrastructure, to enable the computer infrastructure to perform the process steps of the invention.

As used herein, it is understood that the terms “program code” and “computer program code” are synonymous and mean any expression, in any language, code or notation, of a set of instructions intended to cause a computer system having an information processing capability to perform a particular function either directly or after either or both of

the following: (a) conversion to another language, code or notation; and (b) reproduction in a different material form. To this extent, program code can be embodied as one or more types of program products, such as an application/software program, component software/a library of functions, an operating system, a basic I/O system/driver for a particular computing and/or I/O device, and the like.

Although the preferred embodiments of the present invention have been described herein, the above description is merely illustrative. It is not intended to be exhaustive or to limit the invention to the precise form disclosed, and obviously, many modifications and variations are possible. Therefore, the scope of the claims should not be limited by the preferred embodiments set forth in the examples, but rather should be given the broadest interpretation consistent with the description as a whole.

CLAIMS

1. A computer-implemented method executable on a computing device having a processor and a memory for detecting a spoofed command button, comprising performing by the computing device the following steps:

tracking a creation process for a pop-up window;

detecting a command button created in the pop-up window;

comparing an assigned value of the command button with a value assigned to at least one of a valid "No" or a valid "Cancel" button;

determining a follow-up action generated on selection of the command button;

verifying that the follow-up action generated closes the pop-up window; and

alerting a user that the pop-up window poses a potential danger when the follow-up action from selecting the command button includes an action other than closing the pop-up window.

2. The computer-implemented method according to claim 1 wherein the detecting includes performing a background source code validation.

3. The computer-implemented method according to claim 1 further comprising alerting by the computer device a user to proceed with caution when the assigned value of the command button restricts user selection.

4. A computer-implemented method executable on a computing device having a processor and a memory for detecting a spoofed command button, comprising performing by the computing device the following steps:

validating a closing action of a command button of a pop-up window against a web browser Application Program Interface (API) for closing a window, wherein the validating detects an anomaly when the command button does not call on the web browser API; and

alerting a user in response to detecting the anomaly that a follow-up action generated in response to selecting the command button fails to communicate with the web browser API; and

closing the pop-up window in response to detecting the anomaly.

5. A system embodied in one or more computing devices on a network, each computing device having a processor and a memory, comprising:

at least one computing device configured for detecting a spoofed command button, comprising:

a component executed by the processor for tracking a creation process for a pop-up window;

a component executed by the processorfor detecting creation of a command button in the pop-up window;

a component executed by the processorfor comparing an assigned value of the command button with a value assigned to at least one of a valid "No" or a valid "Cancel" button;

a component executed by the processorfor determining a follow-up action generated on selection of the command button;

a component executed by the processorfor verifying that the follow-up action generated closes the pop-up window; and

a component executed by the processor for alerting a user that the pop-up window poses a potential danger when the follow-up action performs an action other than closing the pop-up window.

6. The system of claim 5, wherein the component for detecting performs a background source code validation.

7. The system of claim 5 further comprising a component executed by the processor for alerting the user to proceed with caution when the assigned value of the command button restricts user selection.

8. The system of claim 5 further comprising a component executed by the processor for validating the pop-up window against a web browser Application Program Interface (API) for closing a window; a component executed by the processorfor alerting the user on detecting the follow-up action generated by the user on selecting the command button omits communicating with the web browser API; and a component executed by the processor for closing the pop-up window.

9. A computer program product storing computer executable program code on a non-transitory machine-readable physical storage medium, for detecting a spoofed command button, the computer program product including program code for:

tracking a creation process for a pop-up window; detecting a command button created in the pop-up window;

comparing an assigned value of the command button with a value assigned to at least one of a valid "No" or a valid "Cancel" button;

determining a follow-up action generated on selection of the command button;

program code for verifying that the follow-up action generated closes the pop-up window; and

program code for alerting the user that the pop-up window poses a potential danger when the follow-up action for selecting a command button includes an action other than closing the pop-up window.

