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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
5 December 2013 (05.12.2013)

WIPO | PCT

(10) International Publication Number
WO 2013/181226 A2

(51) International Patent Classification:
*G06F 21/31* (2013.01)

(21) International Application Number:
PCT/US2013/043070

(22) International Filing Date:
29 May 2013 (29.05.2013)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
61/654,047 31 May 2012 (31.05.2012) US

(72) Inventor; and
(71) Applicant : **MALVIN, Michael** [US/US]; 178 Desert Lakes Drive, Rancho Mirage, California 92270 (US).

(74) Agent: **SAYRE, Robert**; Modern Times Legal, 1 Broadway, 14th Floor, Cambridge, Massachusetts 02142 (US).

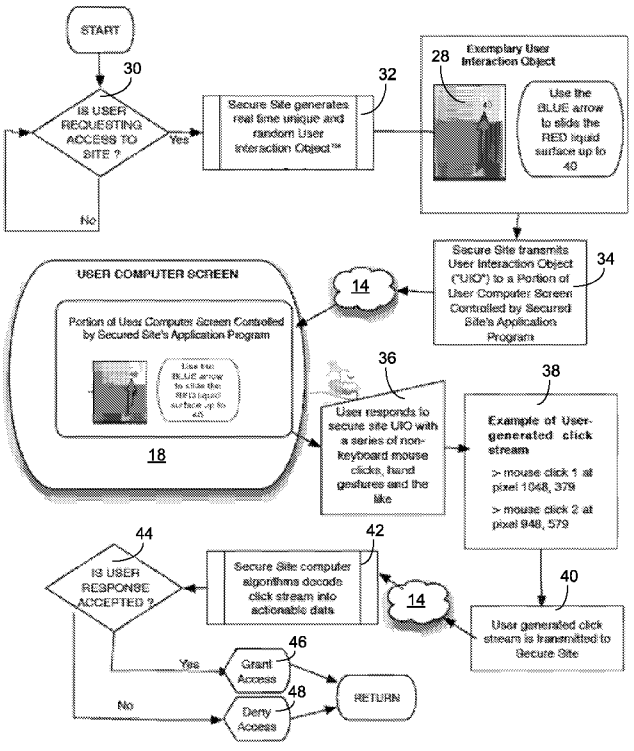
(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, 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, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

*[Continued on next page]*

(54) Title: DYNAMIC DATA ENTRY WITH ENHANCED SECURITY FOR SCREEN-BASED COMPUTING DEVICES

FIG. 4




(57) Abstract: The security of information entered into a screen-based computing device can be enhanced when a user provides information to a secure site by means of the user making non-keyboard screen-based responses (e.g., via mouse click, touch screen or other means) on one or more user interaction objects that are dynamically designed and randomly presented to the user by the secure site. The user interaction objects invite physical actions by the user on a non-keyboard device that generate, e.g., only a "click-stream" on the local computing device. This click stream data would be useless to an intruder (e.g., person or bot) unless the intruder knew what objects, actions and the like the mouse clicks (or other non-keyboard input) represent. The user interaction objects can be created randomly, in real time, allowing only the secure site to interpret and act on such data.

WO 2013/181226 A2

**Published**:

— *without international search report and to be republished upon receipt of that report (Rule 48.2(g))*

# DYNAMIC DATA ENTRY WITH ENHANCED SECURITY FOR SCREEN-BASED COMPUTING DEVICES

## BACKGROUND

The ability to securely enter sensitive data into a plurality of computing devices, including desktop computers, laptops, tablets, smart phones and other devices is foundational to the smooth functioning of the modern economy. Confidential data, such as banking user names and passwords and credit card data, are entered into computing devices millions of times per day. The theft or other misuse of such information is a continuing problem under the current art. Similar issues exist pertaining to corporate and government data. Further, there is a need to protect websites and web applications from attacks by bots (*i.e.*, automated software that pretends to be a bona fide user), thereby wasting or expropriating resources, for example, by registering for thousands of free email accounts.

In a conventional method for passcode entry, as schematically illustrated in FIG. 1, a secure site 12 (*e.g.*, a site operated by a bank, government agency, financial institution or merchant) sends a request 13 for a passcode (*e.g.*, from a central server) via the internet 11 to a client device 16 operated by a user, where the request 13 is presented on the display screen 18 of the client device 16. The user then enters 17 a passcode via a keyboard to gain access to additional content, and the passcode is transmitted back to the secure site 12 for verification. At the client device 16, an embedded malware program can capture 19 the passcode by logging the user's key entry. No matter how many levels of "site keys" and the like, keystroke-logger malware can capture 19 the passcode if it is entered 17 as keystrokes in a known or determinable screen location and then transmitted to the secure site 12.

In a conventional Completely Automated Public Turing Test to tell Computers and Humans Apart (CAPTCHA), as shown in FIG. 2, a secure site 12 sends 21 a CAPTCHA image 23 to a client device 16 operated by a user, where the image is presented on the user's screen 18 and must be deciphered by the user and entered 25 by typing characters on a keyboard into an interactive data-entry field 27 on the screen 18. The current art pertaining to the use of CAPTCHA and similar tests often have numerous drawbacks including (a) placing the burden on the user to decipher difficult-to-read characters and (b) rapid evolution of intruder technology to break existing and new tests.

SUMMARY

Methods and systems for data entry using screen-based computing devices are described herein. Various embodiments of the methods and systems may include some or all of the elements, features and steps described below. These methods and systems can improve the security of user-entered information, as it pertains to the foregoing examples described in the background, and more generally.

According to a method for soliciting data entry on a screen-based computing device, instructions are communicated for generating a graphic user interface on a display screen of a computing device and positioning at least one user interaction object at a first variable position within the graphic user interface. A user is allowed to make an on-screen selection within the graphic user interface using an input device (*e.g.*, a mouse or a touch screen) without using a keyboard (real or virtual) at a position with a defined relation to at least one user interaction object. Data is received that indicates the position of the selection by the user on the display screen; the position of the user's selection is compared with the position of the user interaction object on the display screen; and a determination is made as to whether the selected position is within an activating region of the user interaction object. If the defined position of the selection is within the activating region of at least one user interaction object, a protected user interface is presented to provide information or allow for further input. If the selection is not within the activating region, the user is restricted from accessing the protected user interface. The above steps are then repeated, except the user interaction object is positioned at a second variable position distinct from the first variable position.

The method can be carried out by the execution of computer-readable instructions for performing each of these steps, wherein the instructions are stored in a non-transitory manner on a computer-readable medium and communicated to a computer processor for execution.

The methods and systems can accordingly enhance the security of information entered into screen-based computing devices. Instead of typing letters or numbers into a "response box" sent by a secure site, (*e.g.*, a bank-operated server/website), the secure site can send a dynamically determined user interaction object (in real time) to the user's screen for the user to interact with. As explained below, this process can render keystroke logger programs and Captcha bots useless with respect to such information.

Information to be provided by a user to a requesting source program, application or the like, can be provided by the user by means of the user making screen-based responses (via mouse click, mouse drag, hand swipe across a touch screen or other interaction via other non-keyboard means) on or in response to one or more user interaction objects, which can be dynamically designed and randomly presented to the user for non-keyboard action by the requesting source. Accordingly, the information provided by the user cannot be as readily tracked or captured by unauthorized parties in comparison with the risk of keystroke logging malware where a keyboard (real or virtual—and with fixed key placements) is used for character entry, as in the prior art. The methods and systems can also provide an improved CAPTCHA response mechanism for authenticating that the user is a human being and allowing access to a site.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a schematic illustration of an existing methodology for passcode entry.

