



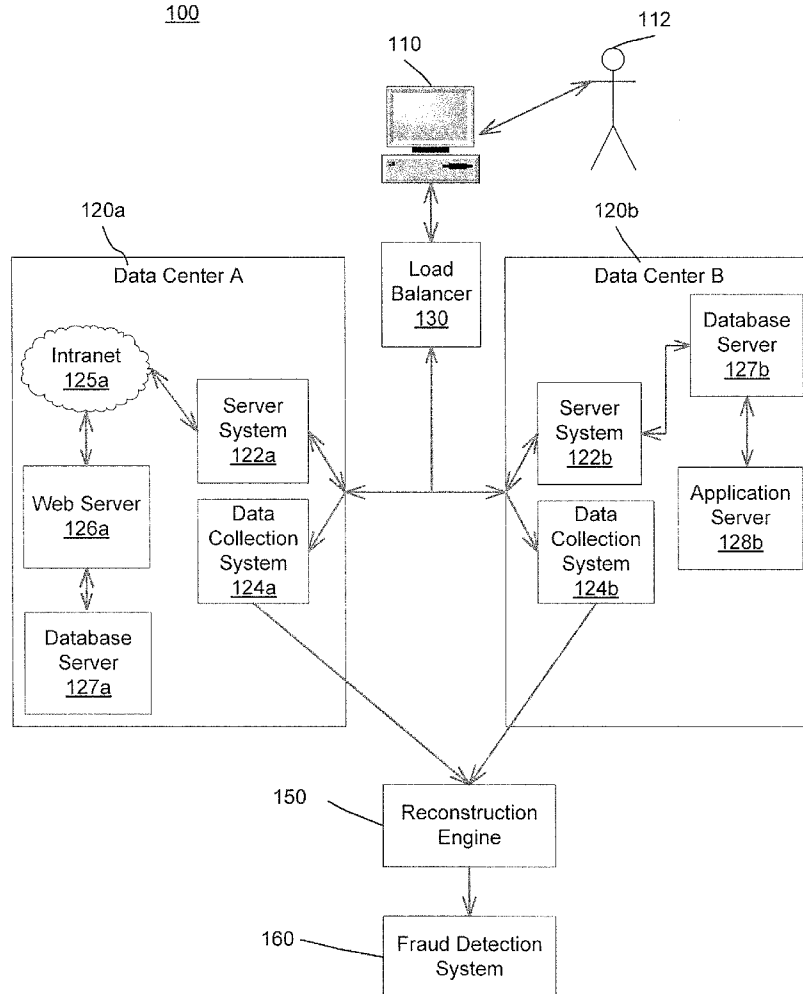
US 20120221721A1

(19) **United States**(12) **Patent Application Publication**
Bhatt et al.(10) **Pub. No.: US 2012/0221721 A1**(43) **Pub. Date: Aug. 30, 2012**(54) **DETECTING FRAUDULENT ACTIVITY**(52) **U.S. Cl. 709/224**(75) **Inventors:** **Vijay C. Bhatt**, Burlington, MA (US); **Stephen Singh**, Westford, MA (US); **Rajandra Laxman Kulkarni**, Burlington, MA (US); **Adam Greenberg**, Lincoln, MA (US)(73) **Assignee:** **FMR LLC**, Boston, MA (US)(21) **Appl. No.:** **13/469,874**(22) **Filed:** **May 11, 2012****Related U.S. Application Data**

(63) Continuation of application No. 11/559,784, filed on Nov. 14, 2006, now Pat. No. 8,180,873.

Publication Classification(51) **Int. Cl.**
G06F 15/173 (2006.01)
G06F 11/30 (2006.01)(57) **ABSTRACT**

Described are computer-based methods and apparatuses, including computer program products, for detecting fraudulent activity. A computing device in a second network receives a plurality of interactions between data centers in a first network and one or more devices, remote from the data centers, of a user, the interactions forming an aggregate user session representing the user's activity over a period of time with the data centers via the devices. The computing device monitors data transaction requests of the interactions and is configured to respond to each request with a response that appears as if the request was executed. If the interactions are harmful, the computing device interdicts the interactions from execution by the data centers and allows the devices to submit additional interactions. If unable to determine whether the subsequent interactions are harmful, the computing device transmits the interactions to the data centers for execution and marks the interactions.