10. The computer program product of claim 9 further including program code for:

validating a closing action of the pop-up window against a web browser Application Program Interface (API) for closing a window;

alerting a user on detecting that the follow-up action generated by the user on selecting the command button fails to communicate with the web browser API; and

closing of the pop-up window.

11. The computer program product of claim 9 further including program code for performing a background source code validation when the command button is detected.

12. The computer program product of claim 9 further including program code for alerting the user to proceed with caution when the assigned value of the command button restricts user selection.

13. A computer-implemented method executable on a computing device having a processor and a memory for deploying an application for detecting a spoofed command button, comprising performing by the computing device the following steps:

providing a computer infrastructure being operable to: track a pop-up window creation process;

detecting a command button created in the pop-up window;

comparing an assigned value of the command button with a value assigned to at least one of a valid "No" or a valid "Cancel" button;

determining a follow-up action generated on selection of the command button;

verifying that the follow-up action generated closes the pop-up window; and

alerting a user that the pop-up window poses a potential danger when the follow-up action from selecting the command button includes an action other than closing the pop-up window.

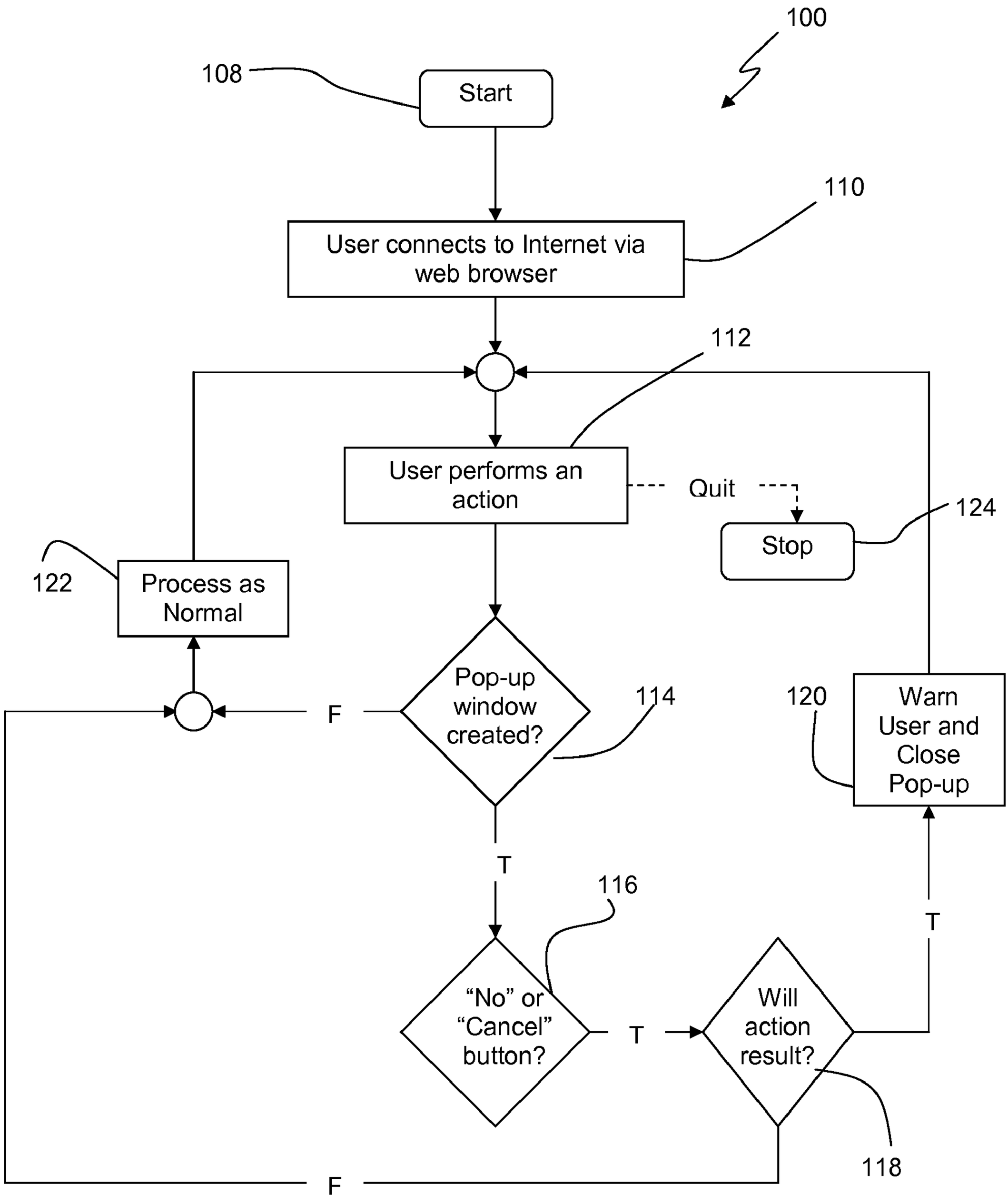

100
108
Start
110
User connects to Internet via web browser
112
User performs an action
Quit
124
Stop
122
Process as Normal
Pop-up window created?
114
F
T
120
Warn User and Close Pop-up
116
"No" or "Cancel" button?
T
Will action result?
T
118
F