FIG. 2 is a schematic illustration of an existing methodology for CAPTCA character recognition and entry.

FIG. 3 is a block diagram of a screen-based computer machine that can be used to solicit data entry according to the methods described herein.

FIG. 4 is a flow chart illustrating an embodiment of an overall logic sequence used to process a secure data request in accordance with methods described herein.

FIG. 5 is a flow chart illustrating a detailed embodiment of a logic sequence used by a secure site when a user requests access to the secure site.

FIG. 6 is a flow chart illustrating a detailed embodiment of a logic sequence used with the user's response to a request for authentication from the secure site.

FIGS. 7-9 illustrate representative embodiments of a user interaction object that can be used in the methods described herein.

In the accompanying drawings, like reference characters refer to the same or similar parts throughout the different views; and apostrophes are used to differentiate multiple instances of the same or similar items sharing the same reference numeral. The drawings are not necessarily to scale, emphasis instead being placed upon illustrating particular principles, discussed below.

DETAILED DESCRIPTION

The foregoing and other features and advantages of various aspects of the invention(s) will be apparent from the following, more-particular description of various concepts and specific embodiments within the broader bounds of the invention(s). Various aspects of the subject matter introduced above and discussed in greater detail below may be implemented in any of numerous ways, as the subject matter is not limited to any particular manner of implementation. Examples of specific implementations and applications are provided primarily for illustrative purposes.

Unless otherwise defined, used or characterized herein, terms that are used herein (including technical and scientific terms) are to be interpreted as having a meaning that is consistent with their accepted meaning in the context of the relevant art and are not to be interpreted in an idealized or overly formal sense unless expressly so defined herein. For example if a particular shape is referenced, the shape is intended to include imperfect variations from ideal shapes.

Although the terms, first, second, third, *etc.*, may be used herein to describe various elements, these elements are not to be limited by these terms. These terms are simply used to distinguish one element from another. Thus, a first element, discussed below, could be termed a second element without departing from the teachings of the exemplary embodiments.

Spatially relative terms, such as “above,” “below,” “left,” “right,” “in front,” “behind,” and the like, may be used herein for ease of description to describe the relationship of one element to another element, as illustrated in the figures. It will be understood that the spatially relative terms, as well as the illustrated configurations, are intended to encompass different orientations of the apparatus in use or operation in addition to the orientations described herein and depicted in the figures. For example, if the apparatus in the figures is turned over, elements described as “below” or “beneath” other elements or features would then be oriented “above” the other elements or features. Thus, the exemplary term, “above,” may encompass both an orientation of above and below. The apparatus may be otherwise oriented (*e.g.*, rotated 90 degrees or at other orientations) and the spatially relative descriptors used herein interpreted accordingly.

Further still, in this disclosure, when an element is referred to as being "on," "connected to" or "coupled to" another element, it may be directly on, connected or coupled to the other element or intervening elements may be present unless otherwise specified.

The terminology used herein is for the purpose of describing particular embodiments and is not intended to be limiting of exemplary embodiments. As used herein, singular forms, such as "a" and "an," are intended to include the plural forms as well, unless the context indicates otherwise. Additionally, the terms, "includes," "including," "comprises" and "comprising," specify the presence of the stated elements or steps but do not preclude the presence or addition of one or more other elements or steps.

Basic components of a system for performing the methods of the invention are schematically shown in FIG. 3. Instructions for positioning and displaying one or more user interaction objects at random locations on a plurality of screen-based computing devices 16 are generated by a computer processor 24 executing software instructions stored on memory 22 (*e.g.*, a computer hard drive) and communicated through a network interface 26 across a network 14 (*e.g.*, the internet) to a plurality of screen-based computing devices 16 (wherein the user interaction object 28 is displayed at a unique location on each).

In a method of data entry, as shown in FIGS. 3 and 4, a secure site 12 (*e.g.*, any internet location, platform, program, application and the like where access is restricted and not open to the general public) creates a real-time unique user interaction object 28. A determination 30 is first made as to whether a user is requesting access to a protected user interface at the secure site 12. If so, the secure site 12, in real time, generates a unique and random user interaction object 28. In this embodiment, the user interaction object 28 includes a test tube filled with a colored liquid, the level of which can be adjusted by upward or downward movement of, across or in relation to an input device 20--*e.g.*, a mouse or a surface of a touch-screen display, or by hand or body gesture or voice, joystick or eye movement, though the user interaction objection 28 can be of nearly any design and nature, other than a keyboard. Examples of user interaction objects include images, numbers, letters, *etc.* The size, shape, color, orientation, and/or other features of the user interaction object 28 can be dynamically and randomly determined in real time by the secure site 12.

In one embodiment, a plurality of user interaction objects are provides in the form of numbers and/or letters (in an arrangement other than that of a conventional keyboard) that

can be selected by a user to form a passcode that can be evaluated by the secure site 12 to see if it matches the access code for access to a protected user interface at the secure site 12. In one example, the numbers/letters can be provided as a universal string that enable recognition of the user as a human being. In another example, the user interaction objects are numbers, and the numbers selected by the user are compared with a credit-card or financial-account number associated with the user, wherein the secure site 12 has access to the credit-card or financial-account number of the user and can compare it with the user's input to determine whether to grant access to the protected user interface (in the case of a match).

As shown in FIG. 4, the secure site 12 determines 30 whether a user is requesting access to the site 12. If so, the secure site 12 generates 32 real-time unique and random user interaction objects 28. The secure site 12 transmits 34 the user interaction object 28 (*e.g.*, via the network interface 26 across the internet 14) to each of a plurality of a computing devices 16 operated by respective users, where the user interaction object 28 is displayed on the user's display screen 18; the secure site 12 dictates the on-screen position and content of the user interaction object 28. The pixel coordinates for locating the user interaction object 28 may also be determined randomly in real time. The user can respond 36 to the displayed user interaction object 28 by via an entry, wherein the user clicks (taps) on (or at a particular location in relation to) and drags the liquid in the test tube in the displayed image. The taps can be made on the display screen 18, *e.g.*, by the user's finger tapping on the image of the user interaction object on a touch-sensitive display (where the screen's surface serves as an input device) or by tapping on a remote input device, such as a computer mouse, that positions a pointer (*e.g.*, a cursor) in the display screen 18.

In one example 38, the user makes two mouse clicks at distinct positions (*e.g.*, pixel 1048, 379 and pixel 948, 579), corresponding to sequential displays of the user interaction object 28 as it is displaced across the screen 18 (facilitating a dynamic interaction with the user), to different features within the user interaction object 28 or to a plurality of different user interaction objects 28 respectively displayed at distinct positions on the screen. The click stream generated by the user is transmitted 40 to the secure site 12. The secure site computer machine 12 decodes 42 the click stream via algorithms into actionable data and determines 44 whether the user's click stream matches the expected response. If so, access to a protected

user interface (*e.g.*, a secure webpage) at the secure site 12 is granted 46 to the user. If not, access to the secure site 12 is denied 48 to the user.

The computer software (described below) with instructions for performing this method can define an activating region, which is a defined area within the display in reference to the user interaction object 28 (*e.g.*, including and surrounding the user interaction object 28, wherein the activating region may be a limited area within the user interaction object 28. In other embodiments, the activating region may share the same boundary as the user interaction object 28 or may expand beyond the boundary of the user interaction object 28.