100

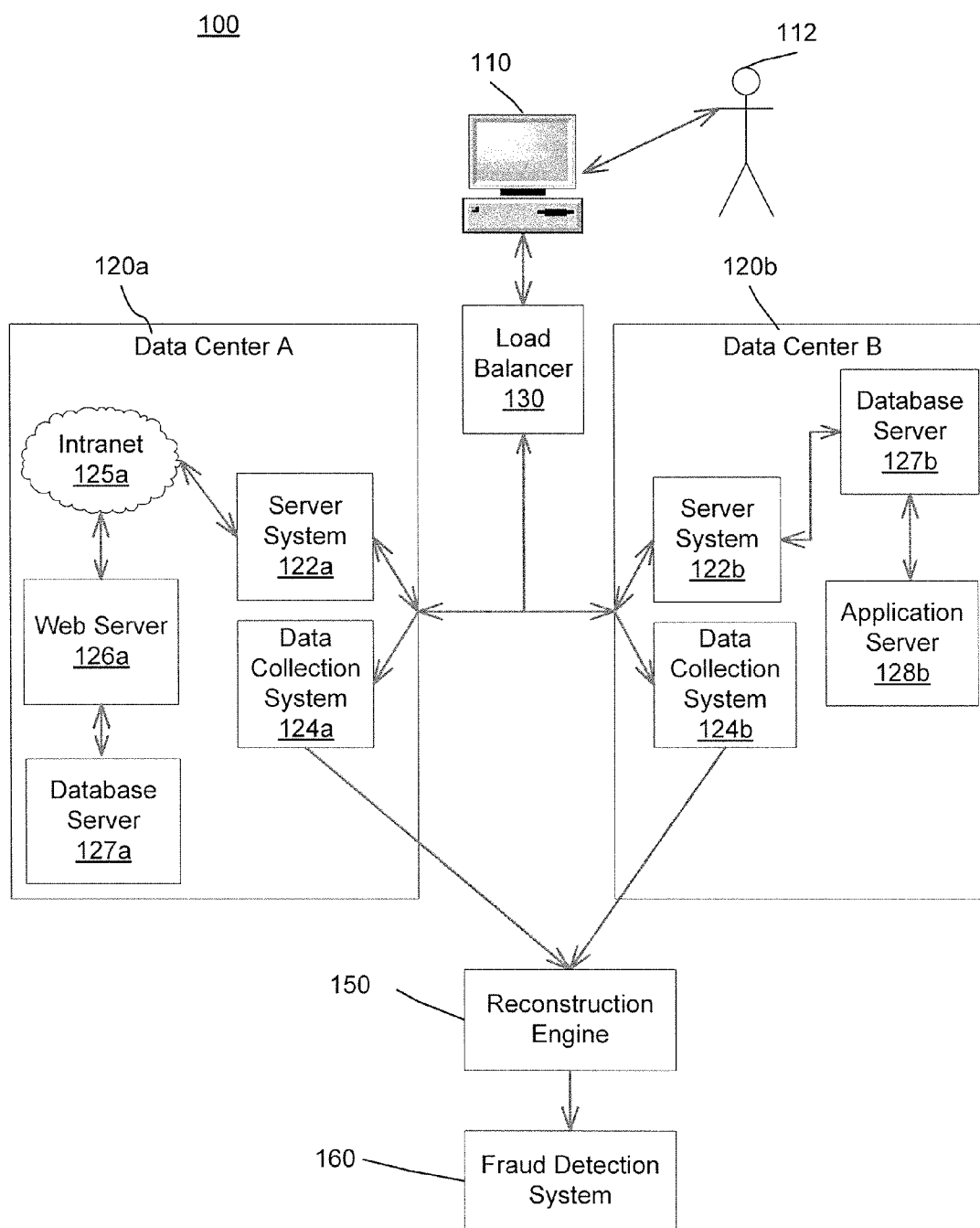


FIG. 1

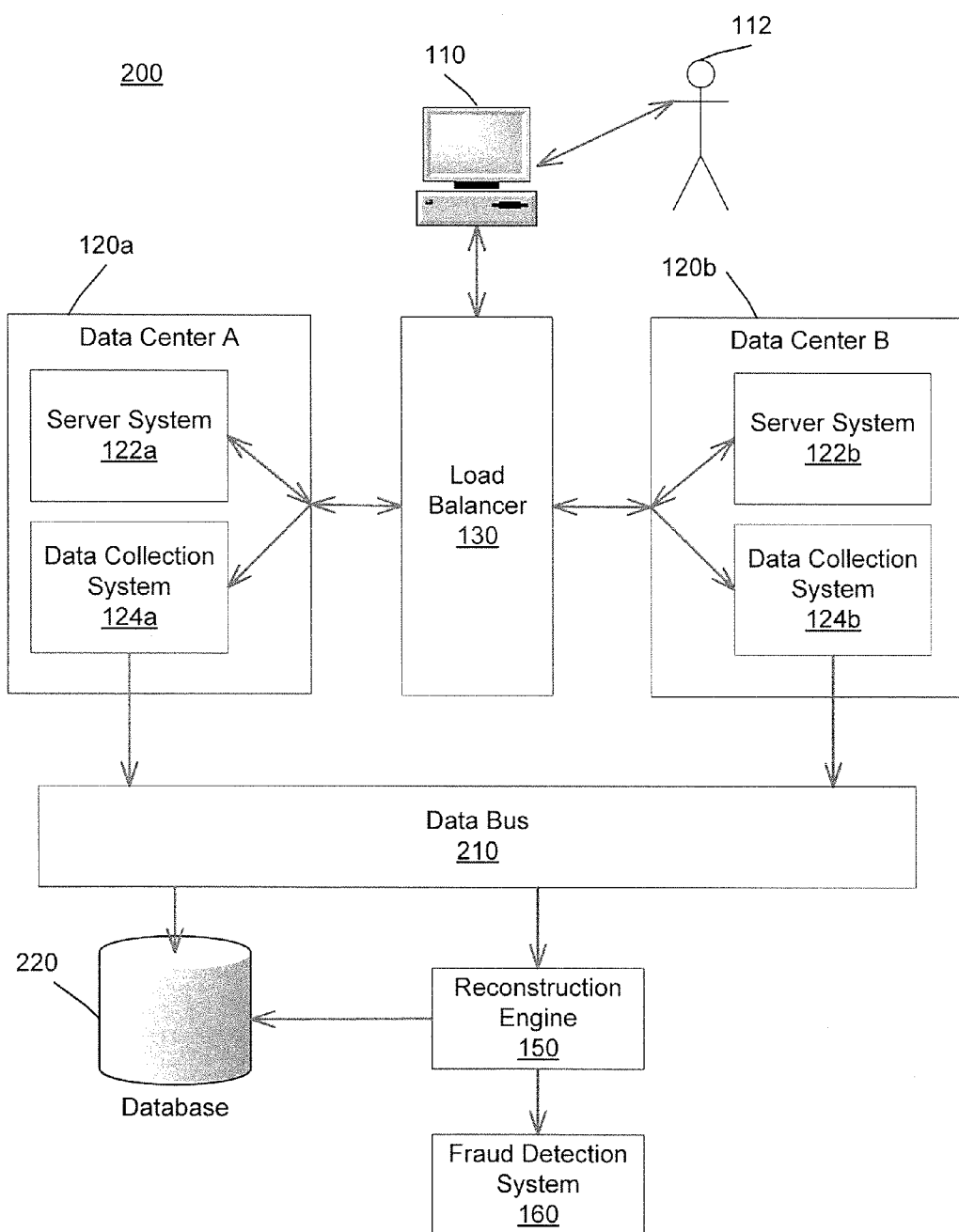


FIG. 2

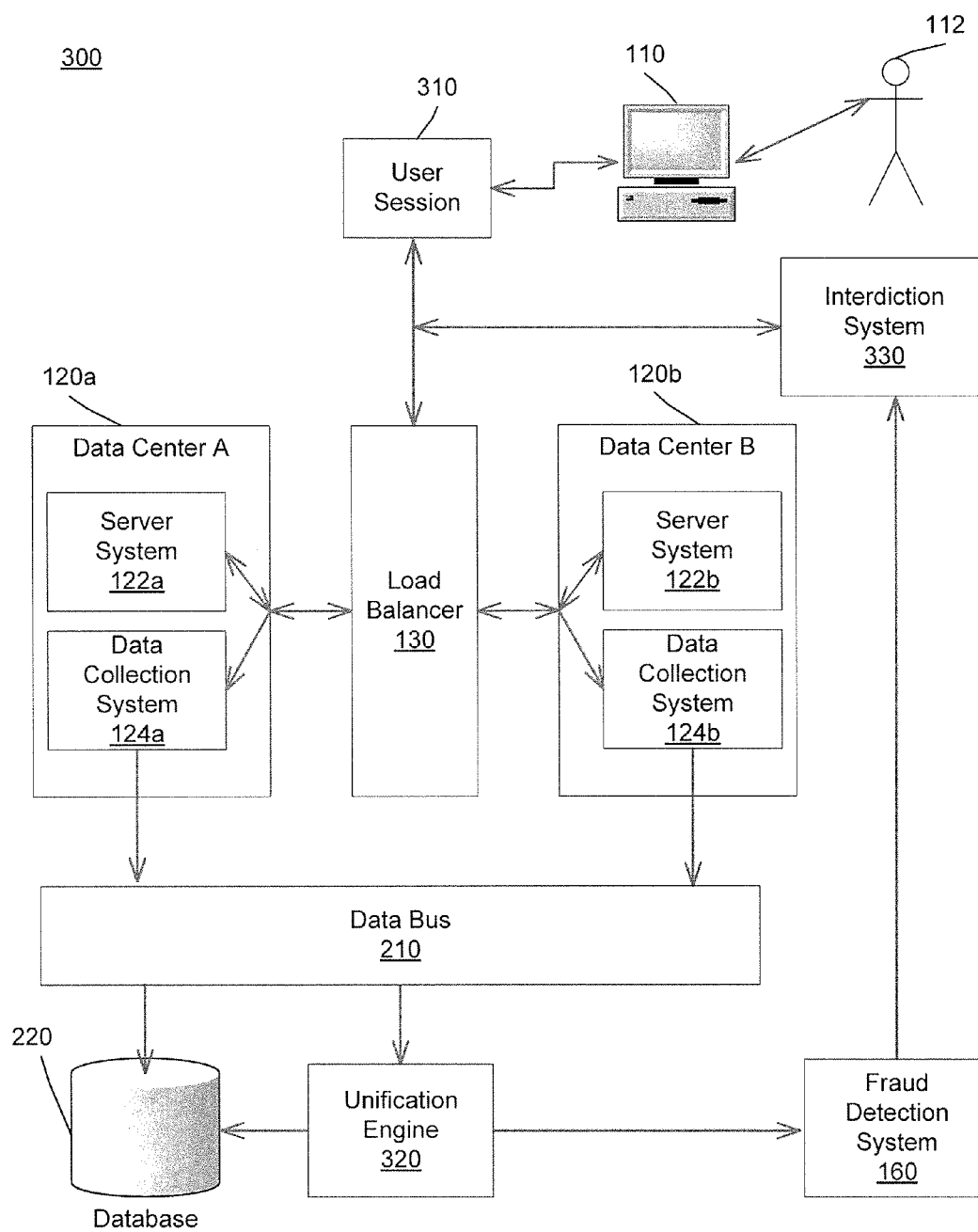


FIG. 3

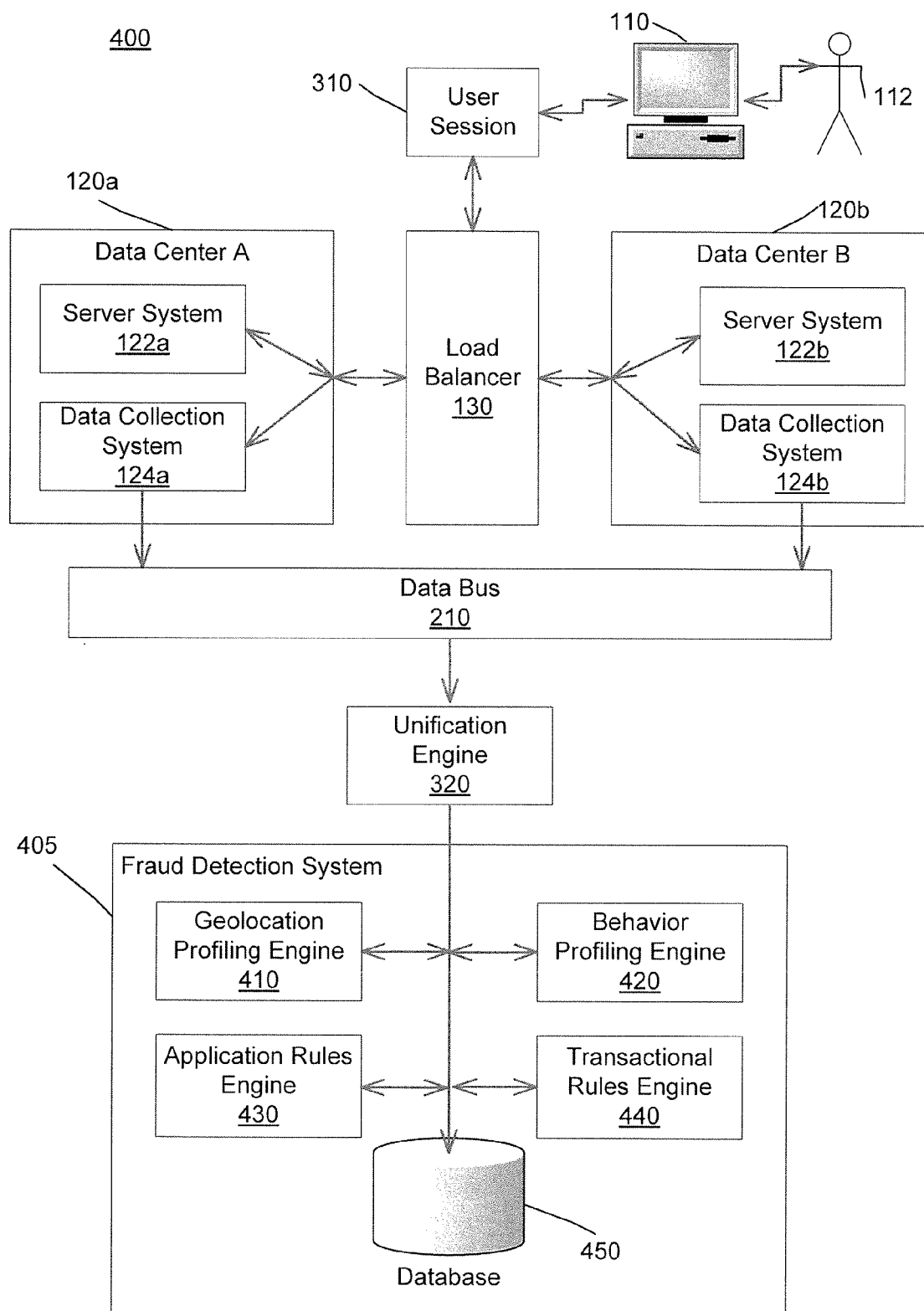


FIG. 4

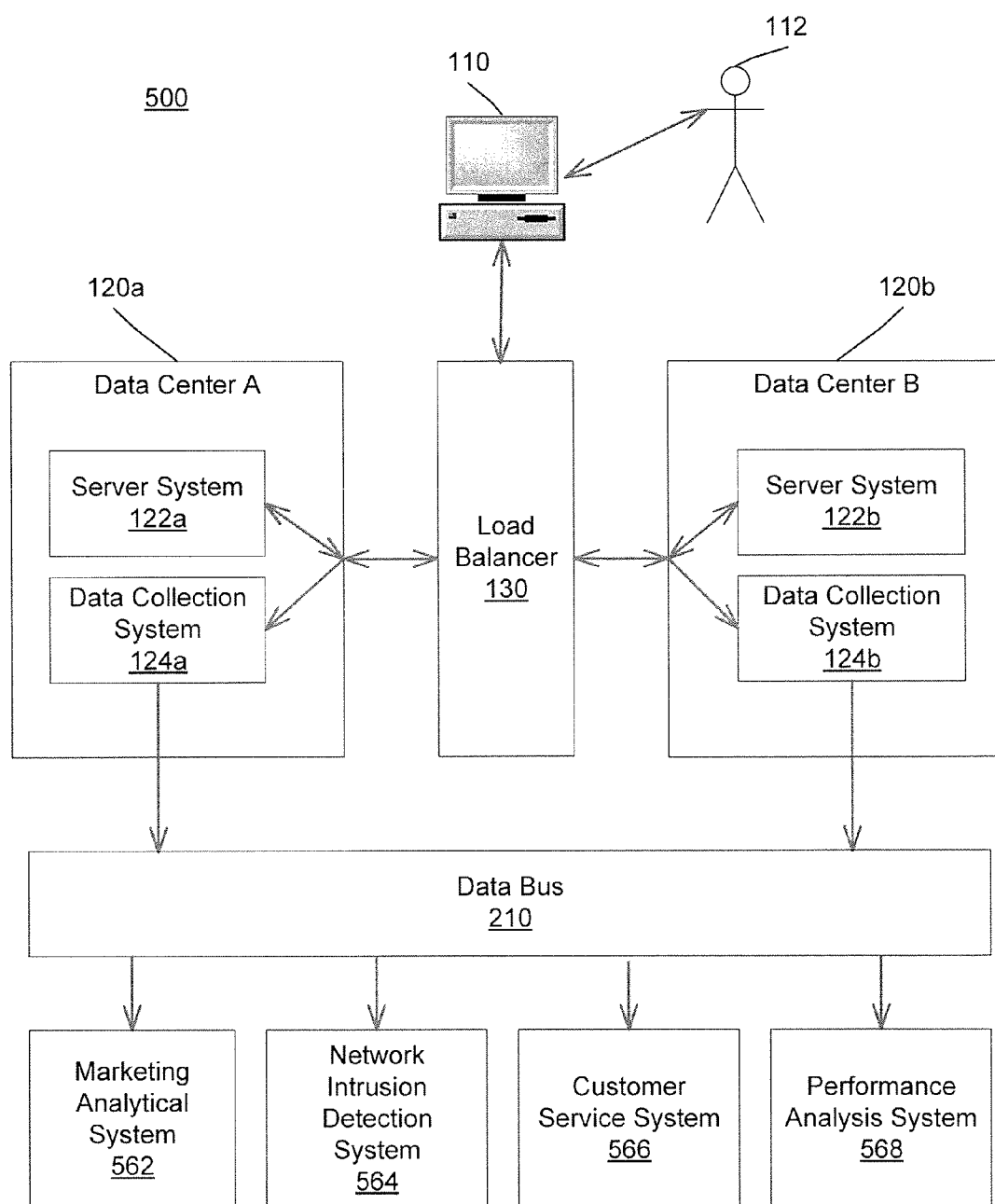


FIG. 5

Log In

600a

SSN or Customer ID* 602a

george

[Forgot Your Customer ID?](#)

PIN 604a

.....

[Forgot Your PIN?](#)

☐ Change your start page

606a

FIG. 6A

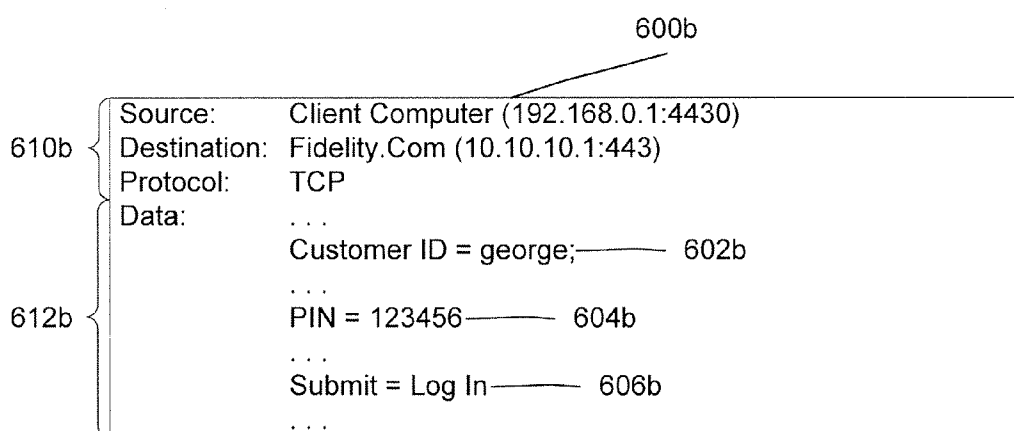


FIG. 6B

600c

Fund EvaluatorSM - Find Funds that Match Your Criteria

Specify one or more criteria then click "Find Now" [Learn more](#)

Investment Category

Fund Performance

Morningstar Rating

FIG. 6C

600d

610d { Source: Client Computer (192.168.0.1:4430)
Destination: Fidelity.Com (10.10.10.1:80)
Protocol: TCP
Data: ...
Investment Category = High Yield Bond;
...
Fund Performance = 10 Year and Greater than or equal to
0%;

612d { ...
Morningstar Rating = Overall 5 Stars
...
Submit = Find Now
...

FIG. 6D

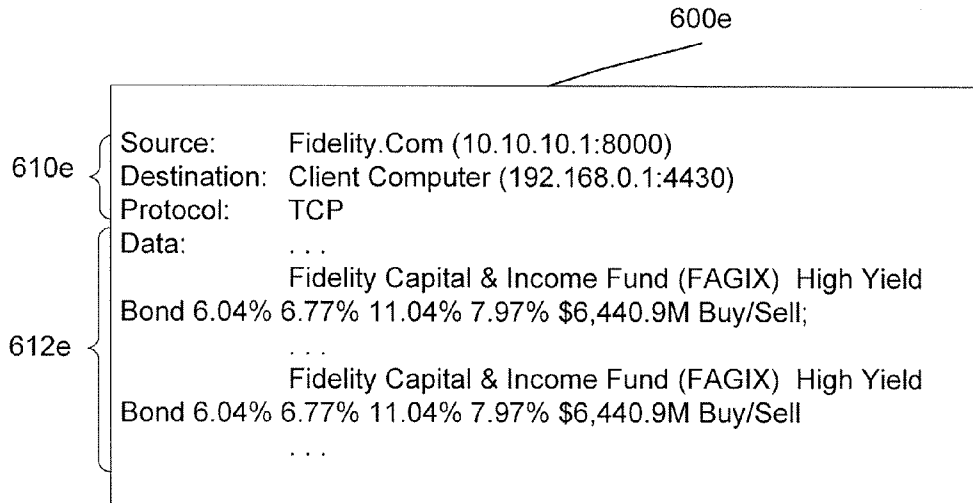


FIG. 6E

600f

First 1 Fidelity fund that match your criteria (1-1 of 1 Fidelity fund available) [View all matching Fidelity Funds](#)

Fidelity Fund Name ^	Investment Category	Load Adjusted Returns*				Net Assets	
		YTD	1 Yr	5 Yr	10 Yr/LOF		
☐ Fidelity Capital & Income Fund (FAGIX)	High Yield Bond	6.04%	6.77%	11.04%	7.97%	\$6,440.9M	Buy/Sell

First 1 fund that match your criteria (1-1 of 1 fund available, including Fidelity funds) [View all matching Funds](#)

Fund Name(all matching funds) ^	Investment Category	Load Adjusted Returns*				Net Assets	
		YTD	1 Yr	5 Yr	10 Yr/LOF		
☐ Fidelity Capital & Income Fund (FAGIX)	High Yield Bond	6.04%	6.77%	11.04%	7.97%	\$6,440.9M	Buy/Sell

[Compare](#) [Add to Watch List](#) [Refine Search](#) [New Search](#)