FIG. 1

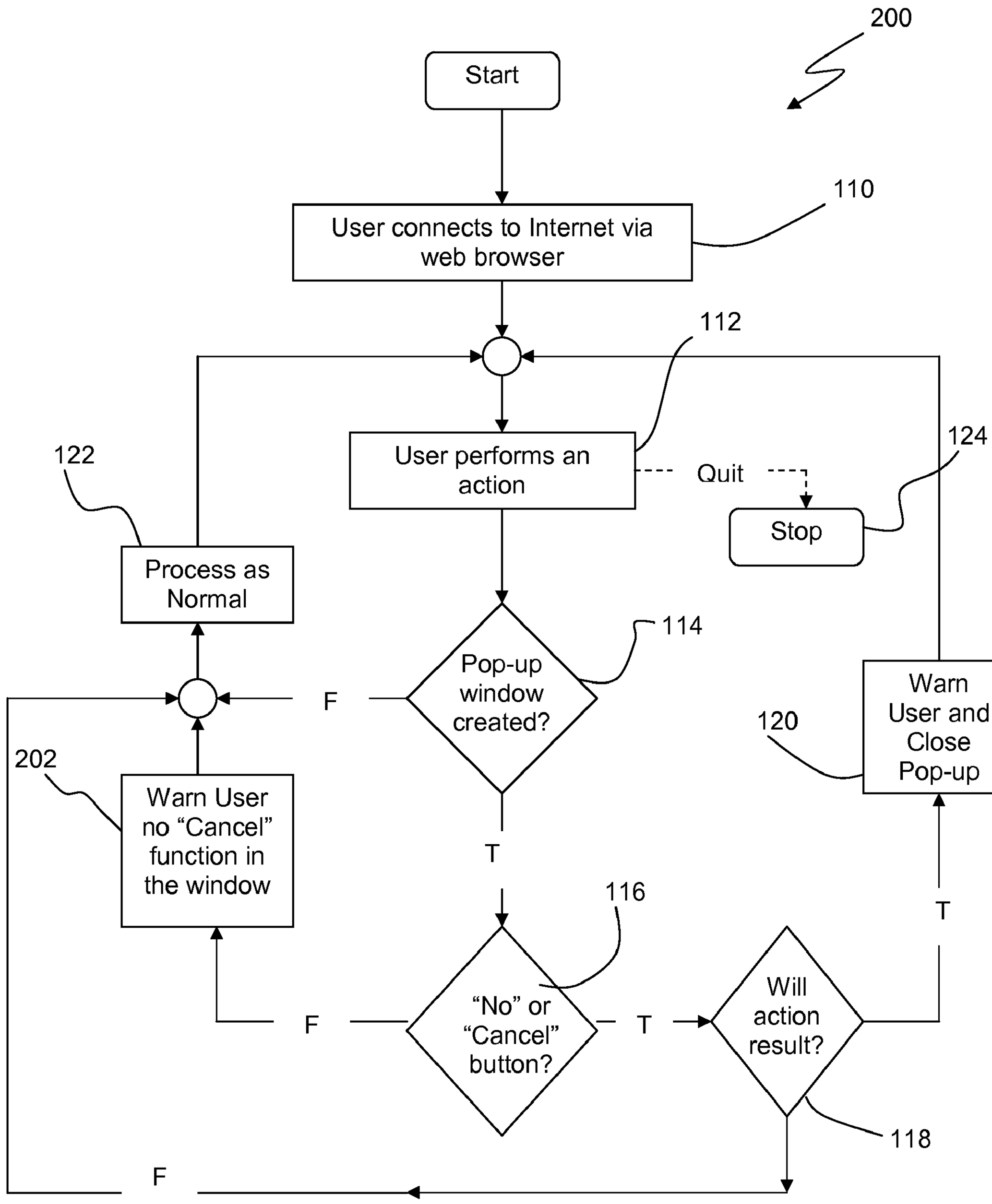

200
Start
User connects to Internet via web browser
110
112
User performs an action
Quit
124
Stop
122
Process as Normal
Pop-up window created?
114
F
T
202