In the method of FIG. 5, a secure site receives 46 a request for access to the site, and the secure site generates 48 a series of random numbers. The secure site uses 50 the random numbers to randomly select a user interaction object 28 from a library of user interaction objects. The secure site then generates additional random numbers (via existing techniques for computer generation of "random" numbers) to dynamically define 52 the size, shape, orientation, required user actions, and the like for the randomly selected user interaction object and to randomly locate 54 the randomly generated user interaction object 28 on the user's screen, as shown by random locations A and B on the user's computer screen 18. The secure site then transmits 56 the user interaction object 28 to a portion of the user's computer screen 18 controlled by the secured site's application program.

In the method of FIG. 6, the user interaction object is received 34 from the secure site and displayed on the user's computer screen 18. Data is entered 36 by a user via, *e.g.*, mouse clicks on a dynamically generated and randomly located user interaction object. The information generated on the local screen-based computing device as result of the user's actions can be limited to a series, such as "mouse click at pixel 1048, 379" and "mouse click at pixel 956,379", "mouse click at pixel 956, 271", *etc.* That click-stream information is transmitted 40 to the secure-site server, where computer code is executed to decode the click stream into actionable data (*e.g.*, a matching of user clicks with elements in the user interaction object). If this information is captured by an intruder, this click-stream information will generally be useless to the intruder (person or bot) unless the intruder knows what objects, actions and the like, the mouse clicks were activating. Since the user interaction object can be different each time and because the user interaction object is located at randomly selected pixel coordinates, only the secure site can interpret and act upon such information.

The secure-site server 12 can then accept or reject that data based on whether it passes authorization, and the resulting response can then be transmitted to the user via the display of the client device.

The user interaction object(s) 28 may be of any design or nature and can be randomly generated, dynamic and different each time a user interaction object 28 is presented to a user. These characteristics can make it virtually impossible for an intruder to "break" (*i.e.*, decode into decipherable/actionable data) the user interaction object 28 because the user interaction object 28 can be randomly and uniquely selected and positioned with each iteration of the presentation of a user interaction object 28. Consequently, there may be no static "target" for the intruder's computer program to study and decipher.

Several other examples of suitable user interaction objects 28 that can be used in these methods are presented in FIGS. 7-9. In the examples of FIGS. 3-6, a test tube cylinder is displayed as the user interaction object 28; and the user is instructed (via instructions displayed on-screen) to slide the top liquid surface up to the 40 (*e.g.*, percent or mL) level on the test tube. In FIG. 7, various patterns of inner-nested white and black circles are provided; and the user may be instructed, for example, to click on the solid black circles. In FIG. 8, a yellow cylinder 58 is displayed amongst a sea of black cylinders, and the user may be instructed via instructions presented on the display to click on the yellow cylinder 58. In FIG. 9, an animatronic user interface object 28 including a swinging pendulum is provided, and the user may be instructed via instructions 60 to click on the pendulum at a particular point in its reciprocating arc.

The following descriptions characterize various hardware, software, and communication protocols that can used with this invention. For example, the hardware, software and communication protocols, described below, can be used in/as the secure site server 12 and in/as the client devices 16 to carry out the methods described above.

**Computers, software, storage media, and other components**

The systems and methods of this disclosure can be implemented in a computing system environment, as shown in FIG. 3. Examples of well-known computing system environments and components thereof that may be suitable for use with the systems and methods include, but are not limited to, personal computers, server computers, hand-held or laptop devices, tablet devices, smart phones, multiprocessor systems, microprocessor-based systems, set top boxes,

programmable consumer electronics, network PCs, minicomputers, mainframe computers, distributed computing environments that include any of the above systems or devices, and the like. Typical computing system environments and their operations and components are described in many existing patents (*e.g.*, US Patent No. 7,191,467, owned by Microsoft Corp.).

The methods may be carried out via non-transitory computer-executable instructions, such as program modules, wherein the programs are read from a computer-readable medium and executed by a computer processor. Generally, program modules include routines, programs, objects, components, data structures, and so forth, that perform particular tasks or implement particular types of data. The methods may also be practiced in distributed computing environments, where tasks are performed by remote processing devices that are linked through a communications network. In a distributed computing environment, program modules may be located in both local and remote computer storage media including memory storage devices.

The systems (*e.g.*, of the "client" and "server") and methods of this disclosure may utilize a computer machine to carry out the processes described herein. The described functions can be stored in the form of software instructions on the client or the server, communicated therebetween, and carried out at the client or at the server. Components of the computer machine may include, but are not limited to, a computer processor, a computer storage medium serving as memory, and a system bus that couples various system components including the memory to the computer processor. The system bus can be of any of several types of bus structures including a memory bus or memory controller, a peripheral bus, and a local bus using any of a variety of bus architectures.

The computer machine typically includes a variety of computer-readable media accessible by the processor and including both volatile and nonvolatile media and removable and non-removable media. By way of example, computer-readable media can comprise computer-storage media and communication media.

The computer storage media can store the software and data in a non-transitory state and includes both volatile and nonvolatile, removable and non-removable media implemented in any method or technology for storage of software and data, such as computer-readable instructions, data structures, program modules or other data. Computer-storage media includes, but is not limited to, RAM, ROM, EEPROM, flash memory or other memory

technology, CD-ROM, digital versatile disks (DVD) or other optical disk storage, magnetic cassettes, magnetic tape, magnetic disk storage or other magnetic storage devices, or any other medium that can be used to store the desired information and that can accessed by the computer processor.

The memory includes computer-storage media in the form of volatile and/or nonvolatile memory such as read only memory (ROM) and random access memory (RAM). A basic input/output system (BIOS), containing the basic routines that help to transfer information between elements within the computer machine, such as during start-up, is typically stored in the ROM. The RAM typically contains data and/or program modules that are immediately accessible to and/or presently being operated on by the processor.

The computer machine may also include other removable/non-removable, volatile/nonvolatile computer-storage media, such as (a) a hard disk drive that reads from or writes to non-removable, nonvolatile magnetic media; (b) a magnetic disk drive that reads from or writes to a removable, nonvolatile magnetic disk; and (c) an optical disk drive that reads from or writes to a removable, nonvolatile optical disk such as a CD ROM or other optical medium. The computer-storage medium can be coupled with the system bus by a communication interface, wherein the interface can include, *e.g.*, electrically conductive wires and/or fiber-optic pathways for transmitting digital or optical signals between components. Other removable/non-removable, volatile/nonvolatile computer storage media that can be used in the exemplary operating environment include magnetic tape cassettes, flash memory cards, digital versatile disks, digital video tape, solid state RAM, solid state ROM, and the like.

The drives and their associated computer-storage media provide storage of computer-readable instructions, data structures, program modules and other data for the computer machine. For example, a hard disk drive inside or external to the housing of the computer machine can store an operating system, application programs, and program data.

The various processes described in the descriptions that follow can be encoded as software instructions in memory and executed by a processor to carry out the processes.

Conventionally, a keyboard is used to enter information into a computing device. According to embodiments of the methods and systems of this disclosure, however, information can be entered without the use of a keyboard (real or virtual). Rather, a user of the computing device can enter commands and information into the computer machine through

an input device, such as a pointing device (such as a mouse, trackball or touch-pad tablet); a touch-screen device, or a motion-detection device (*e.g.*, a Wii game controller, as provided by Nintendo). The input device can be connected to the processor through a communication interface coupled to the system bus, by another interface and bus structure, such as a parallel port or a universal serial bus (USB), or wirelessly via conventional wireless transmitters and receivers and communication protocols, such as Bluetooth or IEEE 802.11 wireless standards.