FIG. 6F

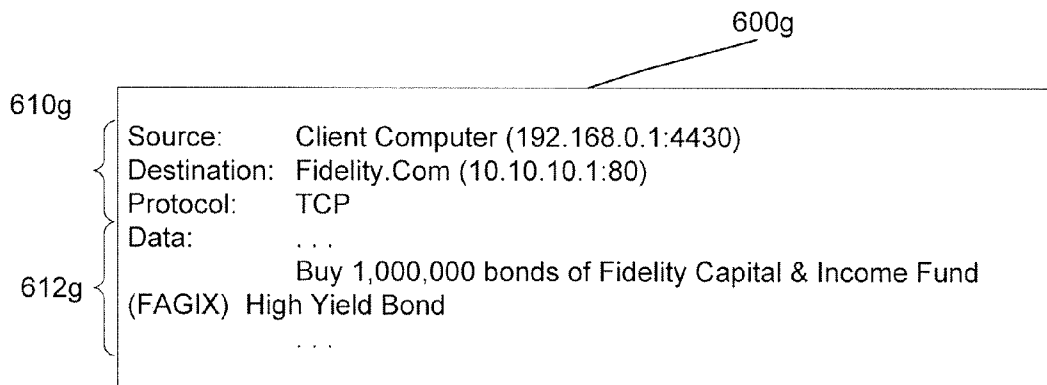


FIG. 6G

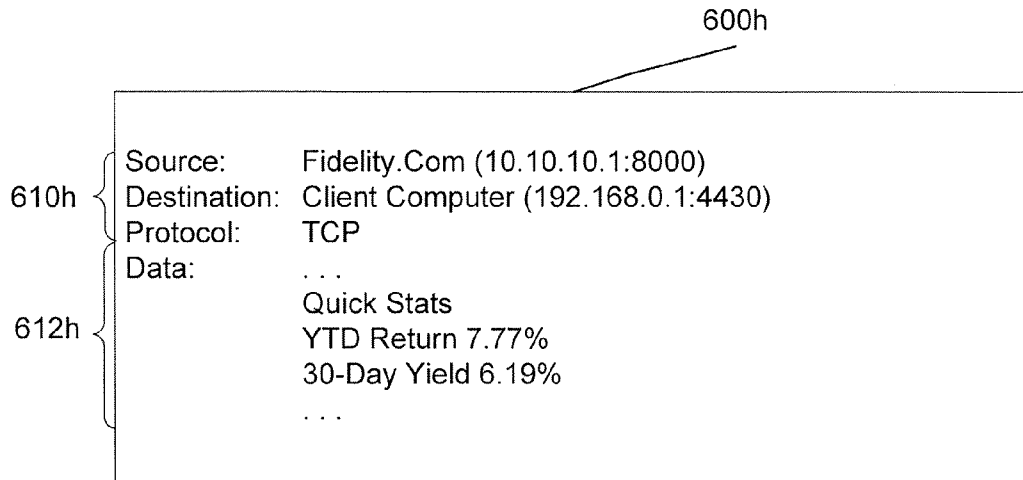


FIG. 6H



FIG. 6I

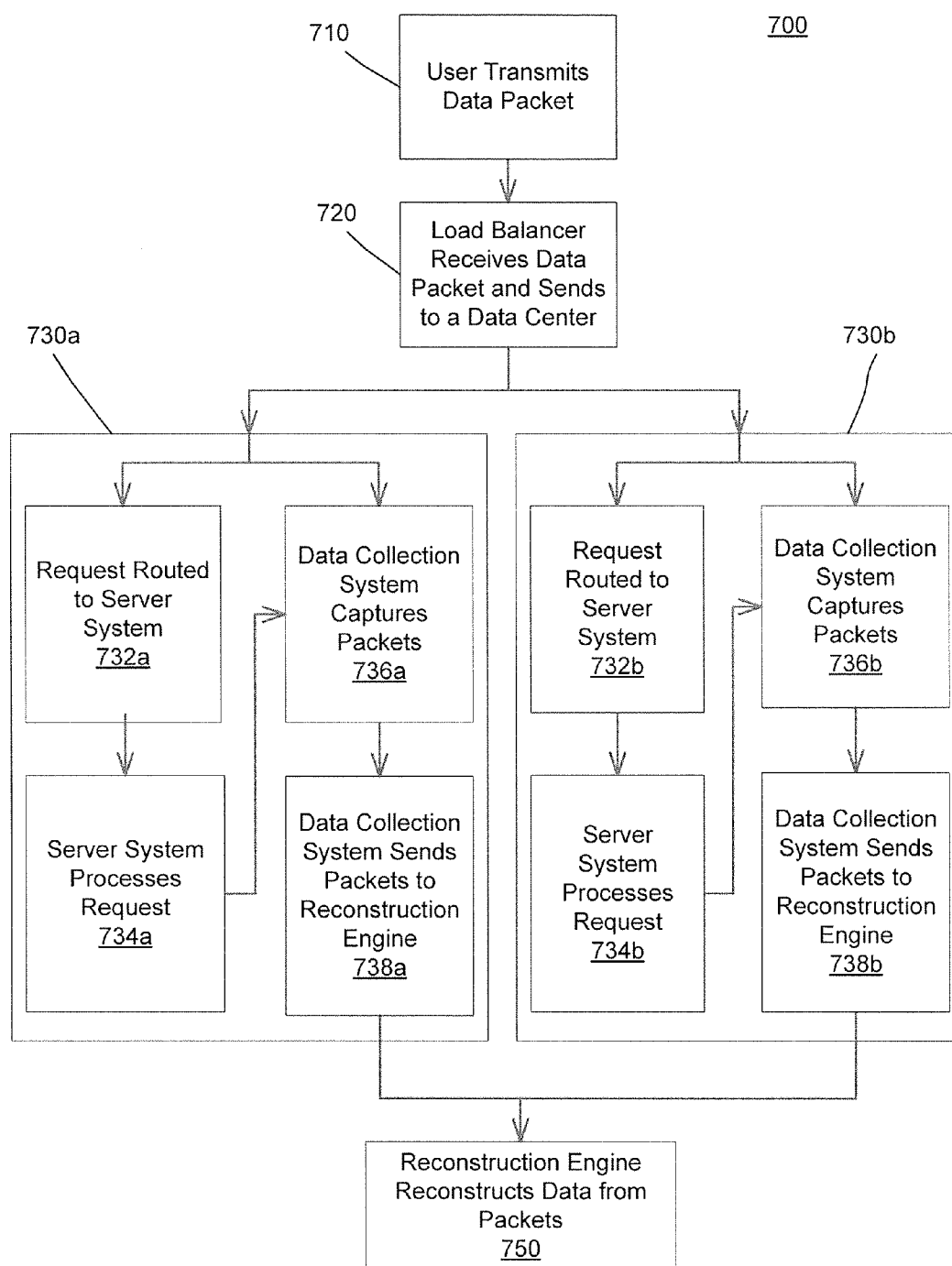


FIG. 7

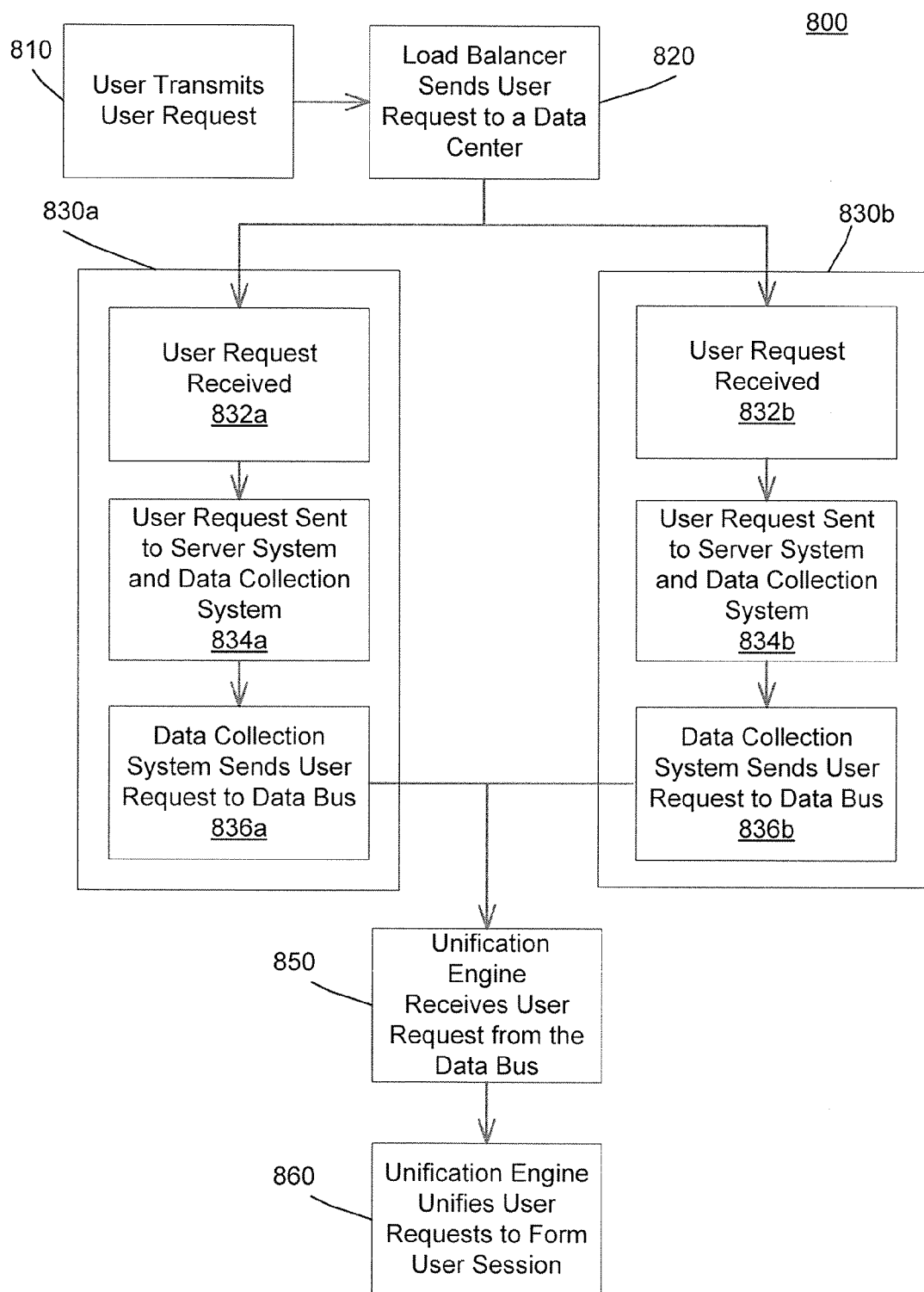


FIG. 8

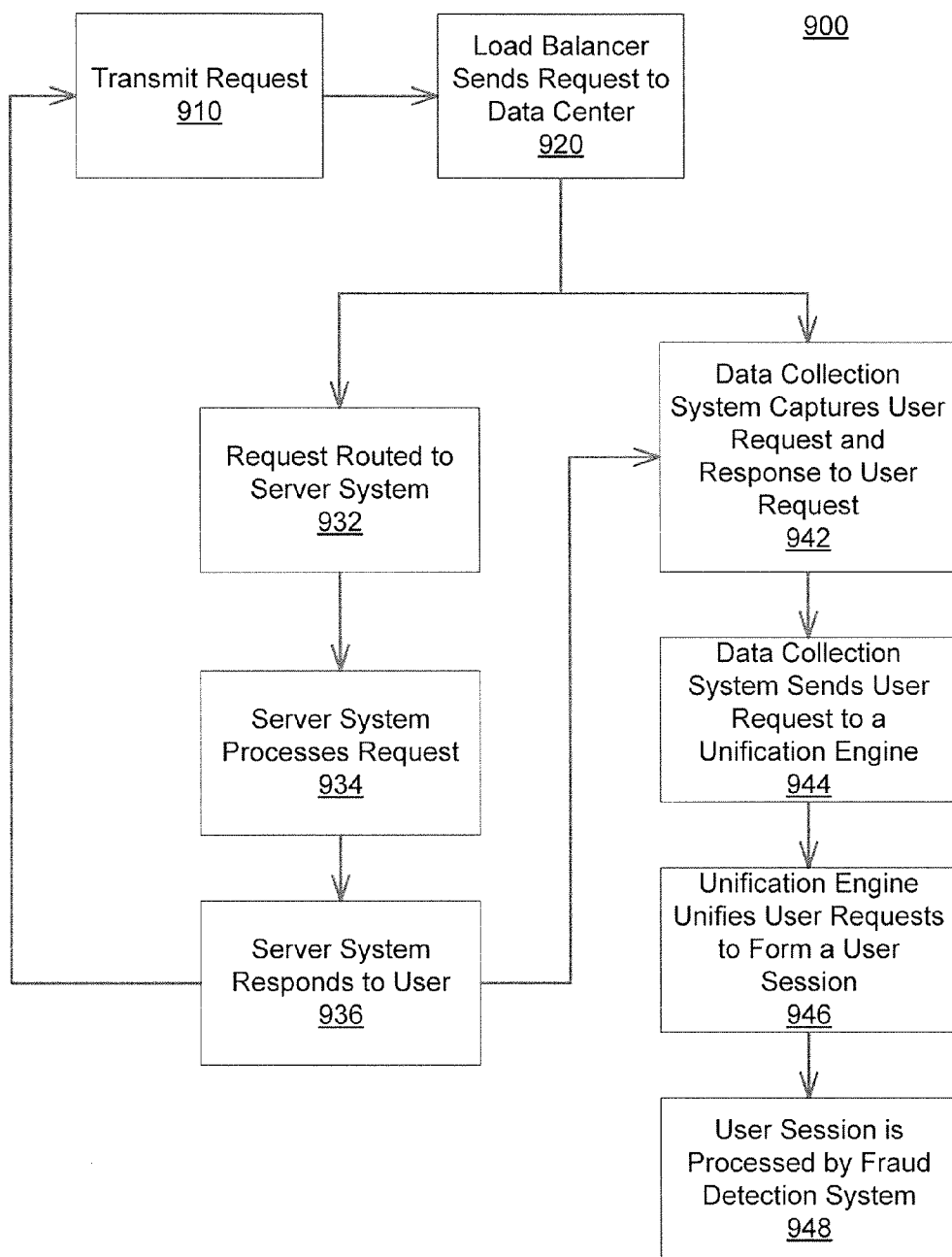


FIG. 9

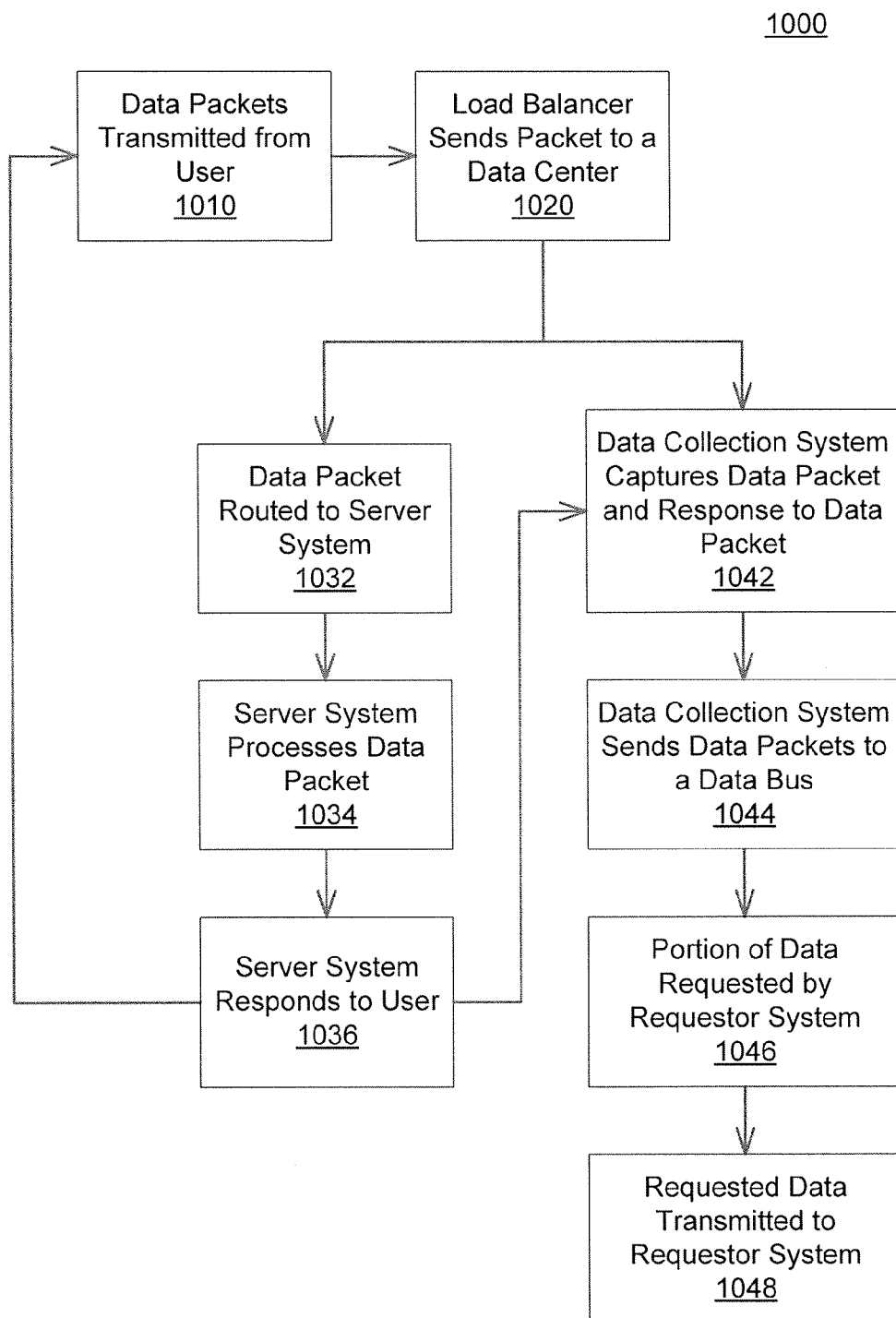


FIG. 10

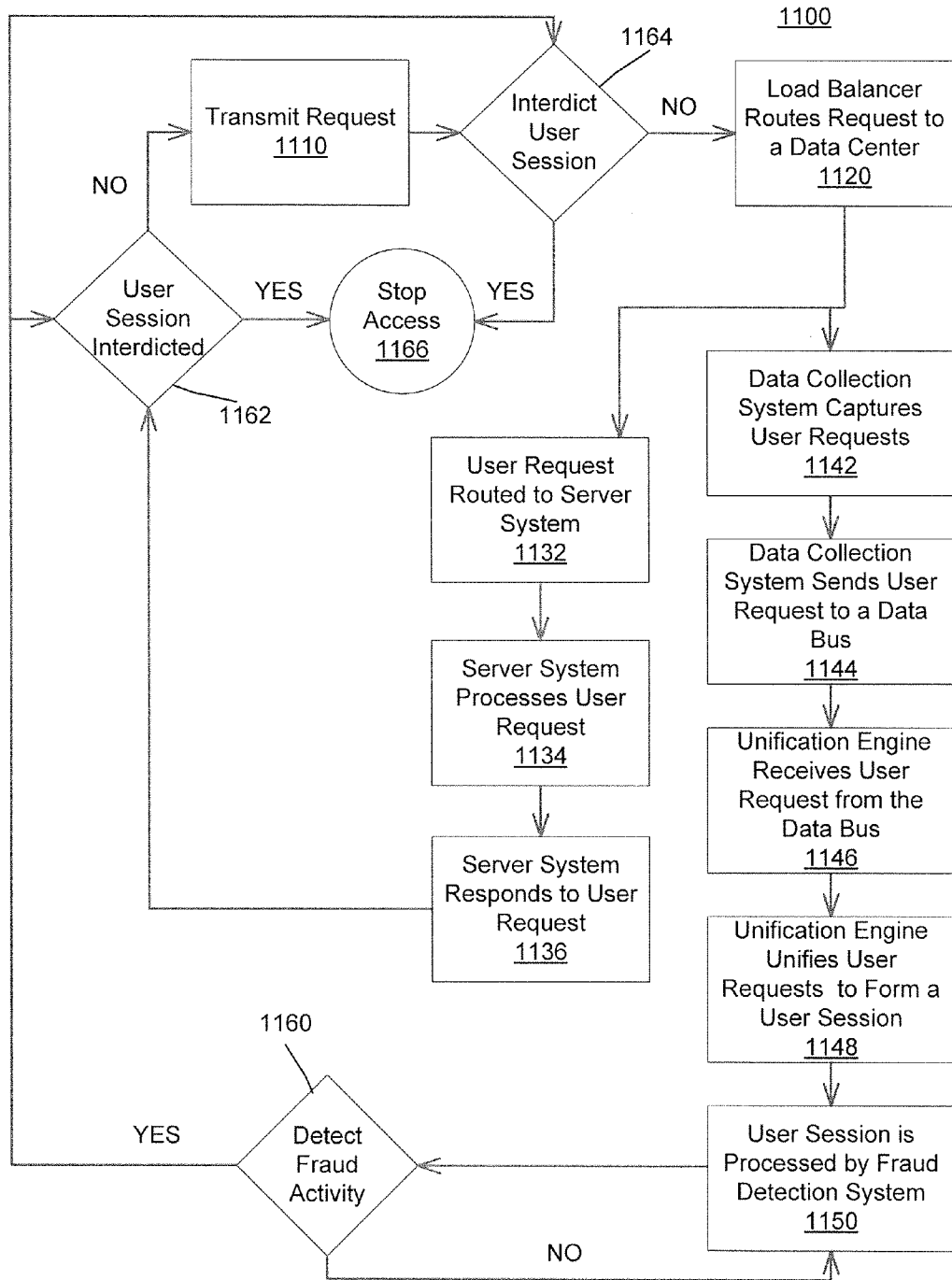


FIG. 11

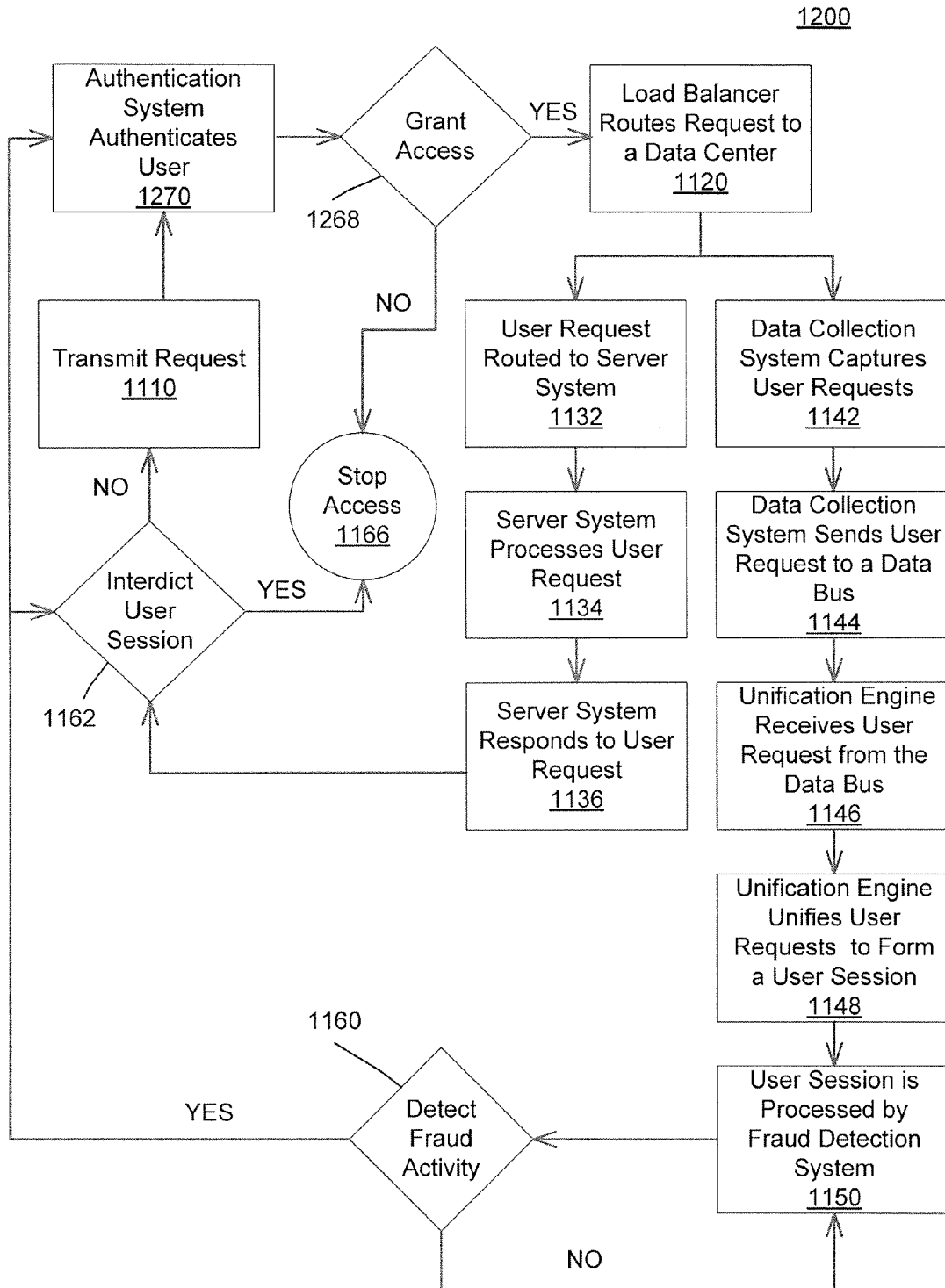


FIG. 12

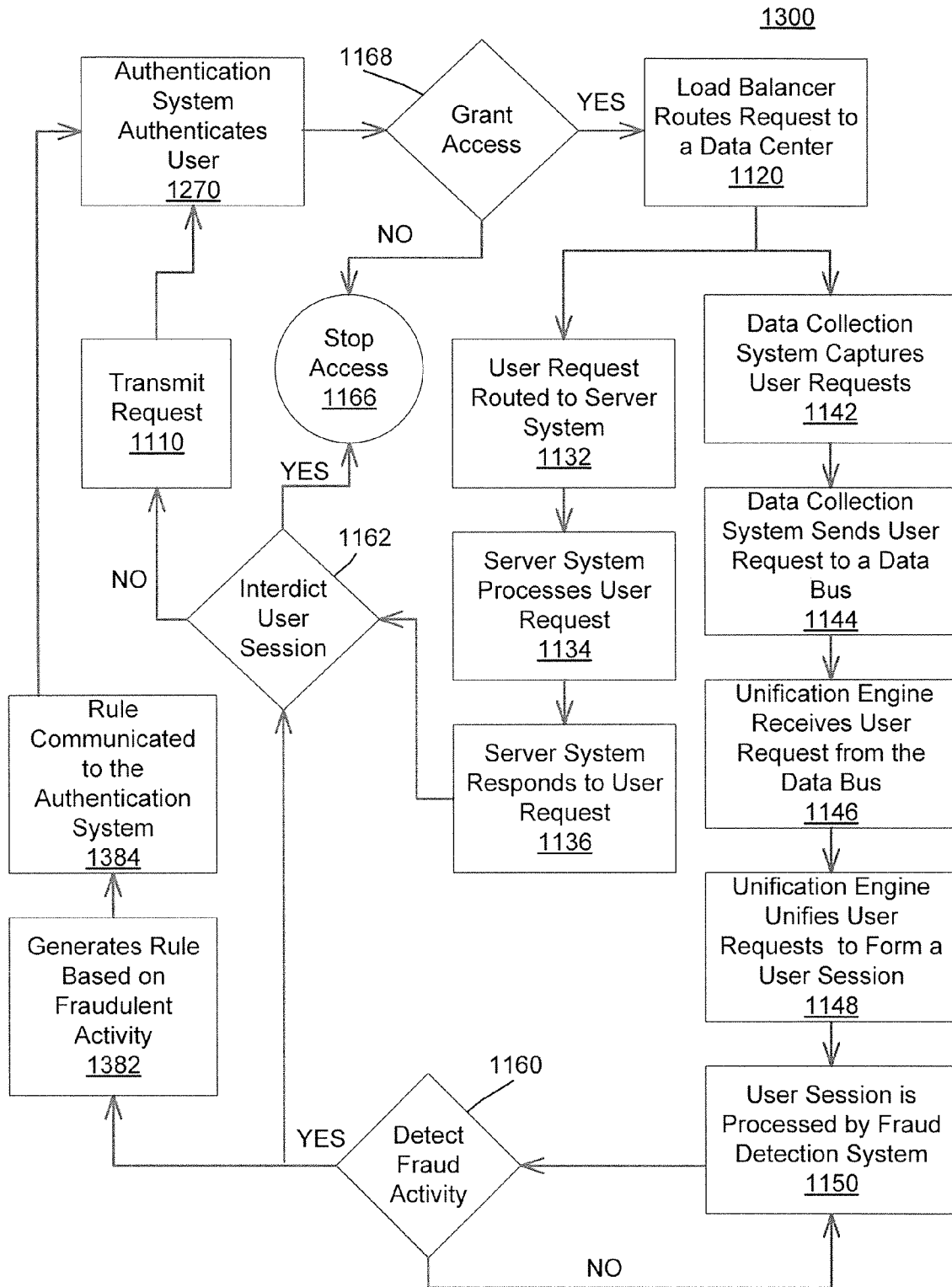


FIG. 13

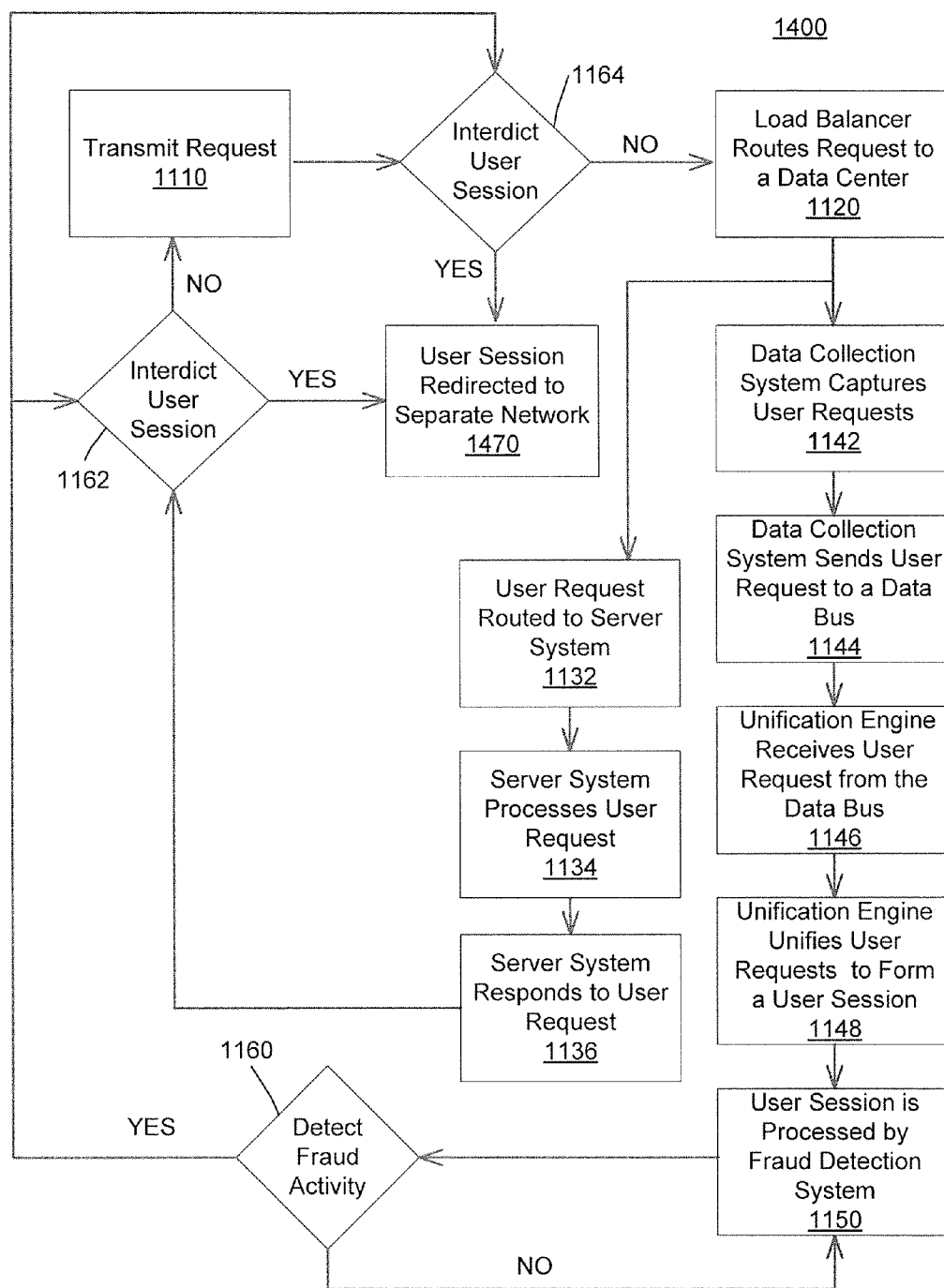


FIG. 14

DETECTING FRAUDULENT ACTIVITY

CROSS REFERENCE TO RELATED APPLICATIONS

[0001] This application is a continuation of U.S. patent application Ser. No. 11/559,784 filed on Nov. 14, 2006, the disclosure of which is hereby incorporated by reference in its entirety.

FIELD OF THE INVENTION

[0002] The present invention relates generally to computer-based methods and apparatuses, including computer program products, for detecting fraudulent activity.

BACKGROUND

[0003] The increased use of networks to access and provide information has caused a dramatic increase in the amount of data transmitted over networks. To handle this increased amount of traffic, computer systems used to store, process, and transmit the information have been increasing in size and distributed among data centers. This distribution among data centers allows for increased speed which allows for decreased response time in the retrieval and processing of information.

[0004] However, the increased speed and diversified data centers create an issue with the data that is sent to a plurality of data centers. The issue is that the data sent to and from users is distributed among the data centers. When the data from only one data center is analyzed, then it is extremely challenging if not impossible to obtain a complete representation of a user's activity.

[0005] The ability to analyze a user's activity as a whole is important in a wide spectrum of industries that provide trusted services over a network. These industries have to be able to analyze user activity to provide feedback and improve performance of the systems. In addition, industries have to be able to effectively and efficiently identify fraudulent activity on their networks.

