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

Canadian Intellectual Property Office
An agency of Industry Canada

CA 2673950 A1 2008/11/06

(21) 2 673 950

(12) DEMANDE DE BREVET CANADIEN
CANADIAN PATENT APPLICATION

(13) A1

(86) Date de dépôt PCT/PCT Filing Date: 2008/04/11
(87) Date publication PCT/PCT Publication Date: 2008/11/06
(85) Entrée phase nationale/National Entry: 2009/06/26
(86) N° demande PCT/PCT Application No.: EP 2008/054412
(87) N° publication PCT/PCT Publication No.: 2008/132036
(30) Priorité/Priority: 2007/04/27 (US11/741,516)

(51) Cl.Int./Int.Cl. *G06F 21/20* (2006.01)

(71) Demandeur/Applicant:
INTERNATIONAL BUSINESS MACHINES CORPORATION, US

(72) Inventeurs/Inventors:
HAMILTON, RICK ALLEN, II, US;
O'CONNELL, BRIAN MARSHALL, US;
PAVESI, JOHN, US;
WALKER, KEITH RAYMOND, US

(74) Agent: CHAN, BILL W.K.

(54) Titre : SYSTEME D'AUTHENTIFICATION EN CASCADE
(54) Title: CASCADING AUTHENTICATION SYSTEM

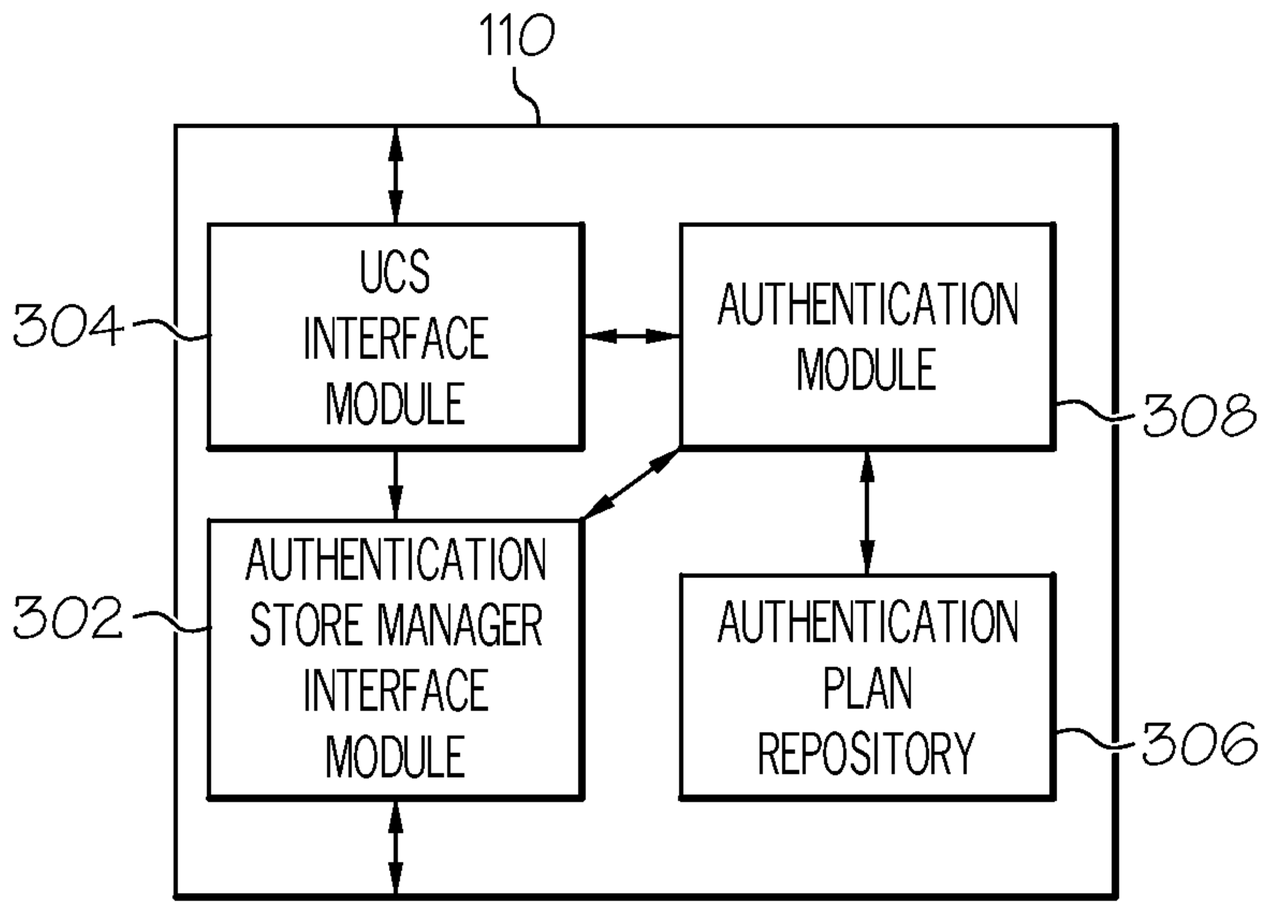

FIG. 3

(57) Abrégé/Abstract:
Generally speaking, systems, methods and media for authenticating a user to a server based on previous authentications to other serversare disclosed. Embodiments of amethod for authenticating a user to a servermay include receiving a request to authenticate the user to the server and determining whether authenticating the user requires matching an authentication plan.If a plan is required, the method may also include accessing a stored authentication plan with authentication records each having expected information relating to user access to a different server. The method may also include receiving an indication of the user s current authentication plan from an authentication store where the plan has authorization records each having current information relating to user access. Embodiments ofthe method may also include comparing the stored authentication plan with the received current authentication plan to determine whether they match and, in response to a match, authenticating the user.

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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
6 November 2008 (06.11.2008)

PCT

(10) International Publication Number
WO 2008/132036 A1

(51) International Patent Classification:
G06F 21/20 (2006.01)

(21) International Application Number:
PCT/EP2008/054412

(22) International Filing Date: 11 April 2008 (11.04.2008)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
11/741,516 27 April 2007 (27.04.2007) US

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

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

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **HAMILTON II, Rick, Allen** [US/US]; 1532 Dairy Road, Charlottesville, Virginia 22903 (US). **O'CONNELL, Brian, Marshall** [US/US]; 226 Mint Hill Drive, Cary, North Carolina 27519 (US). **PAVESI, John** [US/US]; 408 Trailridge Drive, Cedar Park, Texas 78613 (US). **WALKER, Keith, Raymond** [US/US]; 13412 Kinder Pass, Austin, Texas 78727 (US).