A monitor or other type of display device can also be connected to the system bus via a communication interface (*e.g.,* a video interface). The monitor can also be integrated with a touch-screen panel or the like that can input digitized input, such as handwriting, into the computer system via a communication interface. In some embodiments, the monitor and/or touch screen panel can be physically coupled to or incorporated into a housing of the computer machine, such as in a tablet-type personal computer. In addition, the computer machine can also include other peripheral output devices, such as one or more speakers and printers, through a communication interface.

**Network connections and communications**

The computer machine(s) can operate in a networked environment using logical connections from a server to one or more remote client computer machines, as shown in FIG. 3, to perform the methods described herein. The remote computer machine can be, *e.g.*, a personal computer, a server, a router, a network PC, a peer device or other common network node, and typically includes many or all of the elements described relative to the above-described computer machine. The networked environment over which communication can be transmitted between the secure site and the client can include a local area network (LAN), a wide area network (WAN), and/or other networks. Such networking environments are used in offices, enterprise-wide computer networks, intranets and the internet.

When used in a LAN networking environment, the computer machine is connected to the LAN through a network interface or adapter. When used in a WAN networking environment, the computer machine typically includes a modem for establishing communications over the WAN (*e.g.*, over the internet). The modem, which can be internal or external to the computer housing, can be connected to the system bus via the user-input interface or via another appropriate mechanism.

In an embodiment of a WAN environment, a user (via a client computer machine connected with and in communication with the internet) accesses one or more remote computer servers (here, the secure site), also connected with and in communication with the internet, using, *e.g.*, an internet browser (such as Internet Explorer from Microsoft, Firefox from Mozilla, or Chrome from Google) via hypertext transfer protocol (HTTP) communications or via communications generated and/or received by a software program, such as an email application (*e.g.*, Microsoft Outlook) that can be stored in the computer's memory. The computer server can be a computer machine including memory storing a web server application, such as the Apache HTTP Server. The client computer machine can send an HTTP GET request to the server via the communication media that form the internet, and the participating server can respond to the client computer machine via the internet with an appropriate HTTP response.

HTTP is a request-response protocol standard for client-server computing. In HTTP, a personal computer machine running a web browser, for example, acts as a client, while a computer machine hosting a web site acts as a server. The client submits HTTP requests to the responding server by sending messages to it. The server, which stores content (or resources), such as HTML files and images, or generates such content on the fly, sends messages back to the client in response. These returned messages may contain the content requested by the client or may contain other kinds of response indications. Between the client and server there may be several intermediaries, such as proxies, web caches or gateways. In such a case, the client communicates with the server indirectly, and only converses directly with the first intermediary in the chain.

An HTTP request message from the client can include the following: (a) a Request line that requests a resource (such as an image); (b) Headers; (c) an empty line; and, optionally, (d) a message body. The HTTP Headers form the core of the HTTP request, as they define various characteristics of the data that is requested or the data that has been provided. The HTTP Headers can include a referrer that identifies, from the point of view of an internet webpage or resource, the address of the webpage (*e.g.*, the URL) of the resource that links to it. By checking the referrer, the new page can determine the source of the request message. A variety of different request protocols exists; for example, a “GET request” requests a representation of the specified resource from the host.

In describing embodiments of the invention, specific terminology is used for the sake of clarity. For the purpose of description, specific terms are intended to at least include technical and functional equivalents that operate in a similar manner to accomplish a similar result. Additionally, in some instances where a particular embodiment of the invention includes a plurality of system elements or method steps, those elements or steps may be replaced with a single element or step; likewise, a single element or step may be replaced with a plurality of elements or steps that serve the same purpose. Further, where parameters for various properties or other values are specified herein for embodiments of the invention, those parameters or values can be adjusted up or down by $1/100^{th}$, $1/50^{th}$, $1/20^{th}$, $1/10^{th}$, $1/5^{th}$, $1/3^{rd}$, 1/2, $2/3^{rd}$, $3/4^{th}$, $4/5^{th}$, $9/10^{th}$, $19/20^{th}$, $49/50^{th}$, $99/100^{th}$, *etc.* (or up by a factor of 1, 2, 3, 4, 5, 6, 8, 10, 20, 50, 100, *etc.*), or by rounded-off approximations thereof, unless otherwise specified. Moreover, while this invention has been shown and described with references to particular embodiments thereof, those skilled in the art will understand that various substitutions and alterations in form and details may be made therein without departing from the scope of the invention. Further still, other aspects, functions and advantages are also within the scope of the invention; and all embodiments of the invention need not necessarily achieve all of the advantages or possess all of the characteristics described above. Additionally, steps, elements and features discussed herein in connection with one embodiment can likewise be used in conjunction with other embodiments. The contents of references, including reference texts, journal articles, patents, patent applications, *etc.*, cited throughout the text are hereby incorporated by reference in their entirety; and appropriate components, steps, and characterizations from these references may or may not be included in embodiments of this invention. Still further, the components and steps identified in the Background section are integral to this disclosure and can be used in conjunction with or substituted for components and steps described elsewhere in the disclosure within the scope of the invention. In method claims, where stages are recited in a particular order—with or without sequenced prefacing characters added for ease of reference—the stages are not to be interpreted as being temporally limited to the order in which they are recited unless otherwise specified or implied by the terms and phrasing.

CLAIMS

What is claimed is:

1. A method for soliciting data entry on a screen-based computing device, the method comprising:

communicating instructions for generating a graphic user interface on a display screen of a computing device and positioning at least one user interaction object at a first variable position within the graphic user interface;

allowing a user to make an on-screen selection within the graphic user interface using an input device that is not a keyboard at a position with a defined relation to at least one user interaction object;

receiving data indicating the position of the selection by the user on the display screen;

comparing the defined relational position of the user selection with respect to the position of the user interaction object on the display screen and determining if the selected position is within an activating region of the user interaction object;

if the defined position of the selection is within the activating region of at least one user interaction object, presenting a protected user interface to the user providing information or allowing for further input;

if the selection is not within the activating region, restricting the user from accessing the protected user interface; and

repeating the above steps, except positioning the user interaction object at a second variable position distinct from the first variable position.

2. The method of claim 1, wherein the activating region is within the boundary of the user interaction object.

3. The method of claim 1, wherein the input device is selected from a touch-sensitive display, a touch pad, a mouse, a trackball and a joystick.

4. The method of claim 1, wherein the selection is made by manipulating an on-screen pointer via an input device.

5. The method of claim 1, wherein the display screen is touch-sensitive and the selection is made by the user's touch on the touch-sensitive display screen.

6. The method of claim 1, wherein the selection is made by a tapping motion on the input device from the user.

7. The method of claim 1, wherein the selection is made by user action selected from at least one of the following: vocalization, gesturing, and eye movement.

8. The method of claim 1, further comprising providing instructions for displacing the user interaction object across the display screen for dynamic interaction with the user.

9. The method of claim 1, wherein instructions are provided within the graphic user interface for positioning a plurality of user interaction objects.

10. The method of claim 9, further comprising receiving data indicating a plurality of user selections.

11. The method of claim 10, further comprising comparing the position of each of the user selections, with the positions of respective user interaction objects on the display screen and determining if the position of the selections is within the activating region of a respective user interaction object at the time of the selections.