[0006] Fraudulent activity has been increasing along with the rise in network based activity. Industries have responded by utilizing fraud detection systems to attempt to stop the loss of money and prestige. However, it has been challenging if not impossible for these fraud detection systems to collect the data from all of the data centers in real time and without impacting the customer's experience. Since fraudulent activity is increasing, it is important for industries, such as the financial services industry, to have a fraud detection system that can collect data packets from a plurality of data centers and reconstruct the data for the detection of fraudulent activity and other uses.

SUMMARY OF THE INVENTION

[0007] The invention, in one aspect, features a method for detecting fraudulent activity. A computing device in a second network receives a plurality of interactions between a plurality of data centers in a first network and one or more devices, remote from the data centers, of a user, the plurality of interactions forming an aggregate user session representing the user's activity over a period of time with the plurality of data centers via the one or more devices. The computing device monitors data transaction requests of the plurality of interactions, the computing device configured to respond to each data transaction request with a response that appears to the user as if the request was executed by the respective data

center. If the computing device determines that the interactions are harmful, the computing device interdicts the interactions from execution by the plurality of data centers and allows the one or more devices to submit additional interactions. If the computing device is unable to determine whether the subsequent interactions are harmful, the computing device transmits the interactions to the plurality of data centers for execution and marks the interactions for monitoring purposes.