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

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

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

Published:
— *with international search report*

(54) Title: CASCADING AUTHENTICATION SYSTEM

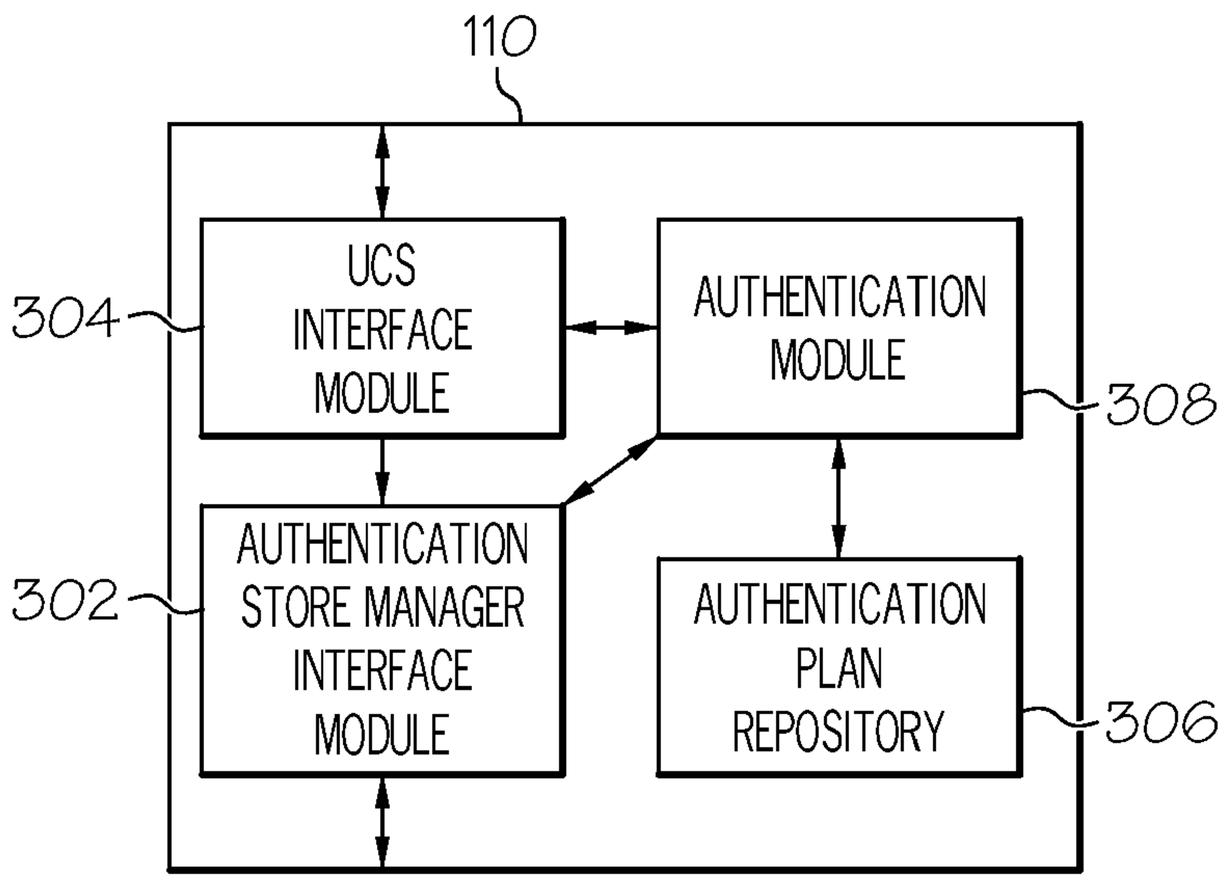

FIG. 3

(57) Abstract: Generally speaking, systems, methods and media for authenticating a user to a server based on previous authentications to other serversare disclosed. Embodiments of amethod for authenticating a user to a servermay include receiving a request to authenticate the user to the server and determining whether authenticating the user requires matching an authentication plan.If a plan is required, the method may also include accessing a stored authentication plan with authentication records each having expected information relating to user access to a different server. The method may also include receiving an indication of the user s current authentication plan from an authentication store where the plan has authorization records each having current information relating to user access. Embodiments ofthe method may also include comparing the stored authentication plan with the received current authentication plan to determine whether they match and, in response to a match, authenticating the user.

WO 2008/132036 A1

CASCADING AUTHENTICATION SYSTEM

FIELD OF INVENTION

The present invention is in the field of data processing systems and, in particular, to systems, methods and media for implementing a cascading authentication system for authenticating users to a server based on previous authentications to other servers by the user.

BACKGROUND OF THE INVENTION

Computer systems are well known in the art and have attained widespread use for providing computer power to many segments of today's modern society. As advances in semiconductor processing and computer architecture continue to push the performance of computer hardware higher, more sophisticated computer software has evolved to take advantage of the higher performance of the hardware, resulting in computer systems that continue to increase in complexity and power. Computer systems have thus evolved into extremely sophisticated devices that may be found in many different settings.

Many organizations utilize server computer systems for more complicated tasks such as providing e-commerce websites, providing complex multi-user applications, maintaining large databases, or performing other resource-intensive tasks. Organizations with significant computing needs often have many servers performing a wide variety of tasks with the servers communicating with each other via a network such as a local area network (LAN). In these systems, individual users may interact with the servers to access various system resources, such as applications, databases, or other resources, so that the systems resources may be shared by multiple users.

Users often arrive at their target server (*i.e.*, the software server to which they desire to gain access) by successfully navigating authentications at multiple levels. A user, for example, desiring to access a target server which is a database may have to first authenticate to their computer's operating system, next authenticate to a Virtual Private Network (VPN) from the Internet to access a corporate network, then authenticate to a firewall to access a lab, and lastly authentication with the database residing on a machine in the lab. Other authentication

steps are possible, such as establishing a remote control session to login to a remote machine, a remote shell session such as with SSH or Telnet, or other steps.

Such a system of cascading authentications, however, can result in security risks if a hacker can "skip" layers and begin their authentication attempt from as few layers from the target server as possible. If someone desires to masquerade as a particular user, for example, it is much easier to guess or obtain one set of credentials rather than multiple sets (assuming different credentials at each layer). It is accordingly typically easier to gain unauthorized access as an "insider" in part because there are fewer layers. In an illustrative example, a system with four layers of authentication can be assumed: an outer wall with a 95% chance of stopping a hacker, an inner firewall with a 93% chance, a secure system with a 90% chance, and application-level authentication with an 85% chance. The cumulative probability of making all the way from the outside to the application is one minus the chance of getting stopped at each point, cascaded through the system, resulting in a probability of (0.05)(0.07)(0.10)(0.15) = 0.0000525. In contrast, an insider in this example with direct access to the application would have a 15% chance (0.15) of penetrating the application as they avoid the previous levels of authentication.

System designers have attempted to solve the problem of hackers skipping levels of authentication by emulating an insider. One known solution is to allow authentication only from a defined IP or MAC address to limit access to the specified address. This solution, however, is often not practical, particularly when a VPN is involved. Moreover, this solution is insufficient when the authorized machine is shared, does not take full advantage of all of the authentication layers, and can be easily spoofed. Another known solution is to require additional authentication, such as a smart card or other device. This solution, however, requires significant infrastructure costs and adds to user inconvenience. Both of these problems are exacerbated if the user has to be authenticating multiple layers as a separate smart card would typically be required for each of the multiple layers.

DISCLOSURE OF THE INVENTION

The problems identified above are in large part addressed by systems, methods and media for authenticating a user to a server based on previous authentications to other servers.

Embodiments of a method for authenticating a user to a server may include receiving a request to authenticate the user to the server and determining whether authenticating the user requires matching an authentication plan. If a plan is required, the method may also include accessing a stored authentication plan with authentication records each having expected information relating to user access to a different, particular server at a previous layer of authentication than the target server. The method may also include receiving an indication of the user's current authentication plan from an authentication store where the plan has authorization records each having current information relating to user access to a particular, different server at a previous layer of authentication than the target server. Embodiments of the method may also include comparing the stored authentication plan with the received current authentication plan to determine whether they match and, in response to a match, authenticating the user.

Another embodiment provides a computer program product comprising a computer-useable medium having a computer readable program wherein the computer readable program, when executed on a computer, causes the computer to perform a series of operations for authenticating a user to a server. The series of operations generally includes receiving a request to authenticate the user to the server and determining whether authenticating the user requires matching an authentication plan. If a plan is required, the series of operations may also include accessing a stored authentication plan with authentication records each having expected information relating to user access to a different, particular server at a previous layer of authentication than the target server. Embodiments of the series of operations may also include receiving an indication of the user's current authentication plan from an authentication store where the plan has authorization records each having current information relating to user access to a particular, different server at a previous layer of authentication than the target server. Embodiments of the series of operations may also include comparing the stored authentication plan with the received current authentication plan to determine whether they match and, in response to a match, authenticating the user.

A further embodiment provides a cascading authentication system. The cascading authentication system may include a target server having an authentication plan manager to access a stored authentication plan associated with a user requesting access to the target

server, where the stored authentication plan includes one or more authentication records each having expected information relating to access by a user to a different, particular server at a previous layer of authentication than the target server. The cascading authentication system may also include an authentication store to store a current authentication plan associated with the user, where the current authentication plan includes one or more authentication records each having current information relating to access by a user to a different, particular server at a previous layer of authentication than the target server. Embodiments of the cascading authentication system may also include an authentication store manager to provide the current authentication plan associated with a particular user to the authentication plan manager of the target server, where the authentication plan manager of the target server determines whether to authenticate a user based on a comparison between the stored authentication plan for the user and the current authentication plan for the user.

Another embodiment provides a method for authenticating a user to a target server. Embodiments of the method may include performing an authentication step for one or more servers at a previous layer of authentication to the target server and storing an authentication event record for each performed authentication step in an authentication store. Embodiments of the method may also include attempting to authenticate to the target server, where the target server requires an authentication plan associated with the user. Embodiments of the method may also include receiving an indication of whether access to the target server was granted.