Warn User no "Cancel" function in the window
120
Warn User and Close Pop-up
116
"No" or "Cancel" button?
Will action result?
F
T
T
118
F

FIG. 2

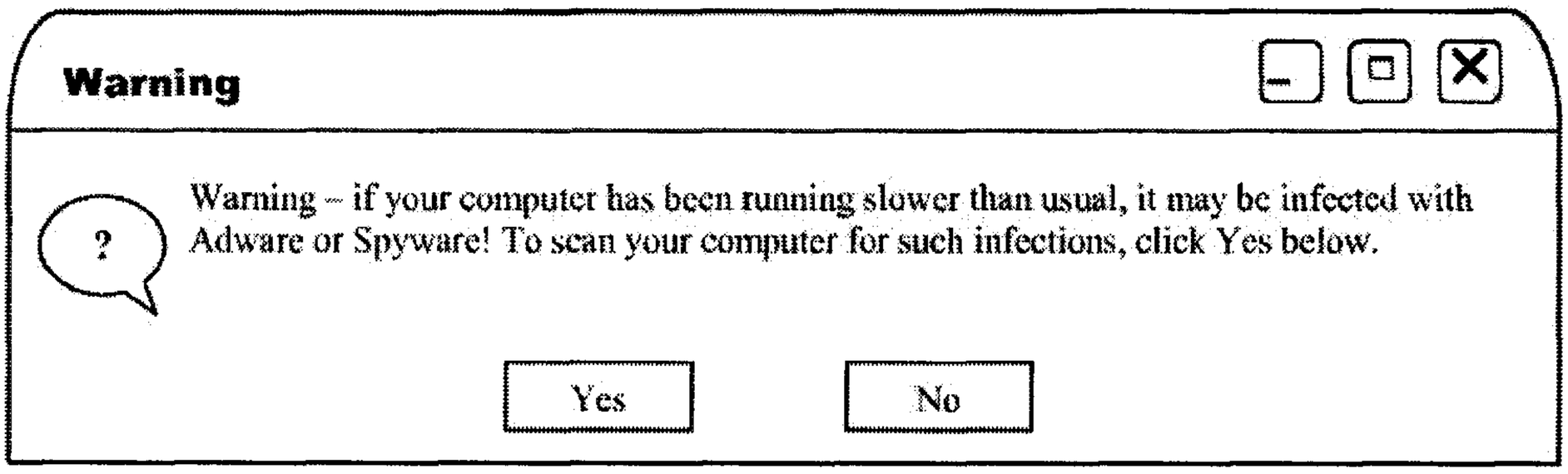


FIG. 3 (Prior Art)