[0008] The invention, in another aspect, features a system for detecting fraudulent activity. The system includes a computing device in a second network configured to receive a plurality of interactions between a plurality of data centers in a first network and one or more devices, remote from the data centers, of a user, the plurality of interactions forming an aggregate user session representing the user's activity over a period of time with the plurality of data centers via the one or more devices. The computing device is configured to monitor data transaction requests of the plurality of interactions, the computing device configured to respond to each data transaction request with a response that appears to the user as if the request was executed by the respective data center. If the computing device determines that the interactions are harmful, the computing device is configured to interdict the interactions from execution by the plurality of data centers and allow the one or more devices to submit additional interactions. If the computing device is unable to determine whether the subsequent interactions are harmful, the computing device is configured to transmit the interactions to the plurality of data centers for execution and mark the interactions for monitoring purposes.

[0009] The invention, in another aspect, features a computer program product, tangibly embodied in a non-transitory computer readable storage medium, for detecting fraudulent activity. The computer program product includes instructions operable to cause a computing device in a second network to receive a plurality of interactions between a plurality of data centers in a first network and one or more devices, remote from the data centers, of a user, the plurality of interactions forming an aggregate user session representing the user's activity over a period of time with the plurality of data centers via the one or more devices. The computer program product includes instructions operable to cause the computing device to monitor data transaction requests of the plurality of interactions, the computing device configured to respond to each data transaction request with a response that appears to the user as if the request was executed by the respective data center. If the computing device determines that the interactions are harmful, the computer program product includes instructions operable to cause the computing device to interdict the interactions from execution by the plurality of data centers and allow the one or more devices to submit additional interactions. If the computing device is unable to determine whether the subsequent interactions are harmful, the computer program product includes instructions operable to cause the computing device to transmit the interactions to the plurality of data centers for execution and mark the interactions for monitoring purposes.

[0010] In some embodiments, any of the above aspects can include one or more of the following features. In some embodiments, if the computing device determines that the interactions are harmful and if the computing device is able to simulate the interactions, the computing device transmits a response for each interaction to the one or more devices, the

transmitted responses appearing to the user as if the plurality of data centers executed the interactions. If the computing device determines that the interactions are harmful and if the computing device is unable to simulate the interactions, the computing device transmits a response for each interaction to the one or more devices, the transmitted responses appearing to the user as if the plurality of data centers failed to execute the interactions due to a system error. The computing device monitors data transaction requests of additional interactions submitted from the one or more devices.

[0011] In some embodiments, the computing device generates a signature based on the monitoring step and correlates the signature with previous user sessions to determine whether the plurality of interactions and the previous user sessions share one or more characteristics. In some embodiments, the characteristics include at least one of: a common user, a common remote device, a common location, or a common activity pattern. In some embodiments, the signature is generated using one or more data analysis tools that have access to the plurality of interactions.

[0012] In some embodiments, detecting fraudulent activity is performed in real time. In some embodiments, the response that appears to the user as if the request was executed by the respective data center includes dummy data to prevent the user from accessing data stored in the data center. In some embodiments, if the computing device determines that the interactions are harmful, the computing device provides responses to the additional interactions that encourage the user to stay connected to the computing device until an enforcement action is executed. In some embodiments, the responses to the additional interactions extend a session between the computing device and the one or more remote devices.

[0013] In some embodiments, if the computing device determines that the interactions are harmful, the computing device determines one or more vulnerabilities associated with the plurality of data centers based on the interactions submitted by the one or more remote devices.

[0014] Any of the aspects and examples above can provide one or more of the following advantages. They can reconstruct data and/or unify user sessions from multiple data centers. The reconstructed data and/or unified user sessions can be used for analysis as a complete or partial representation of a user's activity. The reconstruction of data packets into data allows for an improved representation of the user's activity to be analyzed, since data packets from only one data center most likely only shows the user's interactions with that data center and not the interactions with all of the data centers. A representation of the user's activity allows for a holistic analysis of the activity to detect fraudulent activity, to provide feedback to the user and of the system, and/or to improve performance of the system.

[0015] Other advantages include the unification of user requests into user sessions for fraud detection. The unification of user requests that are transmitted to a plurality of data centers allows the fraud detection system to analyze an improved representation of the user's activity. This improved representation of the user's activity is more than the user's interactions with one data center in the plurality of data centers. It is part or all of the user's activity with the plurality of data centers. This allows the user's activity to be analyzed as a whole instead of piece by piece.

[0016] Other advantages include the capture of data packets and/or user requests from the network without interfering

with the transmission of the data packets and/or user requests to the rest of the network (e.g., the server system). The processing of the data packets and/or the user requests by the server systems is not interfered with by the capture of the data packets and/or the user requests.

[0017] Other advantages include the processing of the data and/or user session by the fraud detection system in real time. The data collection system collects the data packets and/or user requests at the same time that the server systems receive the data packets and/or user requests. The reconstructed data and/or unified user sessions are processed by the fraud detection system at the same time that the server system processes the data and/or user request. The real time fraud detection allows for the interdiction of fraudulent activity before the fraudulent activity can harm the system (e.g., the fraud detection system can stop fraudulent transactions before the fraudulent transaction is completely processed).

[0018] Other advantages include the memory module being volatile memory. The processing of data and/or user sessions to determine whether fraudulent activity exists occurs in real time. The fraud detection system utilizes the fast access capabilities of the volatile memory to access the elements stored in the memory module and analyze the data and/or user sessions using the stored elements in real time.

[0019] Other advantages include the data bus subscription service. The data bus subscription service is the subscription of data feeds on the network. The subscription of data feeds on the network allows other requestor systems to access the data that is collected from the plurality of data centers. By providing the subscription service, the other requestor systems, such as a performance analysis system and a customer service system, can analysis the entire representation of the user's activity instead of just a portion of the user's activity that is transmitted through one data center out of the plurality of data centers.

[0020] Other aspects and advantages of the present invention will become apparent from the following detailed description, taken in conjunction with the accompanying drawings, illustrating the principles of the invention by way of example only.

BRIEF DESCRIPTION OF THE DRAWINGS

[0021] The foregoing and other objects, features, and advantages of the present invention, as well as the invention itself, will be more fully understood from the following description of various embodiments, when read together with the accompanying drawings.

[0022] FIG. 1 is a functional block diagram of an exemplary system which reconstructs the data transmitted on the system and transmits the data to a fraud detection system.

[0023] FIG. 2 is a functional block diagram of an exemplary system depicting the data being transported from the data centers to a data bus.

[0024] FIG. 3 is a functional block diagram of an exemplary system showing the unification of user sessions from user requests.

[0025] FIG. 4 is a functional block diagram of an exemplary system depicting parts of the fraud detection system including a database of user sessions.

[0026] FIG. 5 is a functional block diagram of an exemplary system showing systems that receive data from the data bus.

[0027] FIG. 6A is a screen shot of a login module.

[0028] FIG. 6B is a diagram of information sent to a login module.

[0029] FIG. 6C is a screen shot of a search module.

[0030] FIG. 6D is a diagram of information transmitted to a search module.

[0031] FIG. 6E is a diagram of information received from a search module.

[0032] FIG. 6F is a screen shot of information received from a search module.

[0033] FIG. 6G is a diagram of information transmitted to a transaction module.

[0034] FIG. 6H is a diagram of information received from an information module.

[0035] FIG. 6I is a screen shot of information received from an information module.

[0036] FIG. 7 is a flowchart depicting the transmission of data through an exemplary system.

[0037] FIG. 8 is a flowchart depicting the transmission of user sessions through an exemplary system.

[0038] FIG. 9 is a flowchart showing the processing of a user request being transmitted to a data center and through both the server system at the data center and the data collection system.

[0039] FIG. 10 is a flowchart showing the processing of data packets through the system and transmission of the data packets to requestor systems.

[0040] FIG. 11 is a flowchart showing the interdiction of user sessions when fraudulent activity is detected.

[0041] FIG. 12 is a flowchart showing the communication of fraudulent activity to the authentication system.

[0042] FIG. 13 is a flowchart showing the creation of rules in response to the detection of fraudulent activity.

[0043] FIG. 14 is a flowchart showing the redirection of a user session to a separate network to mitigate fraudulent activity.

DETAILED DESCRIPTION

[0044] FIG. 1 is a functional block diagram of an exemplary system 100 which reconstructs the data transmitted on the system and transmits the data to a fraud detection system 160. In some examples, the data includes records of interactions between the user 112 and the system 100. The records of interactions include, for example, buy and sell orders from the user 112 to a brokerage company and/or credit card transactions of the purchase of products and services by the user 112.

[0045] The transmission on the system 100 can occur over packet-based networks and/or circuit-based networks. Packet-based networks can include, for example, the Internet, a carrier internet protocol (IP) network (e.g., local area network (LAN), wide area network (WAN), campus area network (CAN), metropolitan area network (MAN), home area network (HAN)), a private IP network, an IP private branch exchange (IPBX), a wireless network (e.g., radio access network (RAN), 802.11 network, 802.16 network, general packet radio service (GPRS) network, HiperLAN), and/or other packet-based networks. Circuit-based networks can include, for example, the public switched telephone network (PSTN), a private branch exchange (PBX), a wireless network (e.g., RAN, Bluetooth, code-division multiple access (CDMA) network, time division multiple access (TDMA) network, global system for mobile communications (GSM) network), and/or other circuit-based networks.

[0046] The system 100 includes the user 112 who interacts with a transmitting device 110 which transmits data in one or

more parts as data packets to a load balancer 130. In some examples, the user 112 uses any type of transmitting device 110 to transmit the data packets. The transmitting device 110 includes, for example, a computer, a computer with a browser device, a telephone, an IP phone, a mobile device (e.g., cellular phone, personal digital assistant (PDA) device, laptop computer, electronic mail device), and/or other communication devices. The browser device includes, for example, a computer (e.g., desktop computer, laptop computer) with a world wide web browser (e.g., Microsoft® Internet Explorer®, Mozilla® Firefox). The mobile computing device includes, for example, a Blackberry®.

[0047] The load balancer 130 can be a network system. The network system includes, for example, a network router, a network switch, a network hub, a computer and/or other communication devices. The network system includes, for example, one or more network adapters (e.g., 10/100/1000 Base-T network interface adapter card (NIC), 1000 Base-SX NIC, 1000 Base-LX NIC, 1000 Base-FX NIC). The network system includes one or more modules that process the data packets. The module is implemented in digital electronic circuitry, computer hardware, firmware, and/or software. The module includes, for example, a data routing module, an internet protocol (IP) routing module, a domain name service (DNS) routing module, and/or other routing modules.

[0048] In some examples, the system 100 receives data packets from other systems. The other systems include one or more network systems that each contains one or more transmitting devices 110. The other systems transmit data to the load balancer 130 for transmission to the data centers 120a and 120b for processing.

[0049] The load balancer 130 transmits the data packets to the different data centers 120a or 120b based on load balancing techniques. In some examples, the load balancer 130 transmits the data packets to a data center selected from the plurality of data centers 120a and 120b according to the available capabilities of the data centers (e.g., processor availability, disk capacity), the conditions of the network (e.g., packet trip time, packet losses), a quality of service indicators on the data packet, application availability, number of connections to each data center, and/or a pre-defined routing instruction.

[0050] FIG. 1 shows an exemplary system 100 with data centers A 120a and B 120b. Although the system 100 is shown with two data centers 120a and 120b, other examples include any number of data centers (e.g., three, four, ten, twenty, one hundred, one thousand, ten thousand). Similarly, although the system 100 illustrates one load balancer 130, other examples contain a plurality of load balancers in multiple layers. For example, a user's transmitting device 110 transmits data packets to the first load balancer 130 which then has a layer of load balancers to which the first load balancer 130 transmits the data packet. The second load balancer on the layer below the first load balancer 130 transmits the data packet to a data center selected from the plurality of data centers.

[0051] The data packets in the data centers 120a and 120b are transmitted to the server systems 122a and 122b and the data collection systems 124a and 124b, respectively. For example, a data packet is transmitted from the user's transmitting device 110 to the load balancer 130. The data packet is transmitted to data center A 120a from the load balancer 130. The data packet is transmitted in the data center A 120a to the server system 122a and to the data collection system 124a.

[0052] The transmission of the data packet in the data center A 120a to the server system 122a and to the data collection system 124a can occur simultaneously and independently from each other using for example, a network device. The network device includes, for example, a network router, a network firewall, a network hub, a network switch, a computer, and/or other network devices (e.g., Gigamon GigaVUE-MP available from Gigamon Systems LLC).

[0053] The server systems 122a and 122b each can include one or more servers (e.g., 126a, 127a, 127b, and 128b). The server includes, for example, a web server (e.g., 126a), an application server (e.g., 128b) (e.g., Oracle® Application Server 10g available from Oracle Corporation), a database server (e.g., 127a and 127b) (e.g., Oracle® Database 10g available from Oracle Corporation), a communication server, a fax server, a file server, a game server, an authentication server (e.g., RSA® Authentication Manager available from RSA Security Inc.), a desktop computer, a central ad server, a file transport protocol server, an image server, a mail server, a news server, a proxy server, a printer server, a sound server, a streaming media server, a terminal server, a firewall, a network router, a network hub, a network (e.g., an intranet 125a) and/or a network switch.

[0054] In some examples, the data collection system 124a and 124b includes one or more computing devices which can be, for example, a computer, a laptop computer, a network router, a network switch, and/or a network hub (e.g., Radware® AS4 available from Radware Ltd., Covelight Inflight™ 5000 available from Covelight System, Inc., Gigamon GigaVUE-MP available from Gigamon Systems LLC).

[0055] The data collection systems 124a and 124b can capture the data packets from the network without interfering with the transmission of the data to the rest of the network. For example, the data collection system 124a receives data packets transmitted to the data center 120a from the load balancer 130 while at the same time the data packets are received by the server system 122a, creating a parallel path for receiving the data packets.

[0056] The data collection systems 124a and 124b can capture the data packets transmitted by the server systems 122a and 122b, respectively, back to the user. For example, the user 112 using a transmitting device 110 transmits a data packet requesting information. The data packet is routed through the load balancer 130 to data center B 120b. The data packet is transmitted to the server system 122b and the data collection system 124b. The server system 122b processes the data packet and responds to the user's request by transmitting a data packet response. The data packet response is captured by the data collection system 124b.

[0057] The data packet and/or data packet response are transmitted from the data collection systems 124a and 124b to the reconstruction engine 150. The reconstruction engine 150 receives data packets from the plurality of data centers 120a and 120b. The reconstruction engine 150 processes the data packets to form part or all of the data from the user 112. The reconstructed data is a representation of part or all of the information transmitted over the network by the user's transmitting device 110 (e.g., the raw information that is transmitted over the network).

[0058] The processing of the data packets by the server system 122a can occur at or near the same time and separately from the reconstruction engine 150 reconstructing the data. For example, the data packets are transmitted through the server system 122a for processing at or near the same time as

the data collection system 124a transmits the data packets to the reconstruction engine 150 and the reconstruction engine 150 is reconstructing the data packets into data. The processing by the server system 122a includes, for example, the transmission of the data packets received by the server system 122a to an intranet 125a which transmits the data packet to a web server 126a. The web server 126a processes the data packet to determine if the web server 126a needs to respond to the data packet and/or if the web server 126a needs to access information from the database server 127a. The web server 126a needs information from the database server 127a to respond to the data packet (e.g., a search submission to a search module that is part of the web server 126a). The web server 126a queries the database server 127a and the database server 127a responds to the query (e.g., the information that is requested from the search). The web server 126a processes the information returned from the query and transmits a web page with the requested information back to the transmitting device 110 of the user 112 through the intranet 125a.

[0059] The reconstruction engine 150 can analyze the data packets to determine data packet information. The data packet information includes a destination parameter, and/or an origination parameter. The data packet parameters can be used by the reconstruction engine 150 to match data packets together to form the data of the user 112. The data packet parameters includes a network address, a user address, an operating system (OS) fingerprint, a network card address, a user cookie, form data, an encryption key, and/or a transaction identifier. The network address includes the address of the transmitting device 110 transmitting the data packet, the address of the network address translation device (e.g., firewall) that transmits the data packet, or other network devices that transmit data packets. The user address includes the address of the user 112 at the transmitting device 110 transmitting the data packet, identifying information of the user's transmitting device 110, or other identifying information that is associated with the user 112. The OS fingerprint includes the formatting of the data that indicates the OS that transmits the data packet, identifying information in the data packet from the OS, or other identifying information that is associated with the OS. The network card address includes the address of the network card in the user's transmitting device 110 or other identifying information of the transmitting device that is associated with the transmitting device 110. The user cookie includes information stored on the user's transmitting device 110 (e.g., information stored in a web browser on the user's transmitting device 110) or other identifying information that is stored on the user's transmitting device 110. The form data includes information in the data packets associated with the user 112 (e.g., information in the data packet such as UserID=GeorgeSmith) or other identifying information in the data packets.