Preferably, each authentication record of the authentication plans comprises a server identifier and an authentication event fact. More preferably, the authentication store is an encrypted database. Still more preferably, the authentication store manager executes in non-volatile memory of a user computer system of the user requesting access to the target server. Still more preferably, the authentication store manager executes on a trusted third party computer system. Still more preferably, in response to receiving an indication that access to the target server was not granted, an authentication plan mismatch with the target server is resolved.

BRIEF DESCRIPTION OF THE DRAWINGS

Aspects of certain embodiments of the invention will become apparent upon reading the following detailed description and upon reference to the accompanying drawings in which like references may indicate similar elements:

FIG. 1 depicts an environment for a cascading authentication system with a user computer system, a plurality of target servers, and an authentication store according to some embodiments;

FIG. 2 depicts a block diagram of one embodiment of a computer system suitable for use as a component of the cascading authentication system;

FIG. 3 depicts a conceptual illustration of software components of an authentication plan manager according to some embodiments;

FIG. 4 depicts a conceptual illustration of software components of an authentication store manager according to some embodiments;

FIG. 5 depicts an example of a flow chart for creating an authentication plan for a particular user and target server according to some embodiments;

FIG. 6 depicts an example of a flow chart for authenticating to a target server by a user according to some embodiments; and

FIG. 7 depicts an example of a flow chart for authenticating a user by a target server to some embodiments.

DETAILED DESCRIPTION OF PREFERRED EMBODIMENTS

The following is a detailed description of example embodiments of the invention depicted in the accompanying drawings. The example embodiments are in such detail as to clearly communicate the invention. However, the amount of detail offered is not intended to limit the anticipated variations of embodiments; on the contrary, the intention is to cover all modifications, equivalents, and alternatives falling within the spirit and scope of the present invention as defined by the appended claims. The descriptions below are designed to make such embodiments obvious to a person of ordinary skill in the art.

Generally speaking, systems, methods and media for authenticating a user to a server based on previous authentications to other servers are disclosed. Embodiments of a method for authenticating a user to a server may include receiving a request to authenticate the user to

the server and determining whether authenticating the user requires matching an authentication plan. If a plan is required, the method may also include accessing a stored authentication plan with authentication records each having expected information relating to user access to a different, particular server at a previous layer of authentication than the target server. The method may also include receiving an indication of the user's current authentication plan from an authentication store where the plan has authorization records each having current information relating to user access to a different server, particular server at a previous layer of authentication than the target server. Embodiments of the method may also include comparing the stored authentication plan with the received current authentication plan to determine whether they match and, in response to a match, authenticating the user.

The system and methodology of the disclosed embodiments allows for effective and efficient authentication of a user to a target server by relying on other, previously-made authentications to other servers. Target servers according to the disclosed embodiments are given the ability to check for and require previous layers of authentication according to a pre-established authentication plan before authenticating a user. This solution assists in preventing hackers or others from bypassing earlier layers of authentication, such as by posing as an 'insider', increasing the overall security of the target server. The inside layers of a tiered authorization system may thus be made more secure than previous systems as the inside layers may more directly benefit from authorization schemes of previous layers. In cases where business or user-specified rules dictate that a user must go through multiple defined layers of authentication, the disclosed system and methodology may enhance security, particularly from insiders.

In general, the routines executed to implement the embodiments of the invention, may be part of a specific application, component, program, module, object, or sequence of instructions. The computer program of the present invention typically is comprised of a multitude of instructions that will be translated by the native computer into a machine-readable format and hence executable instructions. Also, programs are comprised of variables and data structures that either reside locally to the program or are found in memory or on storage devices. In addition, various programs described herein may be identified

based upon the application for which they are implemented in a specific embodiment of the invention. However, it should be appreciated that any particular program nomenclature herein is used merely for convenience, and thus the invention should not be limited to use solely in any specific application identified and/or implied by such nomenclature.

While specific embodiments will be described below with reference to particular configurations of hardware and/or software, those of skill in the art will realize that embodiments of the present invention may advantageously be implemented with other substantially equivalent hardware, software systems, manual operations, or any combination of any or all of these. The invention can take the form of an entirely hardware embodiment, an entirely software embodiment or an embodiment containing both hardware and software elements. In a preferred embodiment, the invention is implemented in software, which includes but it not limited to firmware, resident software, microcode, etc.

Aspects of the invention described herein may be stored or distributed on computer-readable medium as well as distributed electronically over the Internet or over other networks, including wireless networks. Data structures and transmission of data (including wireless transmission) particular to aspects of the invention are also encompassed within the scope of the invention. Furthermore, the invention can take the form of a computer program product accessible from a computer-readable medium providing program code for use by or in connection with a computer or any instruction execution system. For the purposes of this description, a computer-usable or computer readable medium can be any apparatus that can contain, store, communicate, propagate, or transport the program for use by or in connection with the instruction execution system, apparatus, or device. The medium may be an electronic, magnetic, optical, electromagnetic, infrared, or semiconductor system (or apparatus or device) or a propagation medium. Examples of a computer-readable medium include a semiconductor or solid state memory, magnetic tape, a removable computer diskette, a random access memory (RAM), a read-only memory (ROM), a rigid magnetic disk and an optical disk. Current examples of optical disks include compact disk – read only memory (CD-ROM), compact disk – read/write (CD-R/W) and DVD.

Each software program described herein may be operated on any type of data processing system, such as a personal computer, server, etc. A data processing system suitable for storing and/or executing program code may include at least one processor coupled directly or indirectly to memory elements through a system bus. The memory elements may include local memory employed during execution of the program code, bulk storage, and cache memories which provide temporary storage of at least some program code in order to reduce the number of times code must be retrieved from bulk storage during execution. Input/output (I/O) devices (including but not limited to keyboards, displays, pointing devices, etc.) may be coupled to the system either directly or through intervening I/O controllers. Network adapters may also be coupled to the system to enable the data processing system to become coupled to other data processing systems or remote printers or storage devices though intervening private or public networks, including wireless networks. Modems, cable modems and Ethernet cards are just a few of the currently available types of network adapters.

Turning now to the drawings, FIG. 1 depicts an environment for a cascading authentication system with a user computer system, a plurality of target servers, and an authentication store according to some embodiments. In the depicted embodiment, the cascading authentication system 100 includes a user computer system 102 and a plurality of target servers 106 in communication via network 104. Target servers 106, as will be described subsequently, are servers in the software sense rather than in the machine classification sense and thus may be considered a software entity (*e.g.*, application, operating system, network interface, etc.) for which authentication may be required for access. The user computer system 102 and/or target servers 106 may also be in communication with an authentication store 108 via network 104.

A user of the user computer system 102 may desire to access a particular target server 106 that is one or more layers down in a series of target servers 106, such as a database target server 106 protected by a firewall target server 106 and an operating system authentication protocol. As will be described in more detail subsequently, the disclosed system may advantageously require information about authentications at lower levels of target server 106 before providing authentication to a particular target server 106, such as by requiring

information about the user's authentication to the firewall or operating system target servers 106 before authenticating to a database target server 106. To accomplish this, the target server 106 may compare a previously stored authentication plan with information about the user's current authentications to determine if they match. If they do not match, the target server 106 may deny access as the user may be posing as an insider to skip multiple levels of authentication while if they do match, the user may be authenticated to the target server 106. Authentication plans may include one or more defined authentication steps that must be performed before a user is allowed to authenticate. An example authentication plan may require, say, that a user must first authenticate at server A before server B, which may be independent of server A, allows the user to authenticate, even if the credentials are otherwise perfect. An authentication plan may require as many steps as is needed or a user or target server 106 desires.

Users may utilize a user computer system 102 according to the present embodiments to facilitate gaining access to a target server 106 via authentication. User computer system 102 may be a personal computer system or other computer system adapted to execute computer programs, such as a personal computer, workstation, server, notebook or laptop computer, desktop computer, personal digital assistant (PDA), mobile phone, wireless device, or set-top box. A user may interact with the user computer system 102 via a user interface to, for example, request access to a target server 106 or to receive information about whether access was granted from the target server 106. User computer system 102 may be in communication with network 104 for transmitting and receiving information.

The user computer system 102 may include an authentication store manager 112 to facilitate cascading authentication. The authentication store manager 112, which will be described in more detail in relation to FIG. 4, may provide for interaction with target servers 106 and/or the authentication store 108. The authentication store manager 112 may, for example, store an authentication event record for each performed authentication step in the authentication store 108. The authentication store manager 112 may also interact with target servers 106, such as when a target server 106 grants or denies access, requests or establishes an authentication plan, or requests resolution of a discrepancy between an authentication plan and the user's current authentications. The authentication store manager 112 may thus serve

as a trusted source of authorization information for authorization mechanisms of various target servers 106 that request such information.