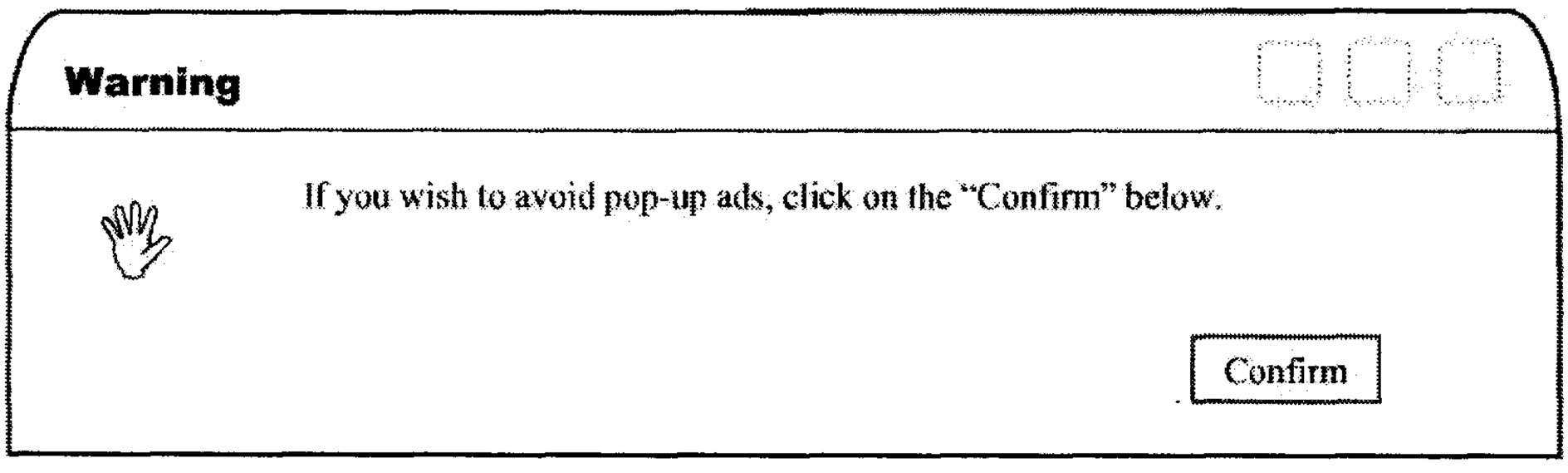


FIG. 4 (Prior Art)