[0060] In some examples, the reconstruction engine 150 processes the data packet to determine if the data packet matches a request from the server systems 122a and 122b. For example, the server system 122a transmits a request for information (i.e., a data packet requesting information from a user 112) to a user 112 (e.g., login information—userid and password). The user 112 responds through the transmitting device 110 by transmitting a data packet (e.g., userid and password). The reconstruction engine 150 matches the request for information from the server system 122a to the data packet from the user 112 (e.g., the request for login information to the login information). The matching of the request for informa-

tion from the server system **122a** to the data packet from the user **112** is used by the reconstruction engine **150** to match data packets together.

[0061] The reconstruction engine **150** can process the data packets. The processing of the data packets includes filtering, decrypting, and/or encrypting (e.g., Radware® SSL Decryption available from Radware Ltd.). The filtering includes, for example, filtering by an internet protocol address, a protocol, a data type (e.g., graphic interface format (gifs), hypertext markup language (html), joint photographic experts group (jpegs), cascading style sheets (css), javascript (js)), a parameter (e.g., password, pin, personal identifying information), content, a content-type, an uniform resource locator (URL) path, and/or an internet protocol (IP) range.

[0062] The reconstruction engine **150** receives, for example, over one billion data packets per hour. For example, the reconstruction engine **150** can filter out data packets that are from a trusted internet protocol address (e.g., address 10.111.23.12 which is from a trusted data center). The reconstruction engine **150** filters out all of the data packets from the trusted internet protocol address to reduce the load on the reconstruction engine **150**. For example, the reconstruction engine **150** can filter out parameters, such as passwords and pin numbers, to protect sensitive information from being transmitted to the fraud detection system **160**.

[0063] The reconstruction engine **150** transmits the data to the fraud detection system **160**. The fraud detection system **160** processes the data to detect fraudulent activity. The processing of the data includes, for example, one or more systems that process the data (e.g., TIBCO® BusinessEvents® available from TIBCO Software Inc., Deep Network Analyzer (DNA) available from SourceForge®).

[0064] FIG. 2 is a functional block diagram of an exemplary system **200** depicting the data being transported from the data centers **120a** and **120b** to a data bus **210**. The system **200** includes the user **112** using the transmitting device **110** which transmits data in one or more parts as data packets to the load balancer **130**. The load balancer **130** transmits the data packets to a data center in the plurality of data centers **120a** and **120b** based on load balancing techniques. The data packets are transmitted to the server systems **122a** and **122b** and to the data collection systems **124a** and **124b**, respectively.

[0065] The data collection systems **124a** and **124b** transmit the data packets to the data bus **210**. In some examples, the data bus **210** aggregates the data packets from the plurality of data centers **120a** and **120b** for distribution to the systems that need and/or request the data such as the reconstruction engine **150**.

[0066] The data packets are transmitted from the data bus **210** to a database **220** and to the reconstruction engine **150**. The reconstruction engine **150** reconstructs data from the data packets received from the plurality of data centers **120a** and **120b**. The reconstructed data is transmitted to the fraud detection system **160** and the database **220**.

[0067] In some examples, the data bus **210** includes one or more computer systems connected to the network (e.g., a computing blade, a network server, a network router, a network switch, a network hub). In some examples, the database **220** stores data packets and/or data. The database **220** includes, for example, a memory module. The memory module includes, for example, persistent storage and/or volatile storage.

[0068] FIG. 3 is a functional block diagram of an exemplary system **300** showing the unification of user sessions from user requests. The system **300** includes the user **112** who has a user session **310**. The user session **310** is a compilation of all of the user requests associated with the user **112**. The user **112** is requesting and receiving information from the system **300**. This information requesting and receiving is accomplished through user requests.

[0069] The user requests are transmitted to the load balancer **130**. The load balancer **130** transmits a user request to a data center **120a** or **120b** based on load balancing techniques. The user request is transmitted to the server system **122a** or **122b** and the data collection system **124a** or **124b** based on the data center (e.g., **120a** or **120b**) to which the user request is sent. The data collection system **124a** or **124b** transmit the user request to a data bus **210**. The user request is transmitted from the data bus **210** to the database **220** and an unification engine **320**. The unification engine **320** combines all of the received user requests from the user **112** to form an unified user session which comprises all or part of the user session **310**, which was initially split apart for transmission to the data centers **120a** and **120b**. The unified user session can, for example, include a representation of part or all of the user's activity including interactions that are not transmitted through the transmitting device **110** over the network (e.g., information associated with applications on the transmitting device such as an application that interfaces with the server systems **122a** and **122b**).

[0070] The unification engine **320** can analyze the user requests to determine user request information. The user request information includes an origination parameter and/or a destination parameter. One or more of the user request parameters are, for example, used by the unification engine **320** to match user requests together to form the unified user session of the user **112**.

[0071] The unification engine **320** can analyze the user requests to determine if the user request matches a request for information from the server systems **122a** and **122b** and/or a request for information from an application module associated with the user's transmitting device **110** (e.g., client side application that communicates with the server systems **122a** and **122b**). For example, the application module associated with the user's transmitting device **110** requests information from the user **112**. The user **112** inputs the information into the application module. The application module through the transmitting device **110** transmits the information in the form of an user request to the server system **122a** or **122b** for processing. The unification engine **320** matches the user request with information that is requested by the application module (i.e., the unification engine **320** is aware of what information is requested by application modules so that it can match the information to user requests). The matching of the request for information from the application module to the user request from the user **112** is used by the unification engine **320** to match user requests together to form user sessions.

[0072] The unification engine **320** can process the user requests. The processing of the user requests includes, for example, filtering, decrypting, and/or encrypting.

[0073] The user session unified by the unification engine **320** is transmitted to the database **220** for storage and to the fraud detection system **160**. If the fraud detection system **160** detects fraudulent activity, then the fraud detection system **160** communicates to the interdiction system **330**. The inter-

diction system 330, for example, stops the communication of the user requests to the load balancer 130.

[0074] FIG. 4 is a functional block diagram of an exemplary system 400 depicting parts of a fraud detection system 405 including a database 450 of user sessions. Parts of the exemplary system 400 are described above in FIGS. 1, 2, and 3.

[0075] The unification engine 320 unifies the user requests and transmits the unified user session to the fraud detection system 405. The fraud detection system 405 includes a geolocation profiling engine 410, a behavior profiling engine 420, an application rules engine 430, a transactional rules engine 440, and a database 450. The fraud detection system 405 can, for example, include one of the engines or combinations of the engines.

[0076] The geolocation profiling engine 410 analyzes the user location, the network path, and/or the network address. For example, the geolocation profiling engine 410 analyzes the network address of the transmitting device 110 to determine if the network address is one that is known as a suspicious address (e.g., a network address from a foreign country that is known to have a high amount of fraudulent activity). The determination that the network address is suspicious includes all or part of the information analyzed by the fraud detection system 405 to determine whether the activity is fraudulent. For example, the geolocation profiling engine 410 analyzes the network path of the user 112 to determine if the pathway that the user 112 is accessing travels through or is intercepted by a suspicious network. If the network path of the user 112 is being intercepted by a suspicious network, the geolocation profiling engine 410 reports the fraudulent activity to the fraud detection system 405.

[0077] The behavior profiling engine 420 analyzes behavior by analyzing how many page views without a logon occur, visits with malicious software, search terms that are categorized as suspect, uniform resource locators (URL) that are categorized as suspect, account access that is categorized as rapid access, access into multiple accounts, multiple network connections per session, and/or multiple users per session.

[0078] For example, if the user 112 accesses ten different accounts which are not interconnected, then the behavior profiling engine 420 determines that the user activity is fraudulent and reports the fraudulent activity to the fraud detection system 405. However, if the user 112 accesses ten different accounts but the accounts are all interconnected (e.g., all have the same mailing address, the same unique user identifier), then the behavior profiling engine 420 determines that the user activity is not fraudulent.

[0079] The behavior profiling engine 420 includes an anomaly-based profiling engine. The anomaly-based profiling engine uses a baseline of normal activity to detect when activity falls outside of the baseline. The baseline of normal activity can, for example, be programmed into the profiling engine, and/or learned by the profiling engine while processing the information, data, and/or user sessions on the network. The activity includes any event, state, content, behavior, transaction, and/or similar processing by the user 112 with the network 400.

[0080] For example, the user 112 changes her email address on the exemplary system 400 from an internet service provider (e.g., standarduser at standardisp.com) that the user had been using for ten years to an email address at a generic easily obtained email hosting service (e.g., genericuser at hotmail.com). This change of the user's email address may be normal

behavior for some users but for this particular user 112, the change in email addresses falls outside of the normal behavior and the behavior profiling engine 420 classifies the activity as fraudulent activity. The classification of the activity as fraudulent activity requires, for example, the user 112 to confirm the change of email addresses outside of the communication channel in which the email change was submitted (e.g., if the email address change was through a website associated with the system 400, then the email address change confirmation outside of the communication channel is a phone call from the system 400).

[0081] The application rules engine 430 monitors the interactions between the modules of the exemplary system 400. The application rules engine 430 monitors, for example, a login module, a search module, a transaction module, an information module, and/or other modules that are part of the exemplary system 400. The application rules engine 430 monitors the modules of the exemplary system 400 for the detection of an interaction that is outside of the normal range of interactions with the module (e.g., violates a rule in the engine). The monitoring of the modules allows for the fraud detection system 405 to detect patterns of fraud that may not be detected by just analyzing the transaction of a user, but may be detected by analyzing the interactions with the applications of the exemplary system.

[0082] For example, a set of users is attempting to exploit a security hole in the authentication module. The application rules engine 430 is monitoring the authentication module and detects that the set of users is attempting to authenticate to the system 400 using the same username and password (e.g., a buffer overflow in the username and/or password fields). The normal range of interactions is not for more than one user to attempt to authenticate to the system 400 using the same username and password. Accordingly, the application rules engine 430 determines that the activity of the set of users is fraudulent.

[0083] The transactional rules engine 440 assigns a confidence level to the user session. The confidence level is an analysis to determine the level at which the system 400 ranks the user session. For example, if the user 112 is transmitting from a location that is within the user's profile (e.g., the user's home computer), but the transaction by the user 112 is outside of the user's profile (e.g., transferring all of the money in a money market account to an overseas bank account when the user never transferred money to an overseas bank account), then the confidence level of the user session is assigned a low number. This low number is utilized by the transactional rules engine 440 and/or the fraud detection system 405 to determine whether the activity is fraudulent.

[0084] However, for example, if the user 112 is not coming from a location that is within the user's profile (e.g., a hotel in New York City when the user never logged in from New York City before), but the transaction by the user 112 is within the user's profile (e.g., transferring all of the money in a money market account to an overseas bank account when the user transfers money to an overseas bank account every other week) and the computer signature of the computer that the user is using matches the computer signature in the user's profile (e.g., the user is using her laptop normally used at home at the hotel in New York City), then the confidence level of the user session is a higher number since the transaction is within the user's normal behavior although the user is logging in from a location outside of the user's profile. This high

number is utilized by the transactional rules engine **440** and/or the fraud detection system **405** to determine whether the activity is fraudulent.

[0085] The transactional rules engine **440** includes a signature-based rules engine. The signature-based rules engine searches for pre-defined patterns (e.g., signatures) in the user session. The patterns include, for example, a set of conditions that characterize fraudulent activities. For example, http headers and/or http payload content are analyzed to detect variable overwriting, parameter overloading, and/or other types of activity that indicate fraudulent activity may be occurring.

[0086] For example, the user **112** is submitting a request to transfer \$1,000,000 from an account to an overseas bank account. The transactional rules engine **440** is monitoring the activity and examines the http payload content to ensure that the content corresponds with the information of the activity. The http payload content contains userid=GeorgeRich while the user is logged on as userid=GeorgeFraud. The user **112** is attempting to transfer funds not from his account but from another user's account by changing the userid parameter in the http payload content. The transactional rules engine **440** detects this discrepancy and classifies the user session as fraudulent.

[0087] The database **450** stores elements of the user sessions in a memory module. The elements include, for example, user identification, network address, time, network path, length of time logged into the system, transactions, and/or other information contained in the user session. The memory module is, for example, volatile memory to speed the access of the stored user sessions. The memory module is, for example, persistent storage to keep for historical analysis (e.g., Oracle® Berkeley DB available from Oracle Corporation).

[0088] For example, the fraud detection system **405** searches the user sessions stored in the database **450** for access by the same network address and/or network subnet. If elements of the user session resemble and/or matches elements stored in the database **450** that have been associated with fraudulent activity, then the fraud detection system **405** classifies the user session as fraudulent (e.g., stored element is that subnet 10.10.10.0 is associated with fraudulent activity, then the user session from 10.10.10.5 is classified as fraudulent).

[0089] For example, the database **450** stores all of the transaction amounts for all bond purchases in volatile memory for fast access. The fraud detection system **405** receives a user session from the unification engine **320**. The user session is processed by the fraud detection system **405** to determine if fraudulent activity exists. Part of the processing by the fraud detection system **405** is comparing the transaction amount in the user session to previous transaction amounts for the bond that is being purchased. The stored elements in the database **450** for the bond being purchased includes ten years of transactions with over a hundred thousand transactions per year. The fraud detection system **405** retrieves from the database **450** in volatile memory the over one million transactions to analyze whether the transaction in the user session is fraudulent activity. The fraud detection system **405** utilizes the speed of the database **450** in volatile memory (e.g., random access memory) to process the transaction at or near the same time that the server system **122a** or **122b** is processing the transaction. If the transaction in the user session (e.g., purchase of 9,232 bonds) is within the normal ranges (e.g., the range of

purchases is 1 to 10,342), then the fraud detection system **405** does not interdict the transaction. If the transaction in the user session (e.g., the purchase of 132,230 bonds) is not within the normal range (e.g., the range of purchases is 1 to 10,342), then the fraud detection system **405** interdict the transaction.

[0090] FIG. **5** is a functional block diagram of an exemplary system **500** showing requestor systems that receive data from the data bus **210**. Parts of the exemplary system **500** are described above in association with FIGS. **1**, **2**, **3**, and **4**. The data bus **210** transmits the data packets to the requestor system. In some examples, the requestor system includes a marketing analytical system **562**, a network intrusion detection system **564**, a customer service system **566**, and/or a performance analysis system **568**.

[0091] The data packets received by the data bus **210** are, for example, reconstructed into data before being transmitted to the requestor system. The user requests received by the data bus **210** are, for example, unified into user sessions before being transmitted to the requestor system. The data bus **210** includes, for example, the unification engine **320** as shown in FIG. **3** and/or the reconstruction engine **150** as shown in FIG. **1**.

[0092] The data bus **210** receives the data packets and/or the user request as encrypted information from the data collection systems **124a** and **124b**. The data bus **210** decrypts the information before transmission to the requestor system.

[0093] The data bus **210** processes the data packet and/or the user request into a format that conforms with a particular protocol. The particular protocol includes a voice over internet protocol (VoIP), a transmission control protocol (TCP), an internet protocol (IP), an extensible markup language (XML), a hypertext markup language (HTML), and/or a standard generalized markup language (SGML).

[0094] FIG. **6A** is a screen shot **600a** generated by a login module. The screen shot **600a** shows the login screen generated by the login module that is included in the network associated with the load balancer **130** of FIG. **1**. The login screen includes a customer identification field **602a**, a pin field **604a**, and an information submission button **606a**. The fields **602a**, **604a**, and **606a** are part or all of the information in the user request.