Network 104 may be any type of data communications channel or combination of channels, such as the Internet, an intranet, a LAN, a WAN, an Ethernet network, a wireless network, telephone network, a proprietary network, or a broadband cable network. In one example, a LAN may be particularly useful as a network 104 between a user computer system 102 and target servers 106 in a corporate environment to facilitate communication within the organization, while in other examples network 104 may connect a user computer system 102 with a Web-based authentication store 108 with the Internet serving as network 104. Those skilled in the art will recognize, however, that the invention described herein may be implemented utilizing any type or combination of data communications channel(s) without departure from the scope and spirit of the invention.

As described previously, target servers 106 are software entities for which authentication may be required, and granted, in order to access resources of each target server 106. Target servers 106 may include a wide variety of software entities, including operating systems, databases, firewalls, virtual private networks (VPNs), networks, applications, or other entities. One or more target servers 106 may be implemented on server computer systems such as an application server as well as any other type of computer system (such as described in relation to FIG. 2). As depicted in FIG. 1, the target servers 106 may be nested in layers so that access to an inner target server 106 first requires access to outer (in FIG. 1), lower level target servers 106. In the depicted embodiment, for example, access to the target server 106 at level 3 would also require access to the target servers 106 at levels 1 and 2.

Each target server 106 may include an authentication plan manager 110 to access a stored authentication plan associated with a user requesting access to the target server 106. The stored authentication plan may include one or more authentication records each having expected information relating to access by a user to a different target server 106 at a previous layer of authentication than the target server 106. The authentication plan manager 110 may provide a current authentication plan representing the user's current authentication situation from the authentication store manager 112 (which itself may access the current

authentication plan from the authentication store 108). The authentication plan manager 110 may also determine whether to authenticate a user based on a comparison between the stored authentication plan and the current authentication plan. By comparing an expected stored authentication plan with a current authentication plan, the authentication plan manager 110 may ascertain whether a user has properly authenticated at lower levels of target server 106 and may deny access to such user even if their other credentials (such as passwords) are correct, providing improved security for the target server 106.

Some target servers 106, such as legacy systems, may not have an authentication plan manager 110 and thus do not ask for the relevant authentication plans, but the authentications from these target servers 106 may still be used by subsequent target servers 106 to enhance their security. The disclosed system is thus compatible with existing infrastructure as target servers 106 that have not implemented the disclosed system will not request authentication information and instead will perform authentication normally. As depicted in FIG. 1, some target servers 106 in one cascading authentication system 100 may have an authentication plan manager 110 (and thus have implemented the disclosed system) while others do not.

Authentication store 108 may include any type or combination of storage devices, including volatile or non-volatile storage such as hard drives, storage area networks, memory, fixed or removable storage, or other storage devices. The authentication store 108 in some embodiments may be an encrypted database of disparate local and remote authentication information that can be written to and read by a trusted source such as the authentication store manager 112 on behalf of any authorized authentication mechanism that requests it. The authentication store 108 may be located in a variety of positions with the cascading authentication system 100, such as being a stand-alone component (perhaps implemented by a trusted third party on a remote server or network of servers) or part of the user computer system 102 or authentication store manager 112.

FIG. 2 depicts a block diagram of one embodiment of a computer system 200 suitable for use as a component of the cascading authentication system. Other possibilities for the computer system 200 are possible, including a computer having capabilities other than those

ascribed herein and possibly beyond those capabilities, and they may, in other embodiments, be any combination of processing devices such as workstations, servers, mainframe computers, notebook or laptop computers, desktop computers, PDAs, mobile phones, wireless devices, set-top boxes, or the like. At least certain of the components of computer system 200 may be mounted on a multi-layer planar or motherboard (which may itself be mounted on the chassis) to provide a means for electrically interconnecting the components of the computer system 200. Computer system 200 may be utilized to implement one or more target servers 106, a user computer system 102, and/or an authentication store 108.

In the depicted embodiment, the computer system 200 includes a processor 202, storage 204, memory 206, a user interface adapter 208, and a display adapter 210 connected to a bus 212 or other interconnect. The bus 212 facilitates communication between the processor 202 and other components of the computer system 200, as well as communication between components. Processor 202 may include one or more system central processing units (CPUs) or processors to execute instructions. The processor 202 may utilize storage 204, which may be non-volatile storage such as one or more hard drives, tape drives, diskette drives, CD-ROM drive, DVD-ROM drive, or the like. The processor 202 may also be connected to memory 206 via bus 212, such as via a memory controller hub (MCH). System memory 206 may include volatile memory such as random access memory (RAM) or double data rate (DDR) synchronous dynamic random access memory (SDRAM). In the disclosed systems, for example, a processor 202 may execute instructions to perform functions of the authentication store manager 112, such as by interacting with an authentication store 108, and may temporarily or permanently store information during its calculations or results after calculations in storage 204 or memory 206. All of part of the authentication store manager 112, for example, may be stored in memory 206 during execution of its routines.

The user interface adapter 208 may connect the processor 202 with user interface devices such as a mouse 220 or keyboard 222. The user interface adapter 208 may also connect with other types of user input devices, such as touch pads, touch sensitive screens, electronic pens, microphones, etc. A user of a client 102 requesting access to a target server 106 or resolving an authentication plan conflict, for example, may utilize the keyboard 222 and mouse 220 to interact with the computer systems. The bus 212 may also connect the

processor 202 to a display, such as an LCD display or CRT monitor, via the display adapter 210.

While the authentication store manager 112 is depicted as located in the processor 202 in FIG. 2 (such as a component of the BIOS), one of ordinary skill in the art will recognize that other alternatives are possible. In preferred embodiments, the authentication store manager 112 may execute in a location with a sufficient level of security so as to minimize the possibility of hacking. Because the authentication store manager 112 is trusted to provide accurate information from the authentication store 108, it may preferably be implemented by a trusted party with trusted encryption and also exchange private keys (or other suitable encryption method) with each target server 106 that is authenticated to prevent spoofing these records. The authentication store manager 112 may thus be implemented at low level in the hardware (BIOS) as depicted in FIG. 2, in an operating system (i.e., kernel), or by a third party such as VeriSign, Inc. or a corporate Lightweight Directory Access Protocol (LDAP) directory extension. Because of firewalls and other limited degrees of network visibility, proxy authentication store managers 112 may be required to bridge across networks to field requests.

FIG. 3 depicts a conceptual illustration of software components of an authentication plan manager 110 according to some embodiments. As described previously (and in more detail in relation to FIG. 7), the authentication plan manager 110 may authenticate a user based on a stored authentication plan and a user's current authentications as found in the authentication store 108. The authentication plan manager 110 may include an authentication store manager interface module 302, a user computer system interface module 304, an authentication plan repository 306, and an authentication module 308. The authentication store manager interface module 302 may provide for communication to and from the authentication store 108 via the authentication store manager 112, thus serving as an interface between the authentication store 108 and other components of the authentication plan manager 110. The user computer system interface module 304 may provide for communication to and from a user computer system, including receiving requests for access to the target server 106 and transmitting an indication of whether access was granted to the user. The functionality of the authentication store manager interface module 302 and the

user computer system interface module 304 may be combined into one module in some embodiments, such as when the user computer system 102 includes the authentication store manager 112.

The authentication module 308 may provide a variety of functions to facilitate authentication of a user according to the present embodiments. The authentication module 308 may store and access authentication plans associated with a plurality of users in the authentication plan repository 306. The authentication module 308 may store an authentication plan when it is received from a user or when it is developed for a user (or in conjunction with a user), and may access a stored authentication plan in response to a user requesting access to the target server 106 implementing the authentication plan manager 110. After the user computer system interface module 304 receives an authentication request from a user, the authentication module 308 may access the stored authentication plan for that user from the authentication plan repository 306 and compare that plan to a current authentication plan (received by the authentication store manager interface module 302).