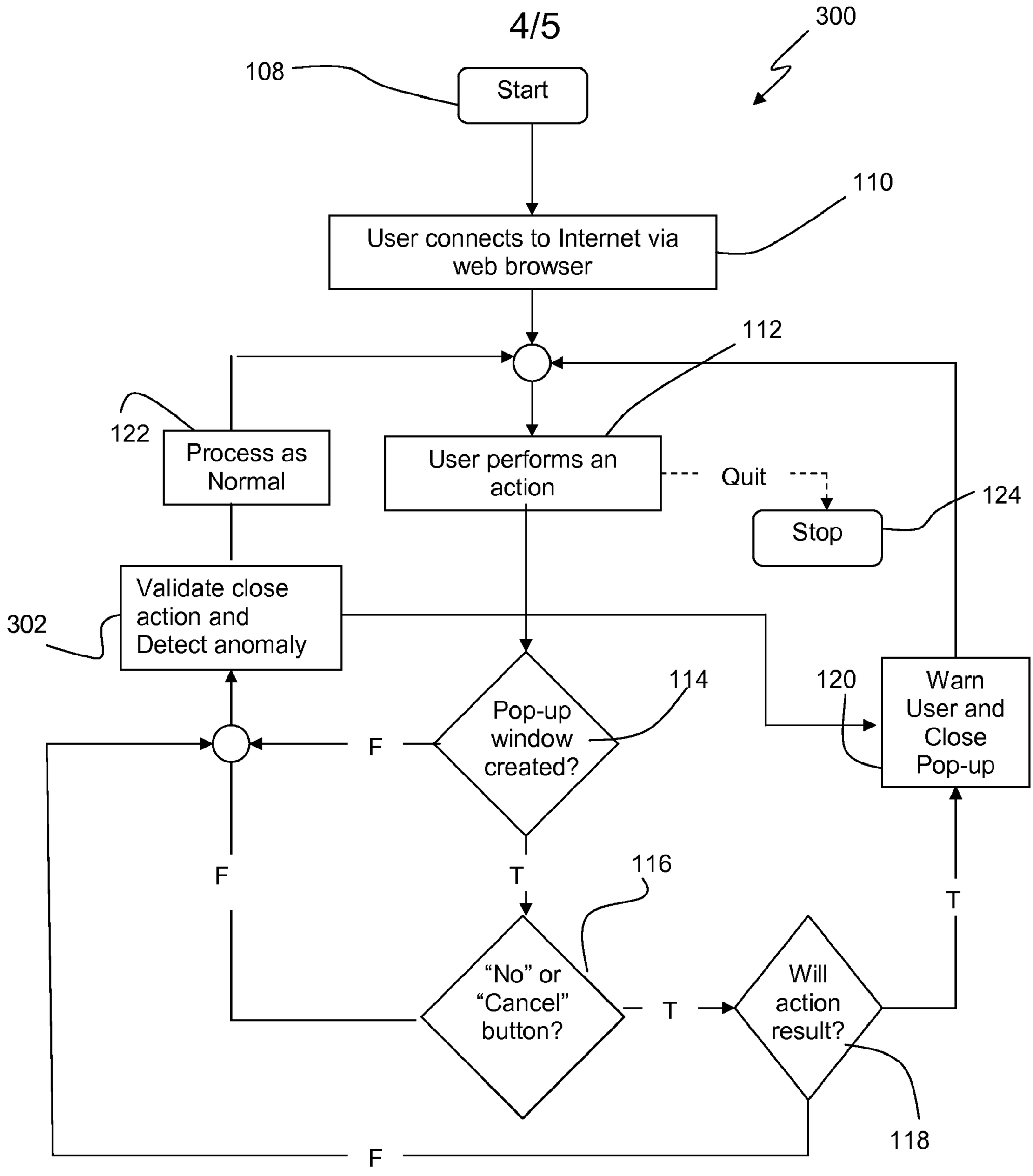

300
108
Start
110
User connects to Internet via web browser
112
122
Process as Normal
User performs an action
Quit
124
Stop
Validate close action and Detect anomaly
302
Pop-up window created?
114
120
Warn User and Close Pop-up
F
F
T
116
"No" or "Cancel" button?
T
Will action result?
T
118
F