[0095] FIG. **6B** is a diagram **600b** of information sent from the transmitting device **110** of FIG. **1** to the login module that includes the information in FIG. **6A**. The diagram **600b** is part or all of the data packet and/or the user request. The information parameters **610b** define how the information is routed and/or processed. The information parameters **610b** include the source of the information, the destination of the information, routing information, the protocol for the information, and/or other types of transmitting parameters. The information data **612b** includes the content information. The information data **612b** includes formatting information, content information, transaction information, and/or other types of content information. The transaction information can comprise the customer id **602b** which corresponds with the customer id field of the login screen **602a**. The transaction information can comprise the pin **604b** which corresponds with the pin field of the login screen **604a**. The transaction information can comprise the login in **606b** command information which corresponds with the log in submission button **606a**. The fields **602b**, **604b**, and **606b** are part or all of the information in the data packet and/or the information in the user request.

[0096] The exemplary system **300** of FIG. **3** unifies the user requests as illustrated by **600a** and **600b** into the user session.

The user session is analyzed by the fraud detection system **160** to look for account access into multiple accounts. The fraud detection system **160** detects that the user **112** logging into the account previously logged into multiple other accounts in or near the same timeframe as the current login. The fraud detection system **160** determines that the access to the account is fraudulent and transmits the fraudulent activity information for the user **112** to the interdiction system **330**. The interdiction system **330** cuts off the user **112** from accessing the system **300**.

[0097] FIG. 6C is a screen shot **600c** generated by a search module in the server system **122a** or **122b** of FIG. 3. The fields in the screen shot **600c** can comprise part or all of the information in the user request.

[0098] FIG. 6D is a diagram **600d** of information transmitted from the transmitting device **110** of FIG. 3 to the search module in the server system **122a** or **122b** that includes the information in FIG. 6C. The diagram **600d** is part or all of the data packet and/or the user request. The information parameters **610d** define how the information is routed and processed. The information data **612d** includes the content information.

[0099] FIG. 6E is a diagram **600e** of information received from the search module in the server system **122a** or **122b** of FIG. 3. The diagram **600e** is part or all of the data packet and/or the user request. The information parameters **610e** define how the information is routed and processed. The information data **612e** includes the content information.

[0100] FIG. 6F is a screen shot **600f** of information generated by the search module in the server system **122a** or **122b** of FIG. 3. The screen shot **600f** shows the information received from a search module. The information shown in the screen shot **600f** corresponds with the information **612e** in the diagram of FIG. 6E.

[0101] FIG. 6G is a diagram **600g** of information transmitted to a transaction module in the server system **122a** or **122b** of FIG. 3. The diagram **600g** is part or all of the data packet and/or the user request. The information parameters **610g** define how the information is routed and processed. The information data **612g** includes the content information.

[0102] FIG. 6H is a diagram **600h** of information received from an information module in the server system **122a** or **122b** of FIG. 3. The diagram **600h** is part or all of the data packet and/or the user request. The information parameters **610h** define how the information is routed and processed. The information data **612h** includes the content information.

[0103] FIG. 6I is a screen shot **600i** of information generated by the information module in the server system **122a** or **122b** of FIG. 3. The information shown in the screen shot **600i** corresponds with the information **612h** in the diagram of FIG. 6H.

[0104] The diagrams of information **600b**, **600d**, **600e**, **600g**, and **600h** represent data packets and/or user requests collected at the data collection system **124a** or **124b** of FIG. 1. The reconstructed data is a combination of all of the data packets collected at the data collection systems **124a** and **124b**. The reconstructed data is the information that is transmitted from the transmitting device **110** and from the server systems **122a** and **122b** (e.g., all of the raw packets that are transmitted over the system **100**). The information included in the screenshots **600a**, **600c**, **600f**, and **600i** and the diagrams of information **600b**, **600d**, **600e**, **600g**, and **600h** represent part or all of the user session. The user session is a

representative of part or all of the user's activity including interactions that are not transmitted through the transmitting device **110** over the network.

[0105] For example, the transmitting device **110** includes an application for interacting with the server systems **122a** and **122b**. The interaction between the user **112** and the application on the transmitting device **110** is part of the user's activity and thus the user session includes the interactions between the application on the transmitting device **110** and the user **112**. The unification engine **320** is aware of the applications that are used on the transmitting device **110** (e.g., by keeping information about the applications and the interactions with the applications in the database **220**) and integrates the interactions associated with the applications into the unified user session.

[0106] FIG. 7 is a flowchart **700** illustrating a process of the transmission of data through the exemplary system **100** of FIG. 1. The user **112** transmits (**710**) the data packet using the transmitting device **110**. The load balancer **130** receives the data packet and sends (**720**) the data packet to one of the data centers, **120a** or **120b**. FIG. 7 includes the processing performed at data center A **120a**, illustrated as grouping **730a** and the processing performed at data center B **120b**, illustrated as grouping **730b**. At either data center, the data packet is routed (**732a** and **732b**) to the server systems **122a** and **122b**, respectively, and the data collection systems **124a** and **124b** capture (**736a** and **736b**), respectively, the data packets. The data packet is processed (**734a** and **734b**) by the server systems **122a** and **122b**, respectively. A response from the server systems **122a** and **122b** is routed back to the user and is captured (**736a** and **736b**) by the data collection systems **124a** and **124b**, respectively. The data collection systems **124a** and **124b** send (**738a** and **738b**) the data packets to the reconstruction engine **150**. The reconstruction engine **150** receives the data packets from the data centers **120a** and **120b**. The reconstruction engine **150** reconstructs (**750**) the data from the data packets.

[0107] The reconstruction (**750**) of data packets can occur in several different ways. For example, the reconstruction (**750**) can include matching origination parameters between the data packets. Using, for example, the exemplary system **100** of FIG. 1 and the parts of the data of FIGS. 6A-6I, the reconstruction engine **150** reconstructs data packets into data. The data packets **600b**, **600d**, and **600g** show the information parameters **610b**, **610d**, and **610g**. The information parameters **610b**, **610d**, and **610g** include the origination parameters (e.g., the source field). The source field shows that the data packets are originating from address 192.168.0.1. The origination field can be matched between the data packets by the reconstruction engine **150**. The matched data packets are transformed into hypertext markup language (HTML) using the information data **612b**, **612d**, and **612g** which contains HTML tags to form part or all of the data.

[0108] An advantage of the exemplary system **100** is that the data packets **600b**, **600d**, and **600g** are routed to different data centers **130a** and **130b** by the load balancer **130** for processing (**734a** and **734b**) by the server systems **122a** and **122b** and the data packets **600b**, **600d**, and **600g** routed to the different data centers **130a** and **130b** are reconstructed into part or all of the data transmitted from and received by the user's transmitting device **110**. The data packet **600b** is routed (**720**) to data center A **120a** by the load balancer **130** while the data packets **600d** and **600g** are routed (**720**) to data center B **120b** by the load balancer **130**. The data packet **600b** is routed

(732a) to the server system 122a and captured (736a) by the data collection system 124a in data center A 120a. The data packets 600d and 600g are routed (732b) to the server system 122b and captured (736b) by the data collection system 124b in data center B 120b. The data collection systems 124a and 124b send (738a and 738b) the data packets to the reconstruction engine 150. The reconstruction engine 150 reconstructs (750) the data packets received from data centers A 120a and B 120b into the data that the user 112 sent. Thus, the data packets are split among the data centers 120a and 120b for processing (734a and 734b) by the server systems 122a and 122b, but are reconstructed (750) by the reconstruction engine 150.

[0109] For example, the reconstruction (750) can include matching origination and destination parameters between the data packets. Using, for example, the exemplary system 100 of FIG. 1 and the parts of the data of FIGS. 6A-6I, the reconstruction engine 150 reconstructs data packets into data. The data packets 600b, 600d, 600e, 600g, and 600h show the information parameters 610b, 610d, 610e, 610g, and 610h. The information parameters 610b, 610d, 610e, 610g, and 610h include the origination parameters (e.g., the source field, the destination field).

[0110] The source field shows that the data packets as illustrated in FIGS. 6B, 6D, and 6G are originating from address 192.168.0.1. The destination field shows that the data packets 600e and 600h are sent to address 192.168.0.1. The origination field and destination fields are matched between the data packets by the reconstruction engine 150 to form the data. The matching is part or all of the reconstruction (750) of the data.

[0111] An advantage of the exemplary system 100 is that the data packets 600b, 600d, and 600g are routed to different data centers 130a and 130b by the load balancer 130 for processing (734a and 734b) by the server systems 122a and 122b and the data packets 600b, 600d, and 600g routed through the different data centers 130a and 130b and the data packets transmitted by the server systems 122a and 122b are reconstructed into the data transmitted from and received by the user's transmitting device 110. The data packet 600b is routed (720) to data center A 120a by the load balancer 130 while the data packets 600d and 600g are routed (720) to data center B 120b by the load balancer 130. The data packet 600b is routed (732a) to the server system 122a and captured (736a) by the data collection system 124a in data center A 120a. The data packets 600d and 600g are routed (732b) to the server system 122b and captured (736b) by the data collection system 124b in data center B 120b. The data collection systems 124a and 124b capture (736a and 736b) the data packets 600e and 600h that are responses to the user 112. The data collection systems 124a and 124b send (738a and 738b) the data packets to the reconstruction engine 150. The reconstruction engine 150 reconstructs (750) the data packets received from data centers A 120a and B 120b into the data that the user 112 sent and received. Thus, the data packets are split among the data centers 120a and 120b for processing (734a and 734b) by the server systems 122a and 122b, but are reconstructed (750) by the reconstruction engine 150.

[0112] FIG. 8 is a flowchart 800 illustrating a process of the transmission of user requests through the exemplary system 300 of FIG. 3. The user 112 transmits (810) the user request using the transmitting device 110. The load balancer 130 receives the user request and sends (820) the user request to one of the data centers, 120a or 120b. FIG. 8 includes the processing performed at data center A 120a, illustrated as

grouping 830a and the processing performed at data center B 120b, illustrated as grouping 830b. At either data center, the user request is received (832a or 832b) at the data center 120a or 120b. The user request is sent (834a or 834b) to the server system 122a or 122b and to the data collection system 124a or 124b, respectively. The data collection systems send (836a or 836b) the user request to the data bus 210. The unification engine 320 receives (850) the user request from the data bus 210. The unification engine 320 unifies (860) the user requests received from the data bus 210 to form the user session.

[0113] The unification (860) of user requests can occur in several different ways. For example, the unification (860) includes matching origination and destination parameters between the user requests. Using, for example, the exemplary system 300 of FIG. 3 and the parts of the user session of FIGS. 6A-6I, the unification engine 320 unifies (860) user requests into user sessions. The user session includes the user requests 600b, 600d, and 600g, responses 600e and 600h, and the information for the screenshots 600a, 600c, 600f, and 600i. The user requests 600b, 600d, and 600g and responses 600e and 600h show the information parameters 610b, 610d, 610e, 610g, and 610h. The information parameters 610b, 610d, 610e, 610g, and 610h include the origination parameters (e.g., the source field, the destination field). The source field shows that the user requests 600b, 600d, and 600g are originating from address 192.168.0.1. The destination field shows that the user requests 600e and 600h are being sent to address 192.168.0.1. The information data 612b, 612d, 612e, 612g, and 612h provides part or all of the information for the screenshots 600a, 600c, 600f, and 600i that are part or all of the user session. The origination field and destination field can be matched between the user requests by the unification engine 320 to form the data. Based on the information stored in the database 220, the unification engine 320 can match the information that is in the screenshots to the user requests (e.g., the unification engine can match a user request containing login information to the screenshot 600a of FIG. 6 generated by the login module). The matching of the fields and the matching of the information in the screenshots are part or all of the unification (860) of the user session.

[0114] An advantage of the exemplary system 300 is that the user requests 600b, 600d, and 600g which are routed to different data centers 130a and 130b by the load balancer 130 are unified with the user requests 600e and 600h being sent to the user 112 and the information contained in the screenshots 600a, 600c, 600f, and 600i.

[0115] FIG. 9 is a flowchart 900 illustrating a process of the transmission of user requests through the exemplary system 300 of FIG. 3. The user 112 using a transmitting device 110 transmits (910) the user request. The user request is sent (920) by the load balancer 130 to the data center 120b. Data center B 120b is used in this example, however the load balancer can send (920) the user request to data center A 120a. The request is routed (932) to the server system 122b. The server system 122b processes (934) the request. Server system 122b responds (936) to the user request. The response is routed back to the user 112 through her transmitting device 110 and is captured (942) by the data collection system 124b. The data collection system 124b captures (942) the user request at or near the same time as the user request is routed (932) to the server system 122b. The data collection system 124b captures (942) the user request without affecting the processing (934) of the user request by the server system 122b. After capturing

(942) the user request, the data collection system 124b sends (944) the user request to the unification engine 320. The unification engine 320 unifies (946) the user request to form the user session. The unification engine 320 unifies (946) the user request by itself or with other user requests to form the user session. The user session is processed (948) by the fraud detection system 160 to detect fraudulent activity.

[0116] Using, for example, the exemplary system 400 of FIG. 4 and the parts of the user session of FIGS. 6A-6I, the fraud detection system 405 processes (948) user sessions to determine if the activity is fraudulent. The fraud detection system 405 uses the geolocation profiling engine 410, the behavior profiling engine 420, and the transactional rules engine 440 to process (948) the user session that includes the buy order in the user request 600g (i.e., 1,000,000 bonds of the fund). The geolocation profiling engine 410 processes (948) the location that the user 112 is transmitting the buy order from. The location is not suspect (e.g., the network address and network subnet are not on a list of suspect networks) based on the network address and network subnet. The geolocation profiling engine 410 generates a report of non-fraudulent activity. The behavior profiling engine 420 analyzes the buy order to determine whether the buy order is outside of the range of the behavior of the user's profile. The behavior profiling engine 420 determines that the buy order is not outside of the range of the behavior of the user 112 because by accessing the stored elements in the database 450, the behavior profiling engine 420 determines that the user 112 has bought and sold millions of shares of bonds over the past ten years. The behavior profiling engine 420 generates a report of non-fraudulent activity. The transactional rules engine 440 analyzes the buy order to determine whether the buy order is outside of the normal range of buy and sell orders for the particular bond. The transactional rules engine 440 determines that the buying and selling of one million bonds is within the normal range for the particular bond (e.g., the bond averages one hundred million buy and sell transactions per week). The transactional rules engine 440 generates a report of non-fraudulent activity. Based on the reports of the geolocation profiling engine 410, the behavior profiling engine 420, and the transactional rules engine 440, the fraud detection system 405 determines that the user session is not fraudulent.

[0117] FIG. 10 is a flowchart 1000 illustrating a process of the transmission of data packets through the exemplary system 500 of FIG. 5. The user 112 using a transmitting device 110 transmits (1010) the data packet. The data packet is sent (1020) by the load balancer 130 to the data center B 120b. Data center B 120b is used in this example, however the load balancer can send (1020) the user request to data center A 120a. The data packet is routed (1032) to the server system 122b. The server system 122b processes (1034) the data packet. Server system 122b responds (1036) to the data packet. The response is routed back to the user 112 through her transmitting device 110 and is captured (1042) by the data collection system 124b. The data collection system 124b captures (1042) the data packet at or near the same as the data packet is routed (1032) to the server system 122b. The data collection system 124b captures (1042) the data packet without affecting the processing (1034) of the data packet by the server system 122b. After capturing (1042) the data packet, the data collection system 124b sends (1044) the data packet to the data bus 210. The requestor system requests (1046) part

of the data from the data bus 210. The requested data is transmitted (1048) from the data bus 210 to the requestor system.

[0118] Using, for example, the exemplary system 500 of FIG. 5 and the parts of the data of FIGS. 6A-6I, the requestor system (e.g., 562, 564, 566, and/or 568) requests (1046) part of the data from the data bus 210. For this example, the requestor system is the performance analysis system 568. The performance analysis system 568 analyzes data packets to determine the response time of the search module. The performance analysis system 568 requests (1046) the data that is sent to and from the search module. The data bus 210 transmits (1048) the requested parts of the data to the performance analysis system 568. The parts of the data are the data packets 600d and 600e which represent the parts of the data sent to and from the search module. The performance analysis system 568 analyzes the response time between the data packet requesting the search 600d and the data packet that transmitted the search results 600e. A high response time indicates that the system 500 is loaded and the performance analysis system 568 makes changes to the configuration of the load balancer 130 and/or the server systems 122a and 122b to decrease the response time. A low response time indicates that the system 500 is responding in an appropriate timeframe to the user request and no corrective action is needed. The performance analysis system 568 monitors the modules of the system 500 and makes changes to the system 500 to decrease response times when appropriate.