FIG. 4 depicts a conceptual illustration of software components of an authentication store manager 112 according to some embodiments. As described previously (and in more detail in relation to FIG. 6), the authentication store manager 112 may facilitate storing and managing authentication information relating to a user's authentication of a plurality of target servers 106 by managing the authentication store 108. The authentication store manager 112 may include a user interface module 402, an authentication store interface module 404, a server interface module 406, an authentication event monitor 408, and an authentication plan generator 410. The user interface module 402 may facilitate communication to and from a user, including receiving requests for access to a target server 106 and transmitting an indication that access was granted or denied, that an authentication plan needs to be created or modified, or other information. The authentication store interface module 404 may facilitate communication to and from the authentication store 108, including storing an indication of authentication events in the authentication store 108 and accessing authentication plans upon request of an authenticating entity such as a target server 106. The server interface module 406 may facilitate communication between a target server 106 (and its authentication plan manager 110) and the authentication store manager 112.

The three interface modules 402, 404, and 406 may each provide communication between components of the authentication store manager 112 and outside entities, and their functionality may be combined or divided in any fashion.

The authentication event monitor 408 may monitor a user's performed authentication steps (*e.g.*, entering a password, using a smart card, etc.) and may store an encrypted indication of such steps in the authentication store 108 (via the authentication store interface module 404). In some embodiments, the authentication event monitor 408 may at every authentication step create an encrypted event record for storage in the authentication store 108 with information that subsequent layers of authentication may request. The information in the authentication record may include one or more of a unique record identifier for internal management use, one or more target server 106 identifiers (such as MAC address, server type, server identifier, server group, IP address, etc.), one or more user identifiers (such as a user name, group name, etc.), one or more authentication event facts (such as the number of failed authentication attempts prior to successful login, timestamp local to the authentication store 108, etc.,) or other information.

The authentication plan generator 410 may facilitate creation and maintenance of an authentication plan for a user. A user may create a plan if required or allowed by a target server 106, such as by identifying current records in their authentication store that should be used in the authentication plan for the particular target server 106. In some embodiments, an administrator of the target server 106 may have pre-established the required authentication steps required in any authentication plan. A server administrator could require, for example, that the authentication plan include at least server types BIOS, OS, VPN, and FIREWALL, and perhaps further pre-known information such as a specific VPN server group or firewall IP address. A user may also request additional steps be included beyond those required by a target server 106 for the user's protection.

FIG. 5 depicts an example of a flow chart 500 for creating an authentication plan for a particular user and target server according to some embodiments. The method of flow chart 500 may be performed, in one embodiment, by components of the cascading authentication system 100 such as the authentication plan manager 110 and the authentication store

manager 112. Flow chart 500 begins with element 502, receiving a request to create an authentication plan for a particular server. The request for an authentication plan may be received from a target server 106 attempting to inform a user that an authentication plan is required or it may be received from a user requesting to establish an authentication plan with a particular target server 106.

At decision block 504, the authentication store manager 112 may determine whether any authentication records current exist for the particular user and other levels of authentication. If so, the authentication store manager 112 may present the current list of existing authentication records (from the authentication store 108) to the user so that the user may select which authentication events they would like to include in the authentication plan for the target server 106. The authentication store manager 112 may receive an identification of the current records in the authentication store 108 that will be used in the authentication plan at element 506. The target server 108 may also require particular authentication events from the user in addition to those chosen by the user. At element 508, the authentication plan generator 410 of the authentication store manager 112 may create the authentication plan based on the preferences and selections of the user and the target server 106.

After the authentication plan has been created, the server interface module 406 of the authentication store manager 112 may then transmit the plan to the target server 106 at element 510. At element 512, the target server 106 may receive the authentication plan and store the plan in the authentication plan repository 306 at element 514, after which the method terminates. The authentication plan repository 306 may serve as storage for a variety of authentication plans for many users in some embodiments.

FIG. 6 depicts an example of a flow chart 600 for authenticating to a target server by a user according to some embodiments. The method of flow chart 600 may be performed, in one embodiment, by components of the cascading authentication system 100 such as the authentication store manager 112. Flow chart 600 begins with optional element 602, updating authentication records in the authentication store 108. The authentication store manager 112 may update the authentication records for a variety of reasons. In some embodiments, for example, the authentication store manager 112 may attempt to avoid stale

information by deleting the authentication record whenever a target server 106 that has been authenticated is logged our disconnected or whenever a target server 106 earlier in the authentication plan is similarly logged out or disconnected. In these embodiments, the authentication record may first be written to another table for archival purposes such as reporting or usage analysis. The authentication store manager 112 may accomplish this by periodically querying the target server's authentication system for a current status, receiving requests to delete records from the target server 106, and/or receiving such requests from the user. The authentication store manager 112 may also perform the reverse methodology by logging out downstream target servers 106 in the event a target server 106 higher in the authentication plan is logged out or in the event such a server revokes or suspends the user.

The user computer system 102 may perform authentication steps for different target servers 106 at element 604, such as by successfully authenticating to a target server 106 such as the machine hardware (with a power on password), their operating system, VPN, firewall, database, etc. At element 606, the authentication store manager 112 may store an encrypted authentication record in the user's authentication store 108, as described previously. The authentication record may include information about performance of the authentication step, such as indication of its success, a timestamp, an indication of how many attempts were required, etc. The user computer system 102 may then attempt to authenticate to a target server 106 at decision block 608. If such target server 106 does not require an authentication plan, the method may return to element 604 for performing the authentication step and storing an authentication record based on the performed step.

If the target server 106 does require an authentication plan, the authentication store manager 112 may receive a request for the current authentication plan from the target server 106 at element 610. As described previously, the current authentication plan may include an authentication record for one or more authentication steps for servers at previous layers of authentication to the target server 106. The authentication records included in the current authentication plan are thus the expected authentication records for the user (*i.e.*, what the target server 106 expects the user to have done). The authentication plan manager 112 may then at element 612 access the authentication plan for the user and may then transmit the expected information from the authentication plan to the target server 106 at element 614.

After the information from the authentication plan has been transmitted, the user computer system 102 and its authentication store manager 112 may receive at element 616 an indication of whether access was granted by the target server 106. If access was granted at decision block 618, the method of flow chart 600 either terminates (and the user performs whatever task they were seeking access to accomplish) or returns to element 604 for performing an authentication step at element 604. If access was not granted at decision block 618, the authentication store manager 112 may be notified that they are accessing via an unauthorized authentication plan. If the authentication plan needs to be changed, the method continues to element 620 where the authentication store manager 112 may resolve any authentication plan mismatch with the target server 106, after which the method terminates. If the authentication plan needs to be changed, that will typically require administrator intervention in coordination with the user, which may be implemented in any fashion, such as by requiring extra authentication such as via a secret passphrase, a smartcard, or other authentication method. This helps prevent someone spoofing the user to request the authentication plan be changed to something easier for the spoofer (or hacker) or as a denial of service attack.

FIG. 7 depicts an example of a flow chart 700 for authenticating a user by a target server to some embodiments. The method of flow chart 700 may be performed, in one embodiment, by components of the cascading authentication system 100 such as the authentication plan manager 110 of a target server 106. Flow chart 700 begins with element 602, receiving a request from a user computer system 102 for the user to authenticate to the target server 106. At decision block 704, the authentication plan manager 110 may determine whether the particular user desiring to authenticate to the target server 106 requires matching an established authentication plan. If no authentication plan is required, the method may advance to element 722, where the user is authenticated in a standard fashion (*i.e.*, verifying their authentication credentials), after which the method terminates. If an authentication plan is required, the method continues to element 706.

At element 706, the authentication module 308 of the authentication plan manager 110 may access the stored authentication plan for the user that is stored in the authentication plan repository 306. The user computer system interface module 304 may transmit at element

708 a request for an authentication plan to the authentication store manager 112. The authentication plan manager 110 may receive an indication from the authentication store manager 112 at element 710 of whether a current authentication plan exists in the authentication store 108. If a current plan does not exist at decision block 714, the method continues to element 724, where the authentication module 308 denies access to the target server 106 and attempts to resolve any authentication plan mismatch with the authentication store manager 112. For example, if the attempt is the user's first attempt to authenticate to the target server 106 (or after a reset), an authentication plan create method may be invoked to work with the user to develop an authentication plan for the target server 106. Alternatively, as described previously, the user may be requested to provide additional authentication credentials, such as a passphrase, in order to authenticate.

If a current authentication plan does exist at decision block 714, the authentication plan manager 110 may at element 716 request the authentication records relevant to the stored authentication plan from the authentication store 108 via the authentication store manager 112. At element 718, the authentication plan manager 110 may receive the current authentication plan (or the subset of authentication records of the current authentication plan that is required to match the stored authentication plan).