FIG. 5

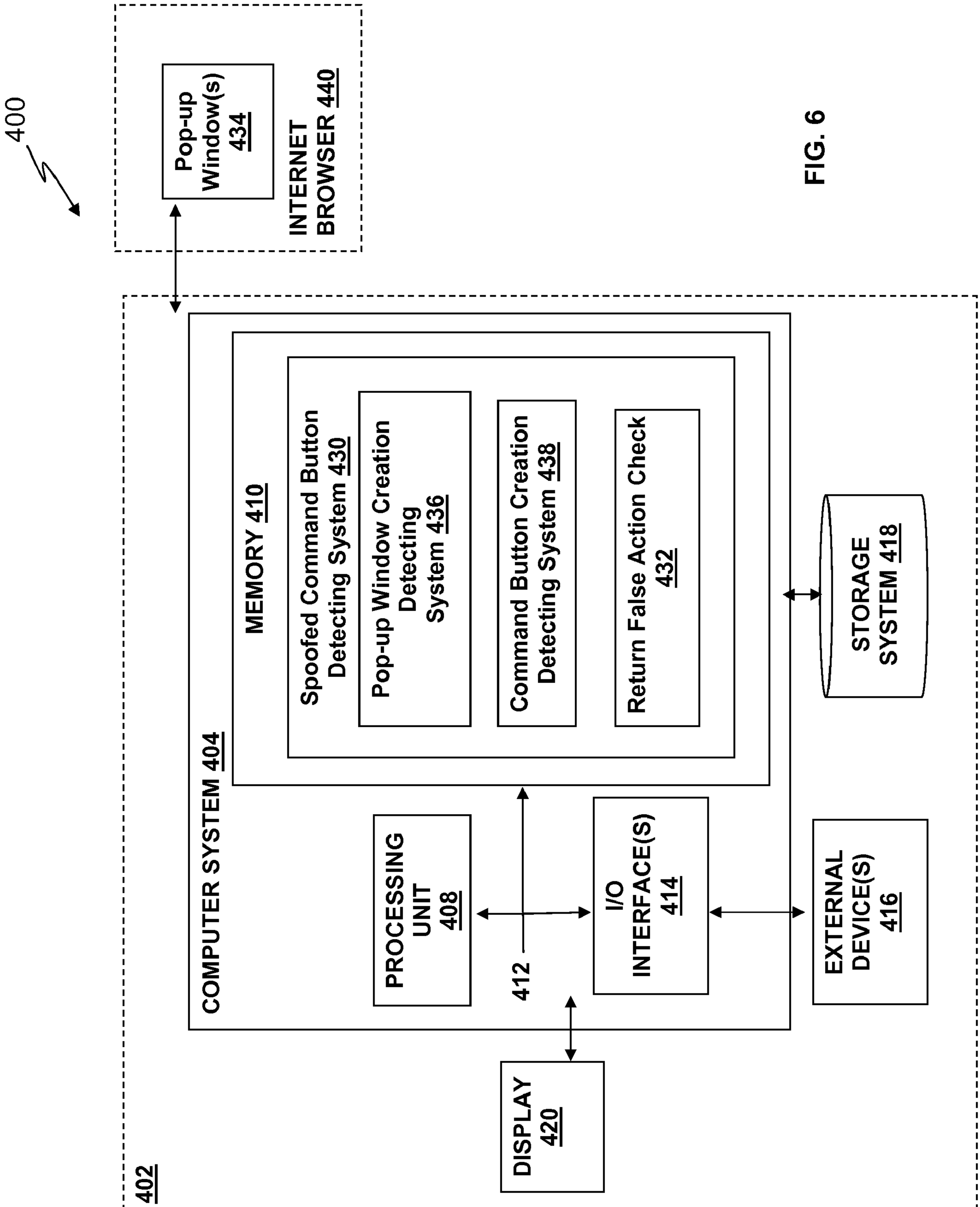
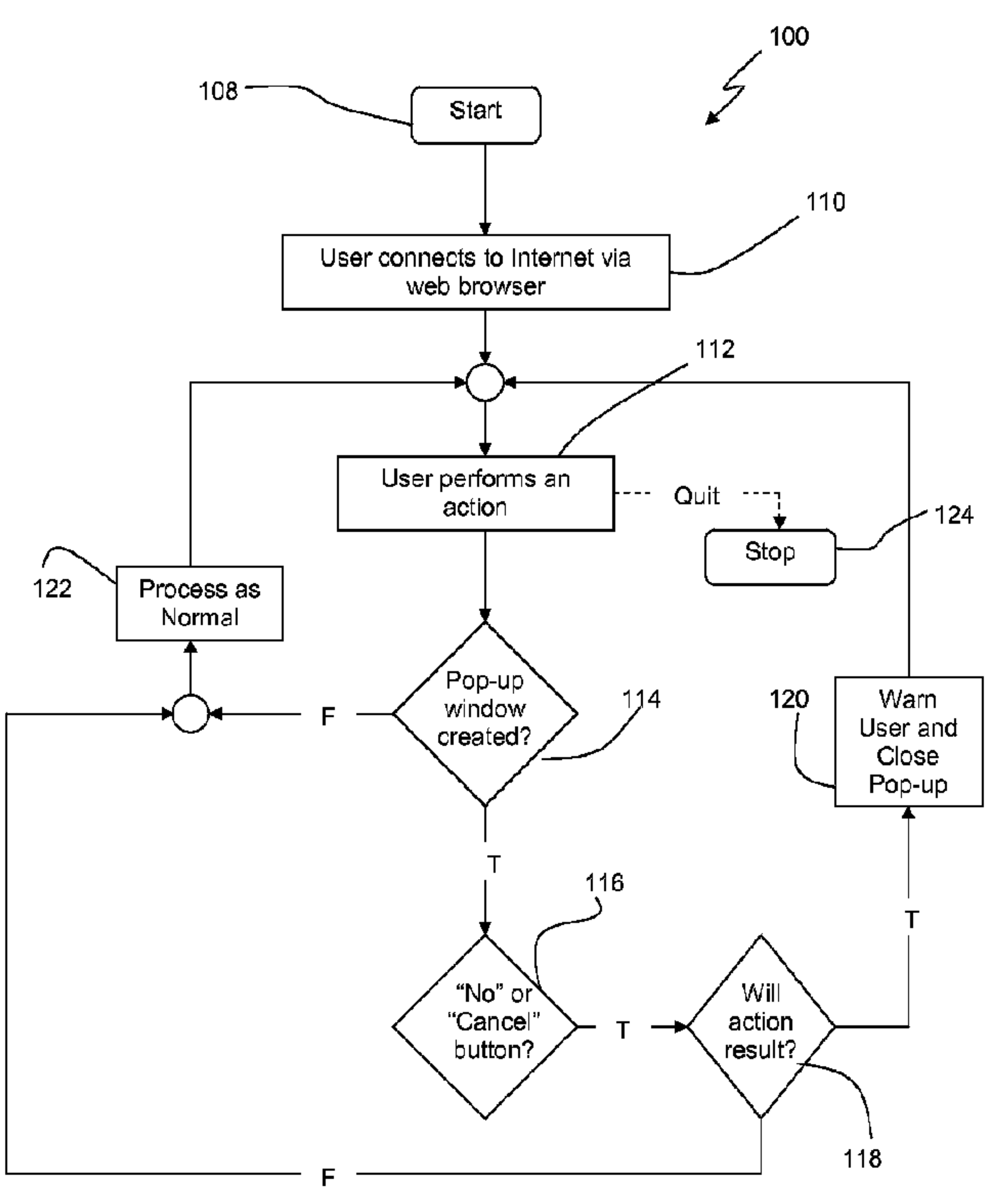

400
INTERNET BROWSER 440
Pop-up Window(s) 434
MEMORY 410
Spoofed Command Button Detecting System 438
Pop-up Window Creation Detecting System 436
Command Button Creation Detecting System 438
Return False Action Check 432
PROCESSING UNIT 408
412
I/O INTERFACE(S) 414
EXTERNAL DEVICE(S) 416
STORAGE SYSTEM 418
DISPLAY 420
COMPUTER SYSTEM 404
402

FIG. 6


100
108
Start
110
User connects to Internet via web browser
112
User performs an action
Quit
124
Stop
122
Process as Normal
Pop-up window created?
114
F
T
116
"No" or "Cancel" button?
T
Will action result?
T
120
Warn User and Close Pop-up
118
F