12. The method of claim 1, wherein instructions are provided for generating a plurality of user interaction objects that include at least one of numbers and letters.

13. The method of claim 12, further comprising comparing the numbers or letters of a plurality of user interaction objects selected by the user with a string of numbers or letters previously selected by the user as a passcode.

14. The method of claim 12, further comprising comparing the numbers or letters of a plurality of user interaction objects selected by the user with a universal string of numbers or letters that enable recognition of the user as a human being.

15. The method of claim 12, wherein the user interaction object includes numbers and wherein the selected numbers are compared with a credit-card or financial-account number associated with the user.

16. The method of claim 1, further comprising generating a random number and using the random number to select a user interaction object from a library including a plurality of user interaction objects.

17. The method of claim 16, further comprising generating at least another random number and using the additional random number to define at least one of size, shape and orientation of the randomly selected user interaction object.

18. The method of claim 16, further comprising generating at least another random number to randomly locate the randomly generated user interaction on the user's screen.

19. A computer-readable storage medium having stored non-transitorally thereon a computer program for dynamic and secure data entry, the computer program comprising a routine set of instructions, which when executed by a computer machine, cause the computer machine to perform the following steps:

communicating instructions for generating a graphic user interface on a display screen of a computing device and positioning at least one user interaction object at a first variable position within the graphic user interface;

allowing a user to make an on-screen selection within the graphic user interface using an input device that is not a keyboard at a position with a defined relation to at least one user interaction object;

receiving data indicating the position of the selection on the display screen as manipulated by the user when the user indicates a selection;

comparing the defined relational position of the selection with respect to the position of the user interaction object on the display screen and determining if the selected position is within an activating region of the user interaction object;

if the defined position of the selection is within the activating region of at least one user interaction object, providing a protected user interface providing information or allowing for further input;

if the selection is not within the activating region, restricting the user from accessing the protected user interface; and

repeating the above steps, except positioning the user interaction object at a second variable position distinct from the first variable position.

20. The computer-readable storage medium of claim 19, wherein the steps further comprise displacing the user interaction object across the display screen for dynamic interaction with the user.

21. The computer-readable storage medium of claim 19, wherein the steps further comprise positioning a plurality of user interaction objects.

22. The computer-readable storage medium of claim 19, wherein the steps further comprise generating a random number and using the random number to select a user interaction object from a library including a plurality of user interaction objects.

23. The computer-readable storage medium of claim 19, wherein the steps further comprise generating at least another random number to randomly locate the randomly generated user interaction on the user's screen.

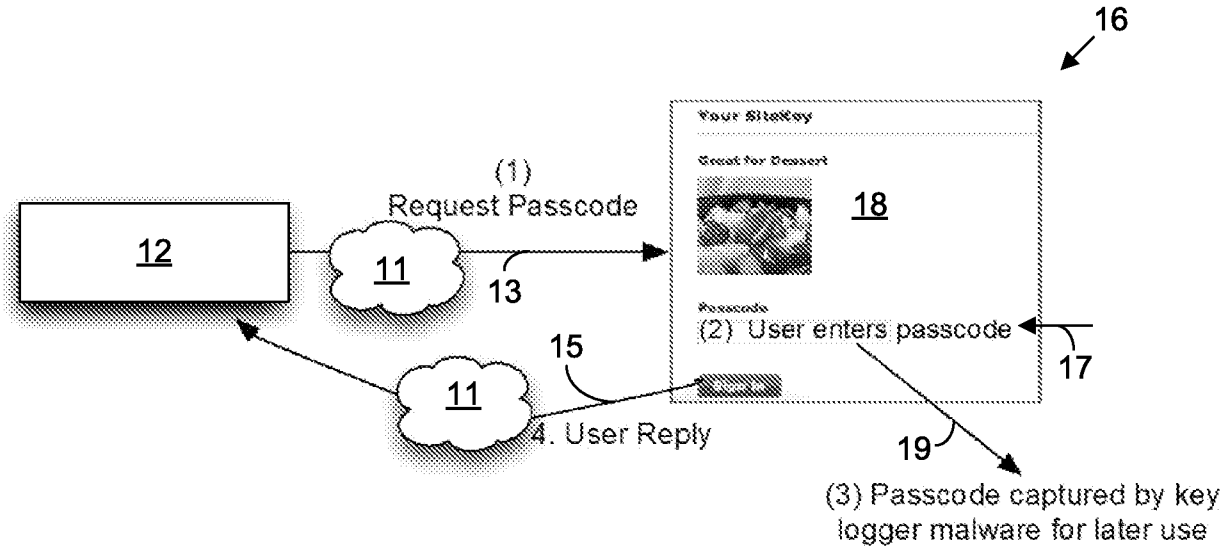

16
(1)
Request Passcode
12
11
13
18
(2) User enters passcode
17
15
11
4. User Reply
19
(3) Passcode captured by key logger malware for later use

FIG. 1

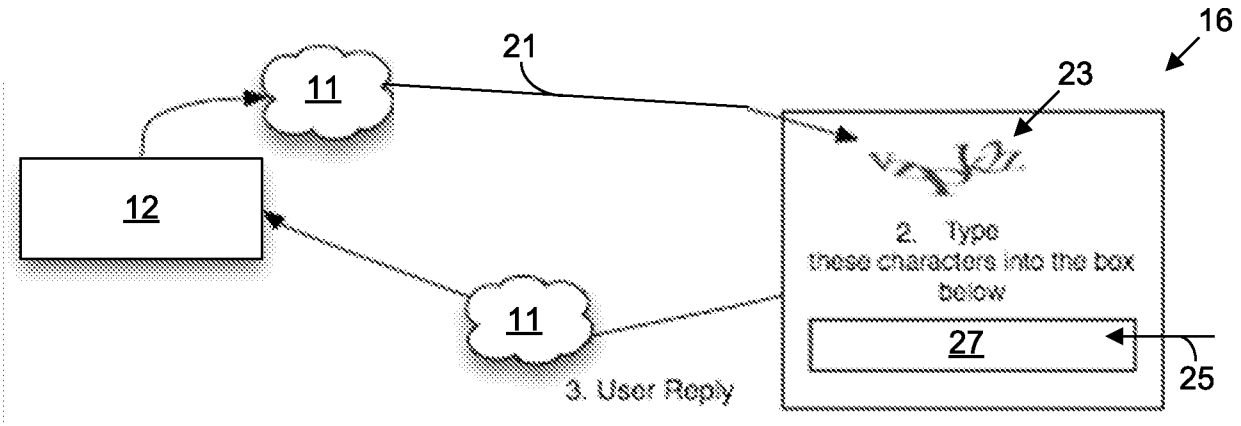

16
21
11
23
12
2. Type these characters into the box below
11
27
25
3. User Reply