[0119] For this example, the requestor system is the customer service system 566. The customer service system 566 is analyzing data packets to diagnose and fix problems that the user 112 is having with the system 500. The user 112 is attempting to search for information utilizing the information data 612d by submitting the information to the search module that is part of the system 500. The search module is not returning the correct information. The user 112 contacts a customer service representative who uses the customer service system 566 to look at the data for the user 112. The portion of the data is requested (1046) by the customer service system 566.

[0120] The customer service representative identifies that the information parameters 610d are not being transmitted to the system 500 correctly and therefore the information request is failing. The customer service representative instructs the user 112 to update her world wide web browser software (e.g., Mozilla® Firefox) and to try again. The user 112 updates her world wide web browser software and submits the data packet 600d to the system 500. The user 112 receives the data packet response 600e (i.e., response to the user's request) as illustrated by the screenshot 600f.

[0121] FIG. 11 is a flowchart 1100 illustrating a process of the transmission of user requests through the exemplary system 300 of FIG. 3. The user 112 using a transmitting device 110 transmits (1110) the user request. The interdiction system 330 determines (1164) whether the user session should be interdicted based on notifications from the fraud detection system 160. The fraud detection system 160 sends the notification to the interdiction system 330 when, for example, the user 112 is making transactions outside of the user's behavior profile (e.g., the user 112 is buying ten millions shares of stock when the user 112 only has one hundred shares of stock in her portfolio before this transaction).

[0122] If the interdiction system 330 does not interdict (1164) the user session, then the user request is routed (1120)

by the load balancer 130 to the data center 120b. If the interdiction system 330 does interdict (1164) the user session, then access to the system 1100 is stopped (1166). Data center B 120b is used in this example, however the load balancer can send (920) the user request to data center A 120a. The request is routed (1032) to the server system 122b. The server system 122b processes (1034) the request. Server system 122b responds (1036) to the user request. The response is routed back to the user 112 through her transmitting device 110 unless the interdiction system 330 determines (1162) that the user session should be interdicted based on notifications from the fraud detection system 160. If the interdiction system 330 does interdict (1162) the user session, then access to the system 1100 is stopped (1166) (i.e., the response to the user request is not sent back to the user 112).

[0123] The data collection system 124b captures (1142) the user request at or near the same as the user request is routed (1132) to the server system 122b. The data collection system 124b captures (1142) the user request without affecting the processing (1134) of the user request by the server system 122b. After capturing (1142) the user request, the data collection system 124b sends (1144) the user request to the data bus 210. The unification engine 320 receives (1146) the user request from the data bus 210. The unification engine 320 unifies (1148) the user request to form the user session. The unification engine 320 unifies (1148) the user request by itself or with other user requests to form the user session. The user session is processed (1050) by the fraud detection system 160 to detect fraudulent activity. If the fraud detection system 160 detects (1160) fraudulent activity, then the fraud detection system 160 notifies the interdiction system 330. If the fraud detection system 160 does not detect (1160) fraudulent activity, then the fraud detection system 160 continues processing (1150) user sessions looking for fraudulent activity.

[0124] Using, for example the exemplary system 400 of FIG. 4 and the parts of the user session of FIGS. 6A-6I, the interdiction system 330 determines (1162 and 1164) whether to interdict user requests and responses to user requests. The unification (1148) of user requests into user sessions by the unification engine 320 is described above. The user session is processed (1150) by the fraud detection system 405 to look for search terms that are categorized as suspect. The fraud detection system 405 uses the behavior profiling engine 420 to process (1150) the user session. The behavior profiling engine 420 processes (1150) the search terms in the information data 612d to determine if the search terms are categorized by the behavior profiling engine 420 as suspect. The search terms for a high yield bond, fund performance of ten years or greater, and a Morningstar rating of five stars is not classified as a suspect search term by the behavior profiling engine 420 because the search is categorized as a long term investment search and not a fraudulent search. Therefore, the behavior profiling engine 420 will not classify the search by the user 112 as fraudulent activity and the fraud detection system 160 will not notify (1160) the interdiction system 330 since no fraudulent activity was detected.

[0125] FIG. 12 is a flowchart 1200 illustrating a process of the transmission of user requests through the exemplary system 400 of FIG. 4. The process of the transmission of user requests is similar in parts to FIG. 11 as described above. The user 112 using a transmitting device 110 transmits (1110) the user request. The user request is sent to the authentication system that authenticates (1270) the user 112 to verify that only user requests from a properly authenticated user 112 are

allowed to access the system 1200. If the user 112 is granted (1268) access to the system 1200, then the user request is transmitted to the load balancer 130. If the user 112 is not granted (1268) access to the system 1200, then the user access is stopped (1166).

[0126] In some examples, the authentication system includes a computer, a network hub, a network switch, a network router, a network firewall, an authentication server (e.g., Kerberos authentication server in Windows® Server 2003 available from Microsoft Corporation, Oracle® Access Manager available from Oracle Corporation), and/or other authentication modules.

[0127] FIG. 13 is a flowchart 1300 illustrating a process of the transmission of user requests through the exemplary system 400 of FIG. 4. The process of the transmission of user requests is similar in parts to FIGS. 11 and 12 as described above. If the fraud detection system 405 detects (1160) fraudulent activity, then the interdiction system 330 generates (1382) an authentication rule based on the fraudulent activity. The authentication rule is communicated (1384) to the authentication system. When the user 112 transmits (1110) the user request, then the authentication system will have the rule when authenticating (1270) the user 112. In some examples, the rule includes the identification of a certain location that is not allowed access to the exemplary system, the identification of a certain user account that is not allowed access to the exemplary system, and/or the identification of a certain user account that is not allowed access to the exemplary system for a set amount of time.

[0128] FIG. 14 is a flowchart 1400 illustrating a process of the transmission of user requests through the exemplary system 400 of FIG. 4. The process of the transmission of user requests is similar in parts to FIGS. 11, 12, and 13 as described above. If the fraud detection system 405 detects (1160) fraudulent activity, then the interdiction system 330 interdicts (1162 and 1164) the user requests and the responses to the user requests for the user session in which fraud was detected. The interdiction system 330 then redirects (1470) the user requests and responses to the user requests to a separate network. In some examples, the separate network includes a honeypot server, a honeypot network, a server system, a network firewall, a network router, a network hub, a network switch, and/or other network communication devices. The separate network is, for example, used to allow the user 115 to continue the fraudulent activity without adversely affecting the data centers 120a and 120b. The ability to allow the user 115 to continue committing the fraudulent activity on the separate network allows for the analysis of the fraudulent activity for the addition or modification of rules and/or profiles for the fraud detection system 405.

[0129] The above-described systems and methods can be implemented in digital electronic circuitry, in computer hardware, firmware, and/or software. The implementation can be as a computer program product (i.e., a computer program tangibly embodied in an information carrier). The implementation can, for example, be in a machine-readable storage device and/or in a propagated signal, for execution by, or to control the operation of, data processing apparatus. The implementation can, for example, be a programmable processor, a computer, and/or multiple computers.

[0130] A computer program can be written in any form of programming language, including compiled and/or interpreted languages, and the computer program can be deployed in any form, including as a stand-alone program or as a

subroutine, element, and/or other unit suitable for use in a computing environment. A computer program can be deployed to be executed on one computer or on multiple computers at one site.

[0131] Method steps can be performed by one or more programmable processors executing a computer program to perform functions of the invention by operating on input data and generating output. Method steps can also be performed by and an apparatus can be implemented as special purpose logic circuitry. The circuitry can, for example, be a FPGA (field programmable gate array) and/or an ASIC (application-specific integrated circuit). Modules, subroutines, and software agents can refer to portions of the computer program, the processor, the special circuitry, software, and/or hardware that implements that functionality.

[0132] Processors suitable for the execution of a computer program include, by way of example, both general and special purpose microprocessors, and any one or more processors of any kind of digital computer. Generally, a processor receives instructions and data from a read-only memory or a random access memory or both. The essential elements of a computer are a processor for executing instructions and one or more memory devices for storing instructions and data. Generally, a computer can include, can be operatively coupled to receive data from and/or transfer data to one or more mass storage devices for storing data (e.g., magnetic, magneto-optical disks, or optical disks).

[0133] Data transmission and instructions can also occur over a communications network. Information carriers suitable for embodying computer program instructions and data include all forms of non-volatile memory, including by way of example semiconductor memory devices. The information carriers can, for example, be EPROM, EEPROM, flash memory devices, magnetic disks, internal hard disks, removable disks, magneto-optical disks, CD-ROM, and/or DVD-ROM disks. The processor and the memory can be supplemented by, and/or incorporated in special purpose logic circuitry.

[0134] To provide for interaction with a user, the above described techniques can be implemented on a computer having a display device. The display device can, for example, be a cathode ray tube (CRT) and/or a liquid crystal display (LCD) monitor. The interaction with a user can, for example, be a display of information to the user and a keyboard and a pointing device (e.g., a mouse or a trackball) by which the user can provide input to the computer (e.g., interact with a user interface element). Other kinds of devices can be used to provide for interaction with a user. Other devices can, for example, be feedback provided to the user in any form of sensory feedback (e.g., visual feedback, auditory feedback, or tactile feedback). Input from the user can, for example, be received in any form, including acoustic, speech, and/or tactile input.

[0135] The above described techniques can be implemented in a distributed computing system that includes a back-end component. The back-end component can, for example, be a data server, a middleware component, and/or an application server. The above described techniques can be implemented in a distributing computing system that includes a front-end component. The front-end component can, for example, be a client computer having a graphical user interface, a Web browser through which a user can interact with an example implementation, and/or other graphical user interfaces for a transmitting device. The components of the system

can be interconnected by any form or medium of digital data communication (e.g., a communication network). Examples of communication networks include a local area network (LAN), a wide area network (WAN), the Internet, wired networks, and/or wireless networks.

[0136] The system can include clients and servers. A client and a server are generally remote from each other and typically interact through a communication network. The relationship of client and server arises by virtue of computer programs running on the respective computers and having a client-server relationship to each other.

[0137] Comprise, include, and/or plural forms of each are open ended and include the listed parts and can include additional parts that are not listed. And/or is open ended and includes one or more of the listed parts and combinations of the listed parts.

[0138] One skilled in the art will realize the invention may be embodied in other specific forms without departing from the spirit or essential characteristics thereof. The foregoing embodiments are therefore to be considered in all respects illustrative rather than limiting of the invention described herein. Scope of the invention is thus indicated by the appended claims, rather than by the foregoing description, and all changes that come within the meaning and range of equivalency of the claims are therefore intended to be embraced therein.

1. A method for detecting fraudulent activity, the method comprising:

receiving, by a computing device in a second network, a plurality of interactions between a plurality of data centers in a first network and one or more devices, remote from the data centers, of a user, the plurality of interactions forming an aggregate user session representing the user's activity over a period of time with the plurality of data centers via the one or more devices;

monitoring, by the computing device, data transaction requests of the plurality of interactions, the computing device configured to respond to each data transaction request with a response that appears to the user as if the request was executed by the respective data center;

if the computing device determines that the interactions are harmful:

interdicting, by the computing device, the interactions from execution by the plurality of data centers; and allowing, by the computing device, the one or more devices to submit additional interactions;

if the computing device is unable to determine whether the subsequent interactions are harmful:

transmitting, by the computing device, the interactions to the plurality of data centers for execution; and marking, by the computing device, the interactions for monitoring purposes.

2. The method of claim 1, if the computing device determines that the interactions are harmful, the method further comprising:

if the computing device is able to simulate the interactions, transmitting a response for each interaction to the one or more devices, the transmitted responses appearing to the user as if the plurality of data centers executed the interactions;

if the computing device is unable to simulate the interactions, transmitting a response for each interaction to the one or more devices, the transmitted responses appear-

ing to the user as if the plurality of data centers failed to execute the interactions due to a system error; and monitoring, by the computing device, data transaction requests of additional interactions submitted from the one or more devices.

3. The method of claim 1, the method further comprising: generating, by the computing device, a signature based on the monitoring step; and correlating, by the computing device, the signature with previous user sessions to determine whether the plurality of interactions and the previous user sessions share one or more characteristics.

4. The method of claim 3, wherein the characteristics include at least one of: a common user, a common remote device, a common location, or a common activity pattern.

5. The method of claim 3, wherein the signature is generated using one or more data analysis tools that have access to the plurality of interactions.

6. The method of claim 1, wherein the method is performed in real time.

7. The method of claim 1, wherein the response that appears to the user as if the request was executed by the respective data center includes dummy data to prevent the user from accessing data stored in the data center.

8. The method of claim 1, if the computing device determines that the interactions are harmful, the method further comprising:

providing, by the computing device, responses to the additional interactions that encourage the user to stay connected to the computing device until an enforcement action is executed.

9. The method of claim 8, wherein the responses to the additional interactions extend a session between the computing device and the one or more remote devices.

10. The method of claim 1, if the computing device determines that the interactions are harmful, the method further comprising:

determining, by the computing device, one or more vulnerabilities associated with the plurality of data centers based on the interactions submitted by the one or more remote devices.

11. A system for detecting fraudulent activity, the method comprising:

a computing device in a second network configured to:

receive a plurality of interactions between a plurality of data centers in a first network and one or more devices, remote from the data centers, of a user, the plurality of interactions forming an aggregate user session representing the user's activity over a period of time with the plurality of data centers via the one or more devices;

monitor data transaction requests of the plurality of interactions, the computing device configured to respond to each data transaction request with a response that appears to the user as if the request was executed by the respective data center;

if the computing device determines that the interactions are harmful:

interdict the interactions from execution by the plurality of data centers; and

allow the one or more devices to submit additional interactions;

if the computing device is unable to determine whether the subsequent interactions are harmful:

transmit the interactions to the plurality of data centers for execution; and

mark the interactions for monitoring purposes.

12. The system of claim 11, if the computing device determines that the interactions are harmful, the computing device further configured to:

transmit a response for each interaction to the one or more devices if the computing device is able to simulate the interactions, the transmitted responses appearing to the user as if the plurality of data centers executed the interactions;

transmit a response for each interaction to the one or more devices if the computing device is unable to simulate the interactions, the transmitted responses appearing to the user as if the plurality of data centers failed to execute the interactions due to a system error; and monitor data transaction requests of additional interactions submitted from the one or more devices.

13. A computer program product, tangibly embodied in a non-transitory computer readable storage medium, for detecting fraudulent activity, the computer program product including instructions operable to cause a computing device in a second network to:

receive a plurality of interactions between a plurality of data centers in a first network and one or more devices, remote from the data centers, of a user, the plurality of interactions forming an aggregate user session representing the user's activity over a period of time with the plurality of data centers via the one or more devices;

monitor data transaction requests of the plurality of interactions, the computing device configured to respond to each data transaction request with a response that appears to the user as if the request was executed by the respective data center;

if the computing device determines that the interactions are harmful:

interdict the interactions from execution by the plurality of data centers; and

allow the one or more devices to submit additional interactions;

if the computing device is unable to determine whether the subsequent interactions are harmful:

transmit the interactions to the plurality of data centers for execution; and

mark the interactions for monitoring purposes.

14. The computer program product of claim 13, if the computing device determines that the interactions are harmful, the computer program product including further instructions operable to cause the computing device to:

transmit a response for each interaction to the one or more devices if the computing device is able to simulate the interactions, the transmitted responses appearing to the user as if the plurality of data centers executed the interactions;

transmit a response for each interaction to the one or more devices if the computing device is unable to simulate the interactions, the transmitted responses appearing to the user as if the plurality of data centers failed to execute the interactions due to a system error; and

monitor data transaction requests of additional interactions submitted from the one or more devices.