The authentication module 308 may at decision block 720 compare the stored authentication plan with the received current authentication plan to determine if they match sufficiently for the user to be allowed access. In some embodiments, the authentication module 308 may require an exact match between the stored authentication plan and the current authentication plan in order to allow access. In other embodiments, the authentication module 308 may make a more sophisticated analysis, such as by analyzing the timestamps of authentication events in the current authentication plans (and rejecting those that are too long ago in time), analyzing the matter of authentication for previous authentication events (*e.g.*, rejecting those that show a suspicious pattern, such as too many attempts before authentication), or other types of analysis.

If the authentication module 308 determines at decision block 720 that a match exists, the method continues to element 722 where the user may be authenticated in a standard fashion

(such as by requiring particular authentication credentials), after which the method terminates. If no match exists, the method of flow chart 700 continues to element 724 for resolving an authentication plan mismatch, as described previously, after which the method may terminate. The method of flow chart 700 may thus provide for improved authentication of a user to a target server 106 by comparing the user's current authentications to a previously established authentication plan and requiring a sufficient match to allow the user to authenticate in the standard manner.

It will be apparent to those skilled in the art having the benefit of this disclosure that the present invention contemplates methods, systems, and media for authenticating users to a server based on previous authentications to other servers by the user. It is understood that the form of the invention shown and described in the detailed description and the drawings are to be taken merely as examples. It is intended that the following claims be interpreted broadly to embrace all the variations of the example embodiments disclosed.

CLAIMS

1. A method for authenticating a user to a target server, the method comprising:

receiving a request from a user computer system to authenticate the user to the target server;

determining whether authenticating the user requires matching an authentication plan;

in response to determining that matching an authentication plan is required, accessing a stored authentication plan associated with the user, the stored authentication plan having one or more authentication records each having expected data associated with user access to a server at a previous layer of authentication than the target server;

receiving an indication of the user's current authentication plan from an authentication store, the current authentication plan having one or more authentication records each having current data associated with user access to a server at a previous layer of authentication than the target server;

comparing the stored authentication plan with the received current authentication plan to determine whether they match; and

in response to a match between the stored authentication plan and the current authentication plan, authenticating the user.

2. The method of claim 1, further comprising in response to determining that matching an authentication plan is required, transmitting a request for a current authentication plan to the user computer system.

3. The method of claim 1, further comprising in response to a mismatch between the stored authentication plan and the current authentication plan, resolving the authentication plan mismatch.

4. The method of claim 3, wherein resolving the authentication plan mismatch comprises at least one of: requiring additional authentication data; modifying an authentication plan and creating an authentication plan.

5. The method of claim 1, wherein comparing the stored authentication plan with the current authentication plan to determine whether they match comprises comparing

authentication records of each authentication plan to determine whether at least a specified subset of authentication records match between the authentication plans.

6. The method of claim 1, further comprising:

performing an authentication step for one or more servers at a previous layer of authentication to the target server; and

storing an authentication event record for each performed authentication step in an authentication store.

7. The method of claim 1, further comprising

attempting to authenticate to the target server, the target server requiring an authentication plan associated with the user; and

receiving an indication of whether access to the target server was granted.

8. The method of claim 1, further comprising transmitting a current authentication plan to the target server, the current authentication plan having a stored authentication record for one or more performed authentication steps for servers at previous layers of authentication to the target server.

9. The method of claim 1, further comprising receiving a request for a current authentication plan from the target server.

10. A computer program comprising program code means adapted to perform, when the program is executed on a computer, the steps of:

receiving a request from a user computer system to authenticate a user to a target server;

determining whether authenticating the user requires matching an authentication plan;

in response to determining that matching an authentication plan is required, accessing a stored authentication plan associated with the user, the stored authentication plan having one or more authentication records each having expected data associated with user access to a server at a previous layer of authentication than the target server;

receiving an indication of the user's current authentication plan from an authentication store, the current authentication plan having one or more authentication records each having current data associated with user access to a server at a previous layer of authentication than the target server;

comparing the stored authentication plan with the received current authentication plan to determine whether they match; and

in response to a match between the stored authentication plan and the current authentication plan, authenticating the user.

11. The computer program of claim 10, further comprising in response to determining that matching an authentication plan is required, transmitting a request for a current authentication plan to the user computer system.

12. The computer program of claim 10, further comprising in response to a mismatch between the stored authentication plan and the current authentication plan, resolving the authentication plan mismatch.

13. The computer program of claim 12, wherein resolving the authentication plan mismatch comprises at least one of: requiring additional authentication data; modifying an authentication plan and creating an authentication plan.

14. The computer program of claim 10, wherein comparing the stored authentication plan with the current authentication plan to determine whether they match comprises comparing authentication records of each authentication plan to determine whether at least a specified subset of authentication records match between the authentication plans.

15. The computer program of claim 10, further comprising:

performing an authentication step for one or more servers at a previous layer of authentication to the target server; and

storing an authentication event record for each performed authentication step in an authentication store.

16. The computer program of claim 10, further comprising

attempting to authenticate to the target server, the target server requiring an authentication plan associated with the user; and

receiving an indication of whether access to the target server was granted.

17. The computer program of claim 10, further comprising transmitting a current authentication plan to the target server, the current authentication plan having a stored authentication record for one or more performed authentication steps for servers at previous layers of authentication to the target server.

18. The computer program of claim 10, further comprising receiving a request for a current authentication plan from the target server.

19. An apparatus for authenticating a user to a target server, the apparatus comprising:

means for receiving a request from a user computer system to authenticate the user to the target server;

means for determining whether authenticating the user requires matching an authentication plan;

means, responsive to determining that matching an authentication plan is required, for accessing a stored authentication plan associated with the user, the stored authentication plan having one or more authentication records each having expected data associated with user access to a server at a previous layer of authentication than the target server;

means for receiving an indication of the user's current authentication plan from an authentication store, the current authentication plan having one or more authentication records each having current data associated with user access to a server at a previous layer of authentication than the target server;

means for comparing the stored authentication plan with the received current authentication plan to determine whether they match; and

means, responsive to a match between the stored authentication plan and the current authentication plan, for authenticating the user.

20. The apparatus of claim 19, further comprising: means, responsive to a determination that matching an authentication plan is required, for transmitting a request for a current authentication plan to the user computer system.

21. The apparatus of claim 19, further comprising: means, responsive to a mismatch between the stored authentication plan and the current authentication plan, for resolving the authentication plan mismatch.

22. The apparatus of claim 21, wherein the means for resolving further comprises at least one of:

means for requiring additional authentication data;

means for modifying an authentication plan and

means for creating an authentication plan.

23. The apparatus of claim 19, wherein the means for comparing the stored authentication plan with the current authentication plan to determine whether they match further comprises: means for comparing authentication records of each authentication plan to determine whether at least a specified subset of authentication records match between the authentication plans.

24. The apparatus of claim 19, further comprising:

means for performing an authentication step for one or more servers at a previous layer of authentication to the target server; and

means for storing an authentication event record for each performed authentication step in an authentication store.

25. The apparatus of claim 19, further comprising

means for attempting to authenticate to the target server, the target server requiring an authentication plan associated with the user; and

means for receiving an indication of whether access to the target server was granted.

26. The apparatus of claim 19, further comprising: means for transmitting a current authentication plan to the target server, the current authentication plan having a stored authentication record for one or more performed authentication steps for servers at previous layers of authentication to the target server.

27. The apparatus of claim 19, further comprising: means for receiving a request for a current authentication plan from the target server.

28. A system for authenticating a user to a target server, the system comprising:

a target server having an authentication plan manager operable to access a stored authentication plan associated with a user requesting access to the target server, the stored authentication plan comprising one or more authentication records each having expected data associated with access by a user to a server at a previous layer of authentication than the target server;

an authentication store operable to store a current authentication plan associated with the user, the current authentication plan comprising one or more authentication records each having current data associated with access by a user to a server at a previous layer of authentication than the target server;

an authentication store manager operable to communicate with the target server and the authentication store and operable to provide the current authentication plan associated with a particular user to the authentication plan manager of the target server; and

wherein the authentication plan manager of the target server is operable to determine whether to authenticate a user based on a comparison between the stored authentication plan and the current authentication plan.