FIG. 2

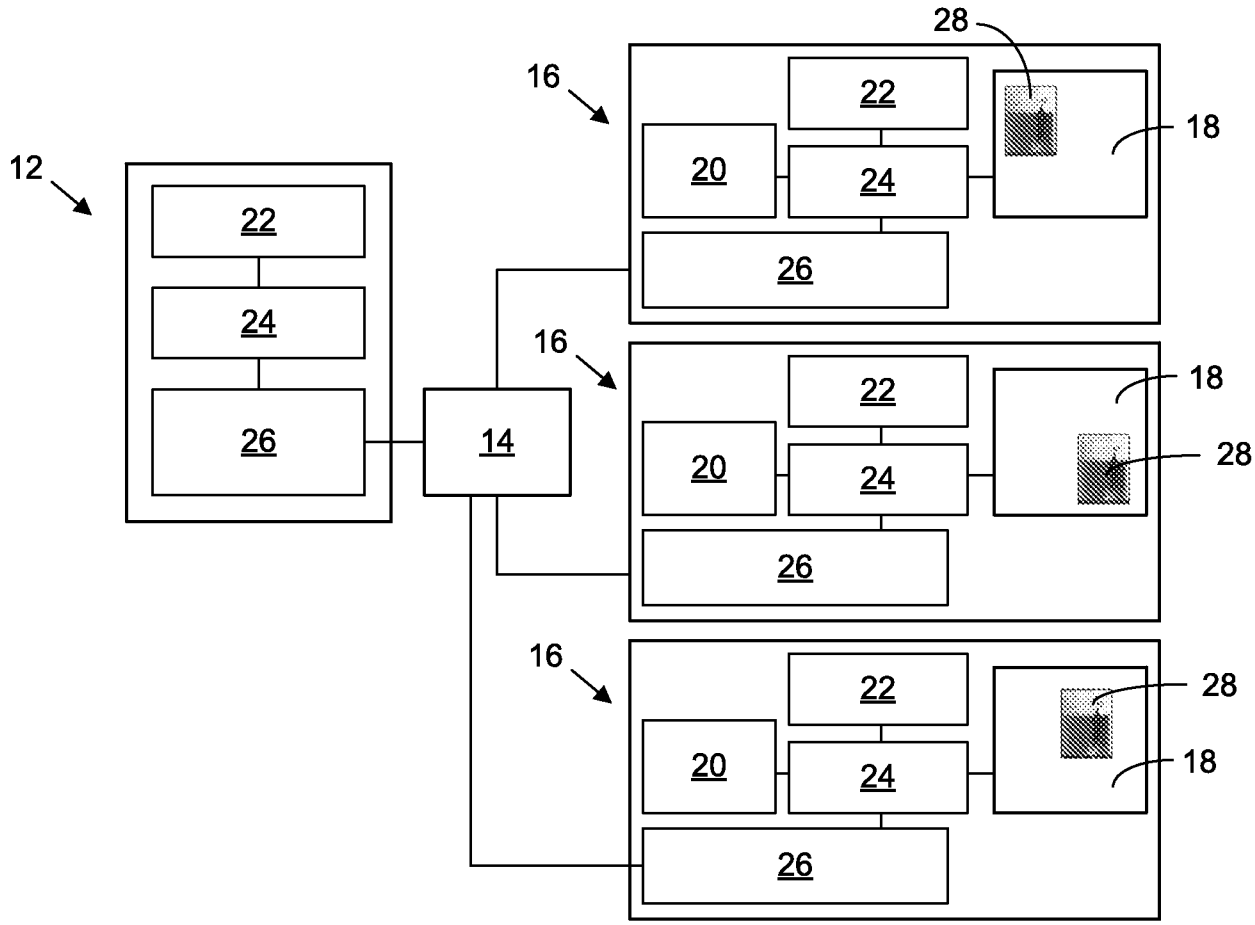

12
22
24
26
14
16
20
22
24
26
18
28
16
20
22
24
26
18
28
16
20
22
24
26
28
18

FIG. 3

FIG. 4

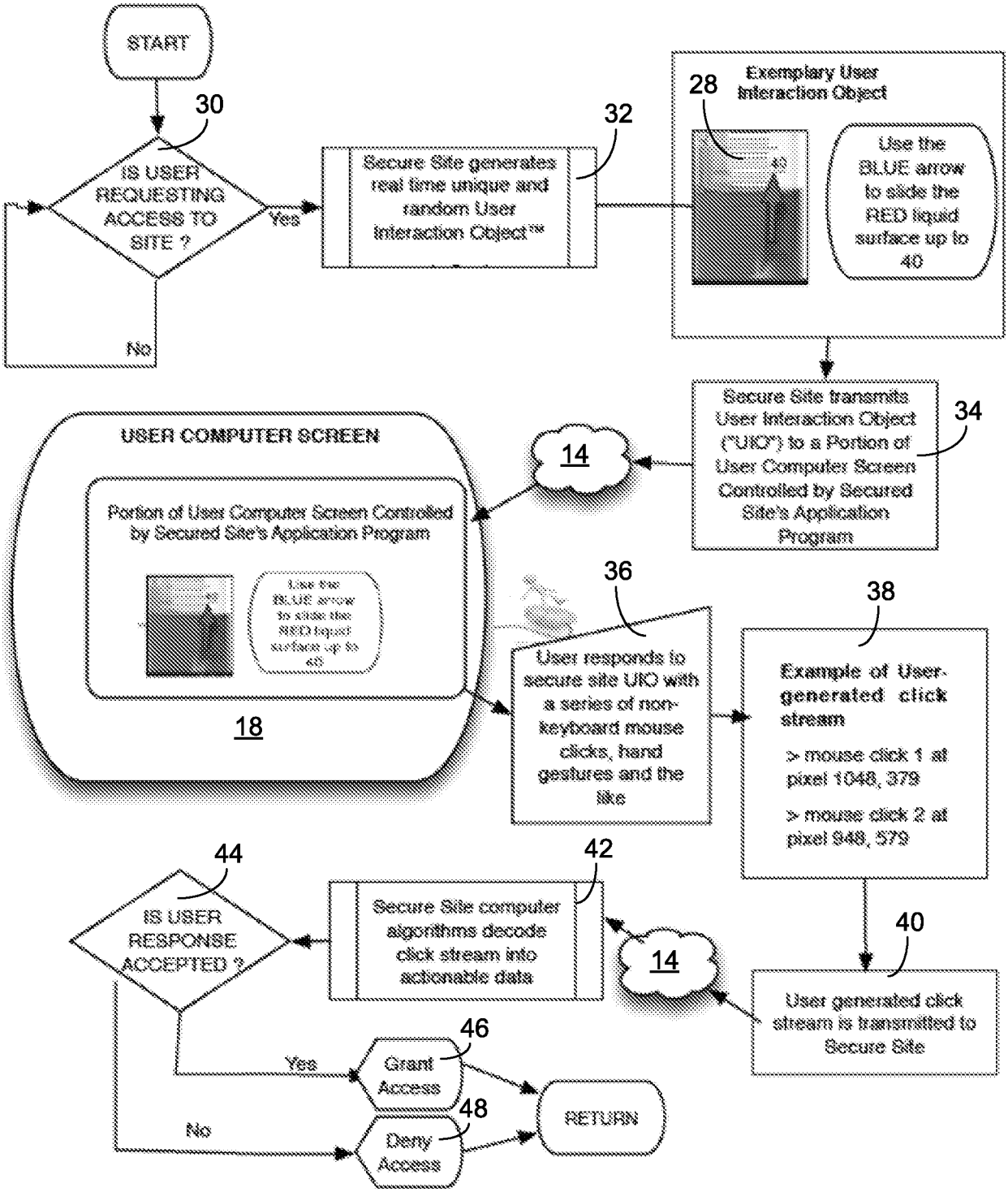
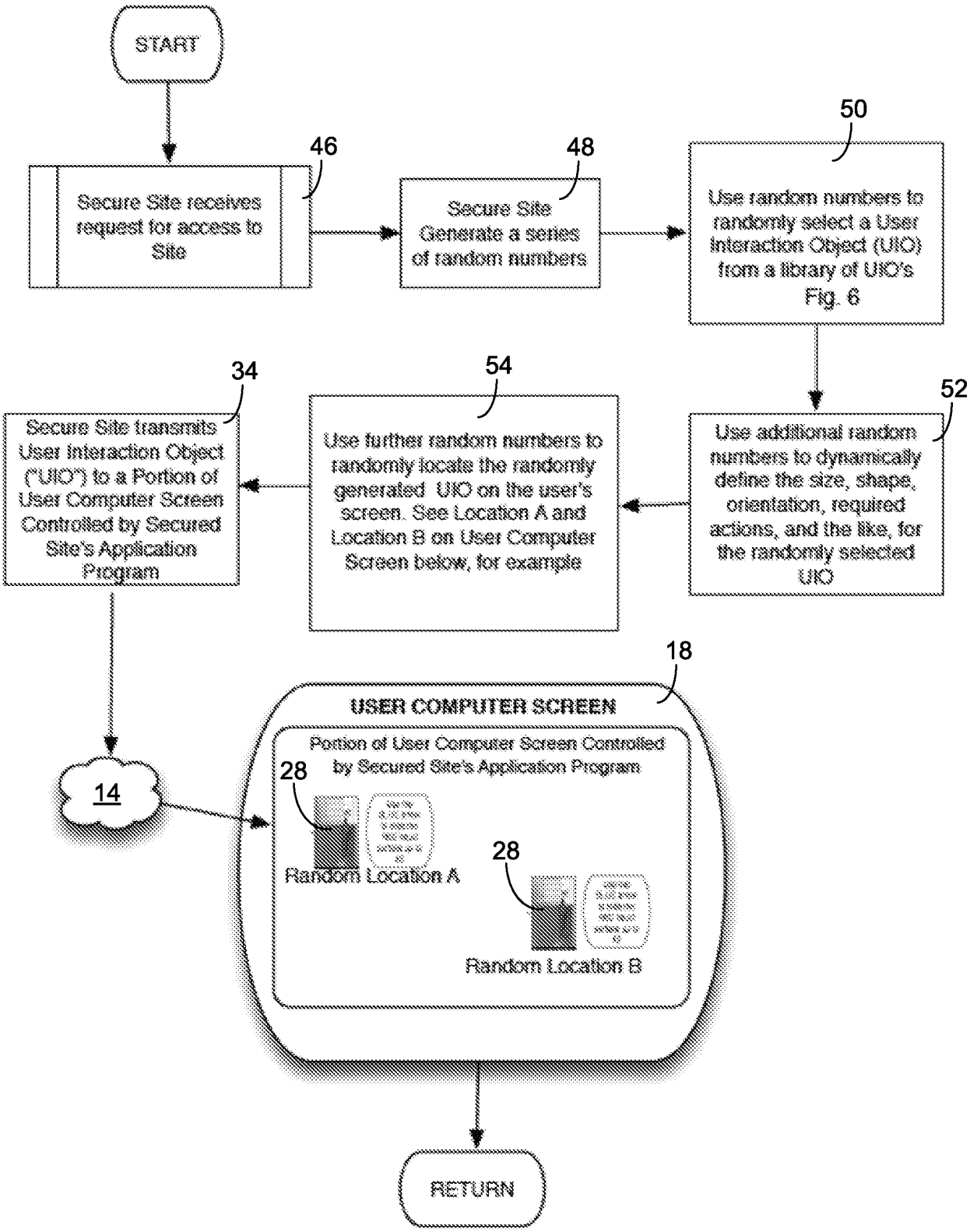

START
30
IS USER REQUESTING ACCESS TO SITE ?
Yes
No
32
Secure Site generates real time unique and random User Interaction Object™
28
Exemplary User Interaction Object
40
Use the BLUE arrow to slide the RED liquid surface up to 40
Secure Site transmits User Interaction Object ("UIO") to a Portion of User Computer Screen Controlled by Secured Site's Application Program
34
14
USER COMPUTER SCREEN
Portion of User Computer Screen Controlled by Secured Site's Application Program
18
36
User responds to secure site UIO with a series of non-keyboard mouse clicks, hand gestures and the like
38
Example of User-generated click stream
> mouse click 1 at pixel 1048, 379
> mouse click 2 at pixel 948, 579
40
User generated click stream is transmitted to Secure Site
14
42
Secure Site computer algorithms decode click stream into actionable data
44
IS USER RESPONSE ACCEPTED ?
Yes
No
46
Grant Access
48
Deny Access
RETURN

FIG. 5


START
46
Secure Site receives request for access to Site
48
Secure Site Generate a series of random numbers
50
Use random numbers to randomly select a User Interaction Object (UIO) from a library of UIO's Fig. 6
52
Use additional random numbers to dynamically define the size, shape, orientation, required actions, and the like, for the randomly selected UIO
54
Use further random numbers to randomly locate the randomly generated UIO on the user's screen. See Location A and Location B on User Computer Screen below, for example
34
Secure Site transmits User Interaction Object ("UIO") to a Portion of User Computer Screen Controlled by Secured Site's Application Program
18
USER COMPUTER SCREEN
Portion of User Computer Screen Controlled by Secured Site's Application Program
14
28
Random Location A
28
Random Location B
RETURN