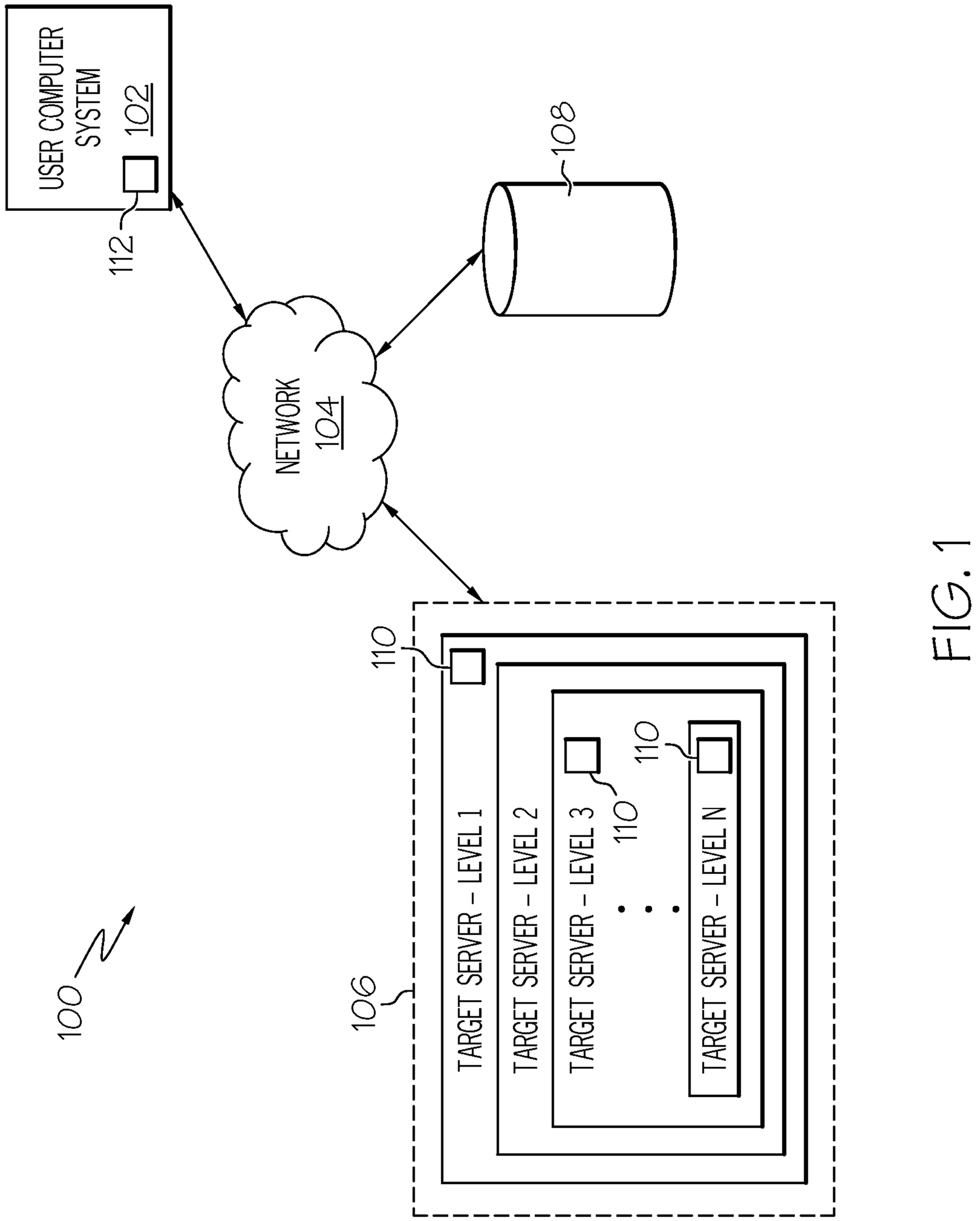
USER COMPUTER SYSTEM 102
112
108
NETWORK 104
110
100
106
TARGET SERVER – LEVEL 1
TARGET SERVER – LEVEL 2
TARGET SERVER – LEVEL 3
110
110
TARGET SERVER – LEVEL N

FIG. 1

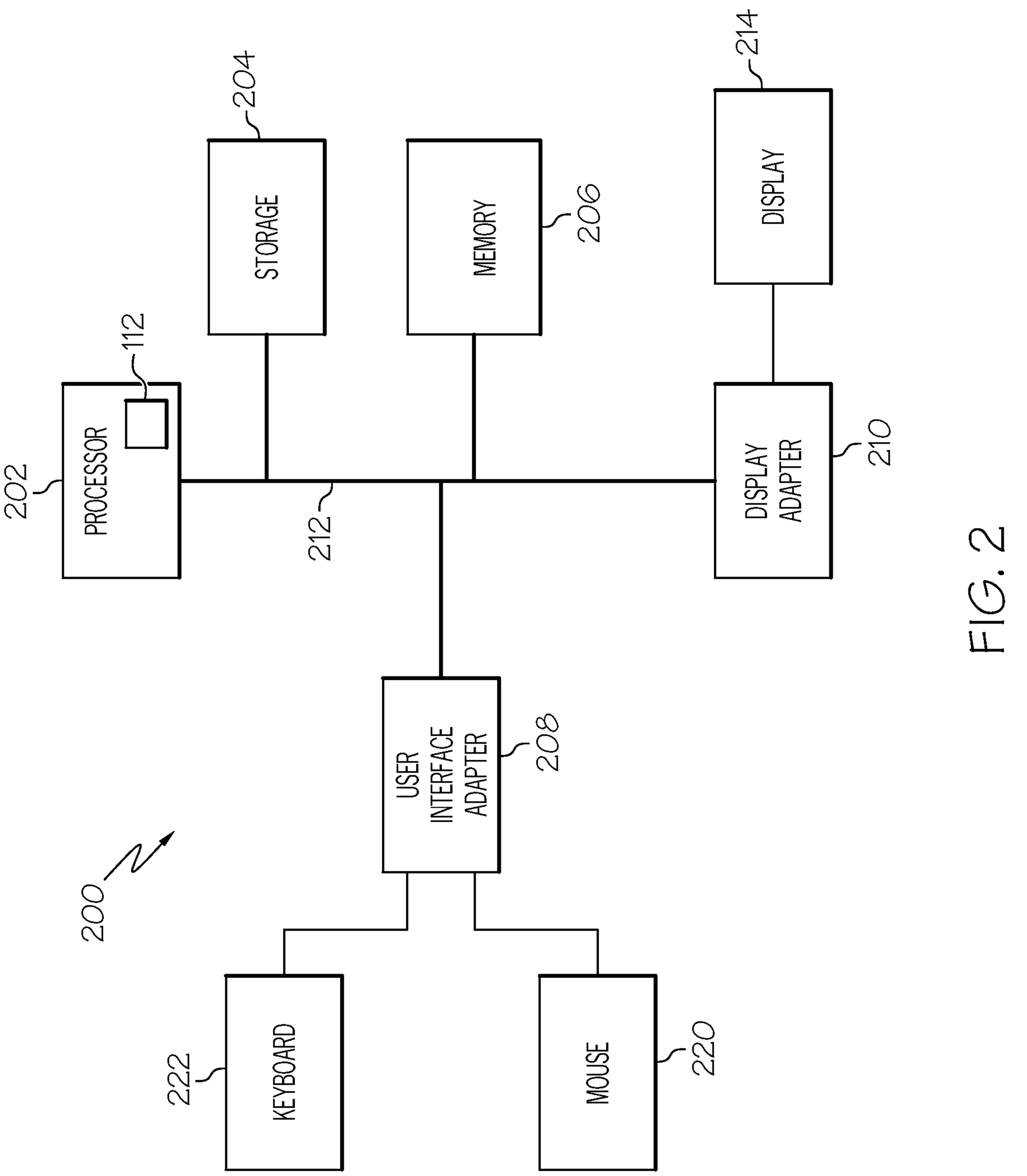
200
202
PROCESSOR
112
204
STORAGE
206
MEMORY
212
208
USER INTERFACE ADAPTER
210
DISPLAY ADAPTER
214
DISPLAY
220
MOUSE
222
KEYBOARD