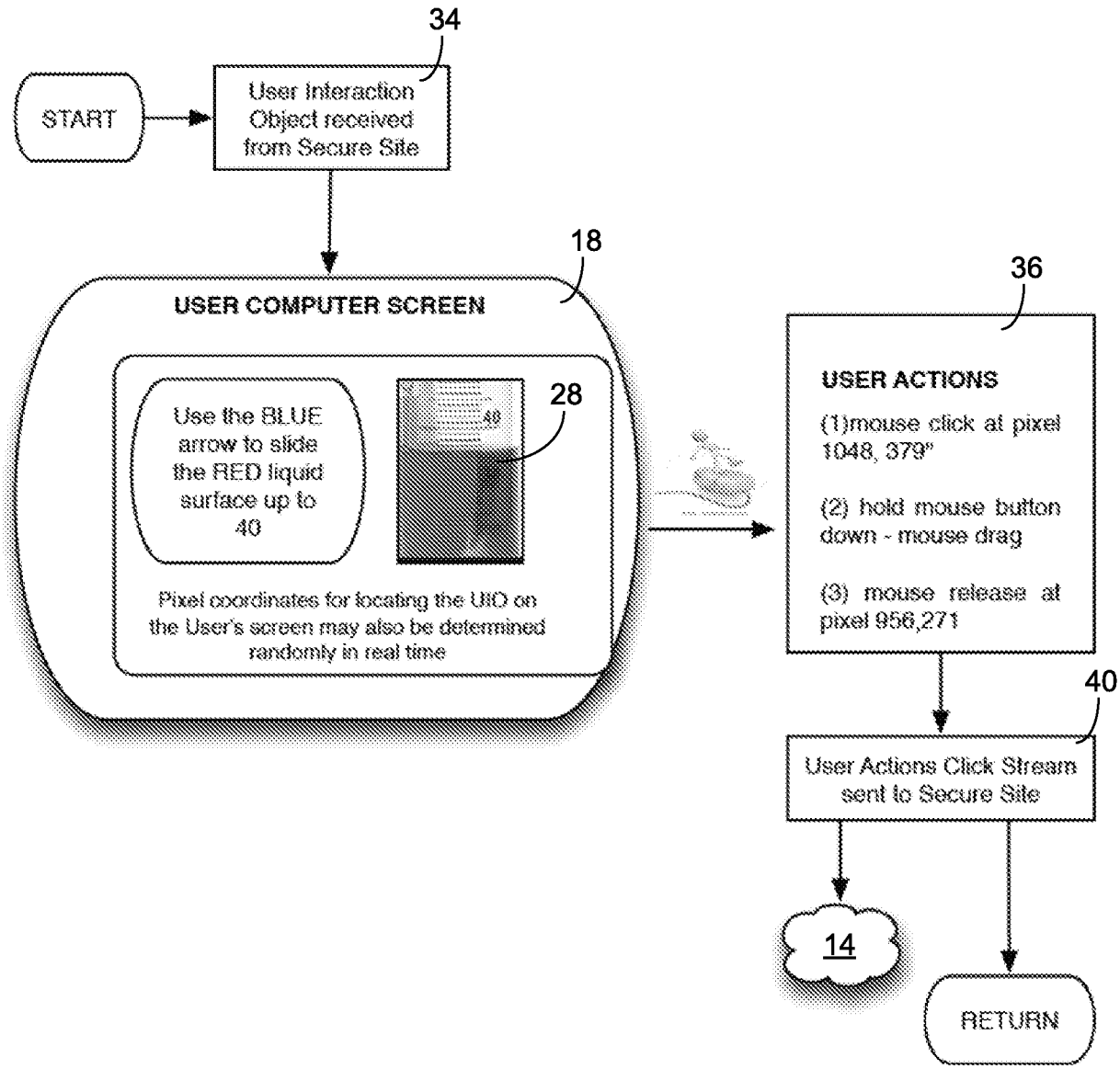
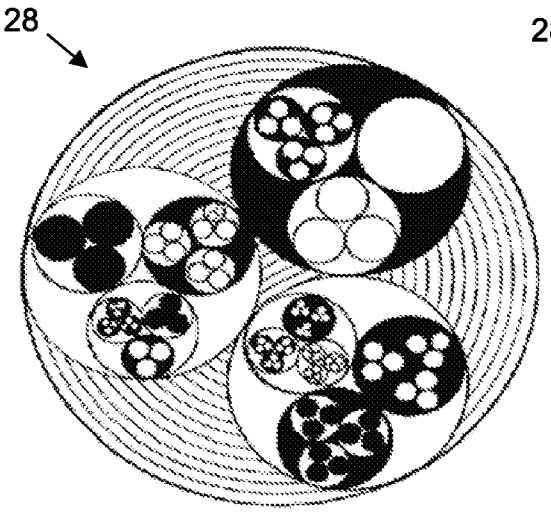
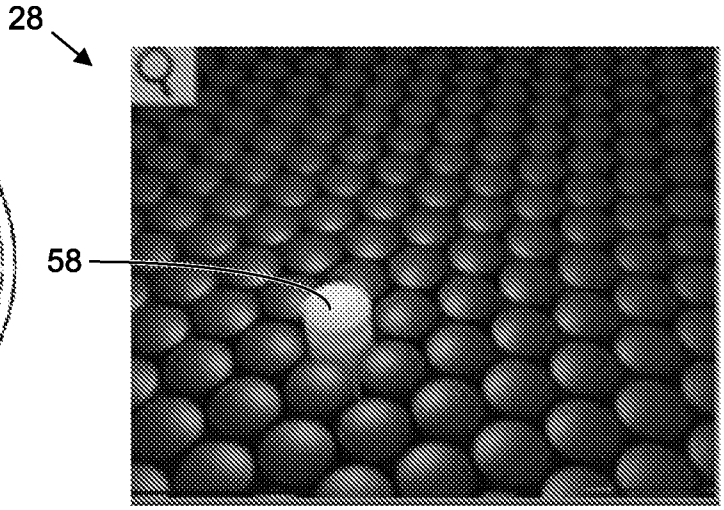

34
START
User Interaction Object received from Secure Site
18
USER COMPUTER SCREEN
Use the BLUE arrow to slide the RED liquid surface up to 40
28
Pixel coordinates for locating the UIO on the User's screen may also be determined randomly in real time
36
USER ACTIONS
(1)mouse click at pixel 1048, 379"
(2) hold mouse button down - mouse drag
(3) mouse release at pixel 956,271
40
User Actions Click Stream sent to Secure Site
14
RETURN

FIG. 6




Click the solid black circles

FIG. 7




Touch the yellow cylinder

FIG. 8

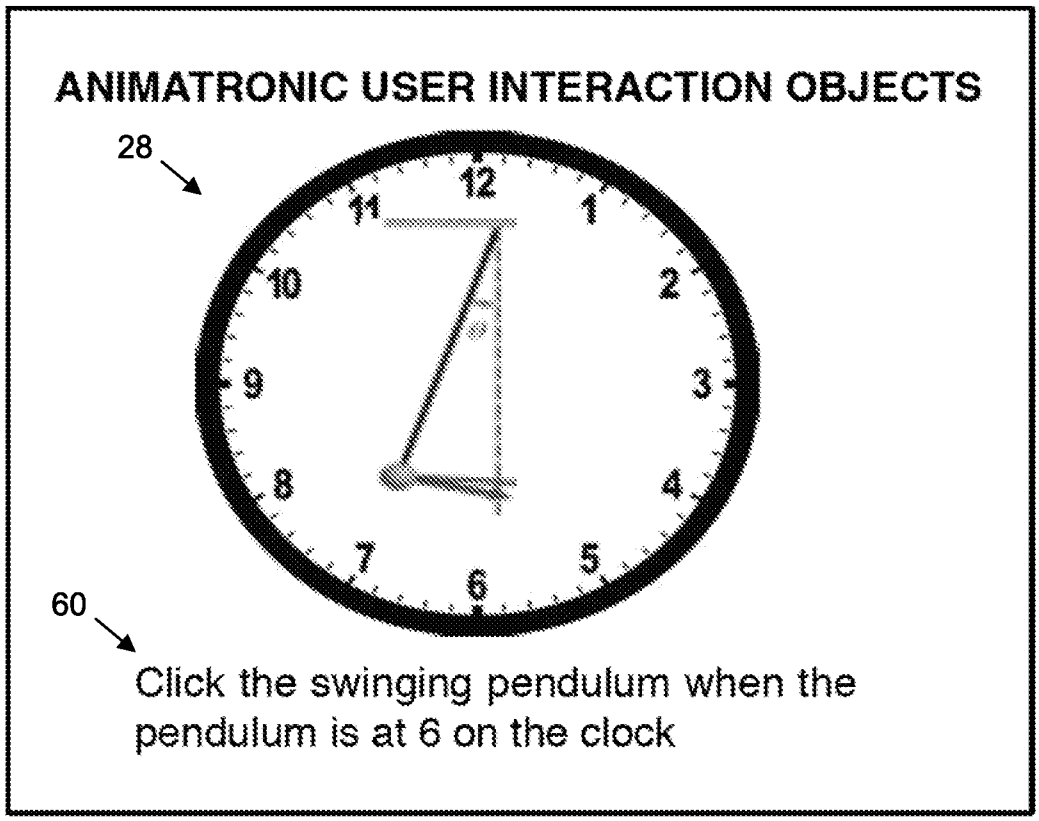



FIG. 9