FIG. 2

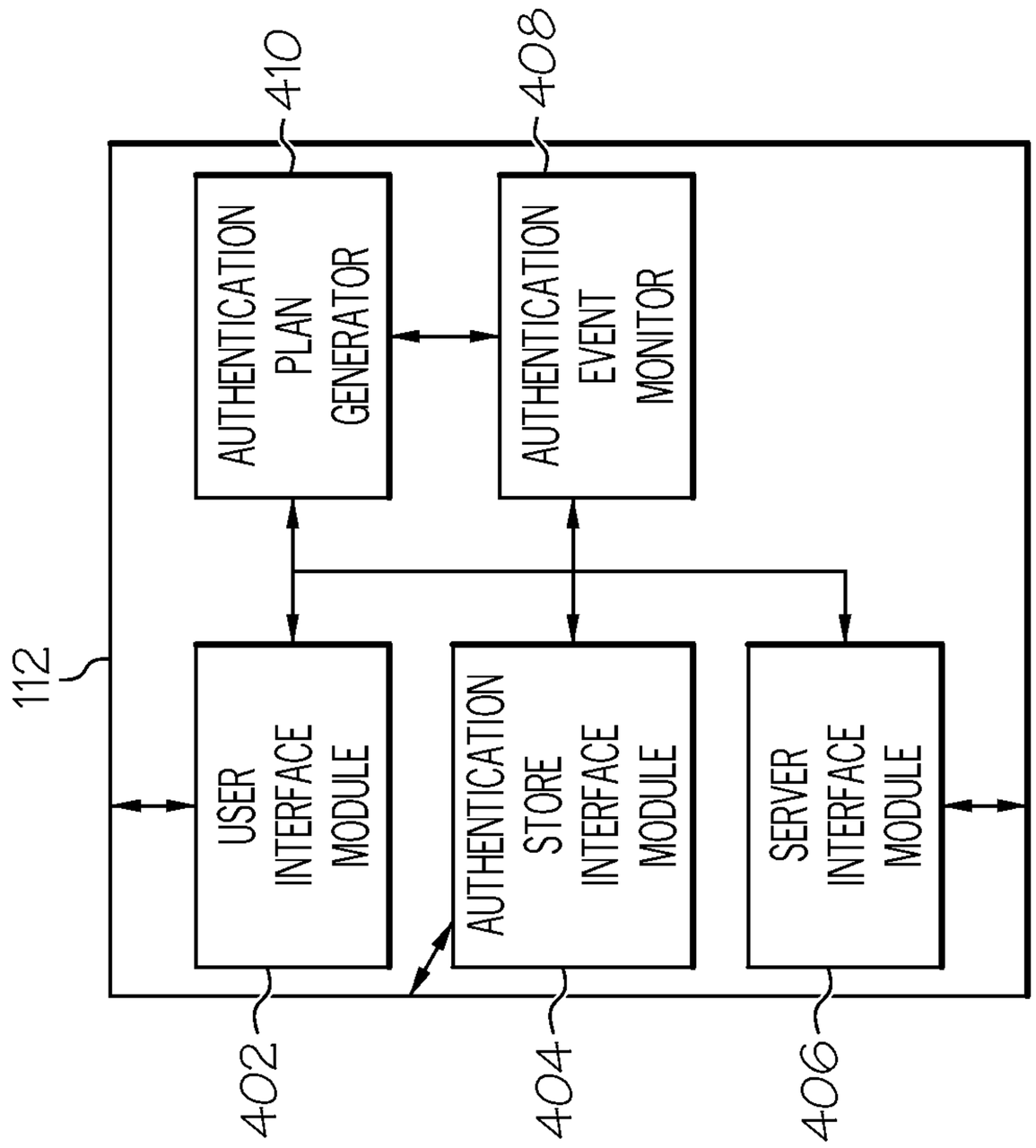
112
410
408
AUTHENTICATION PLAN GENERATOR
AUTHENTICATION EVENT MONITOR
USER INTERFACE MODULE
AUTHENTICATION STORE INTERFACE MODULE
SERVER INTERFACE MODULE
402
404
406

FIG. 4

110
308
306
AUTHENTICATION MODULE
AUTHENTICATION PLAN REPOSITORY
UCS INTERFACE MODULE
AUTHENTICATION STORE MANAGER INTERFACE MODULE
304
302

FIG. 3

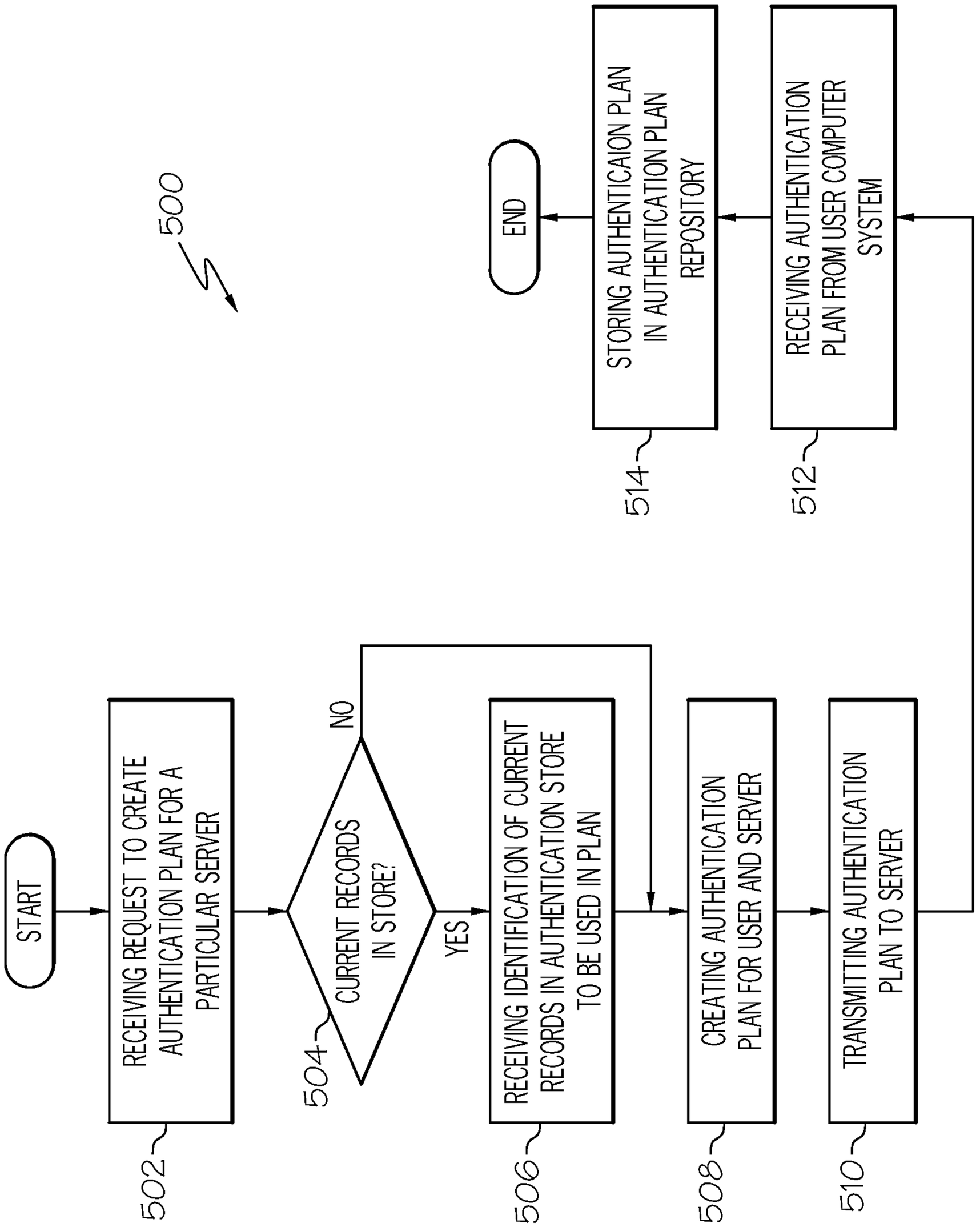
500
START
RECEIVING REQUEST TO CREATE AUTHENTICATION PLAN FOR A PARTICULAR SERVER
502
CURRENT RECORDS IN STORE?
504
NO
YES
RECEIVING IDENTIFICATION OF CURRENT RECORDS IN AUTHENTICATION STORE TO BE USED IN PLAN
506
CREATING AUTHENTICATION PLAN FOR USER AND SERVER
508
TRANSMITTING AUTHENTICATION PLAN TO SERVER
510
RECEIVING AUTHENTICATION PLAN FROM USER COMPUTER SYSTEM
512
STORING AUTHENTICAION PLAN IN AUTHENTICATION PLAN REPOSITORY
514
END

FIG. 5

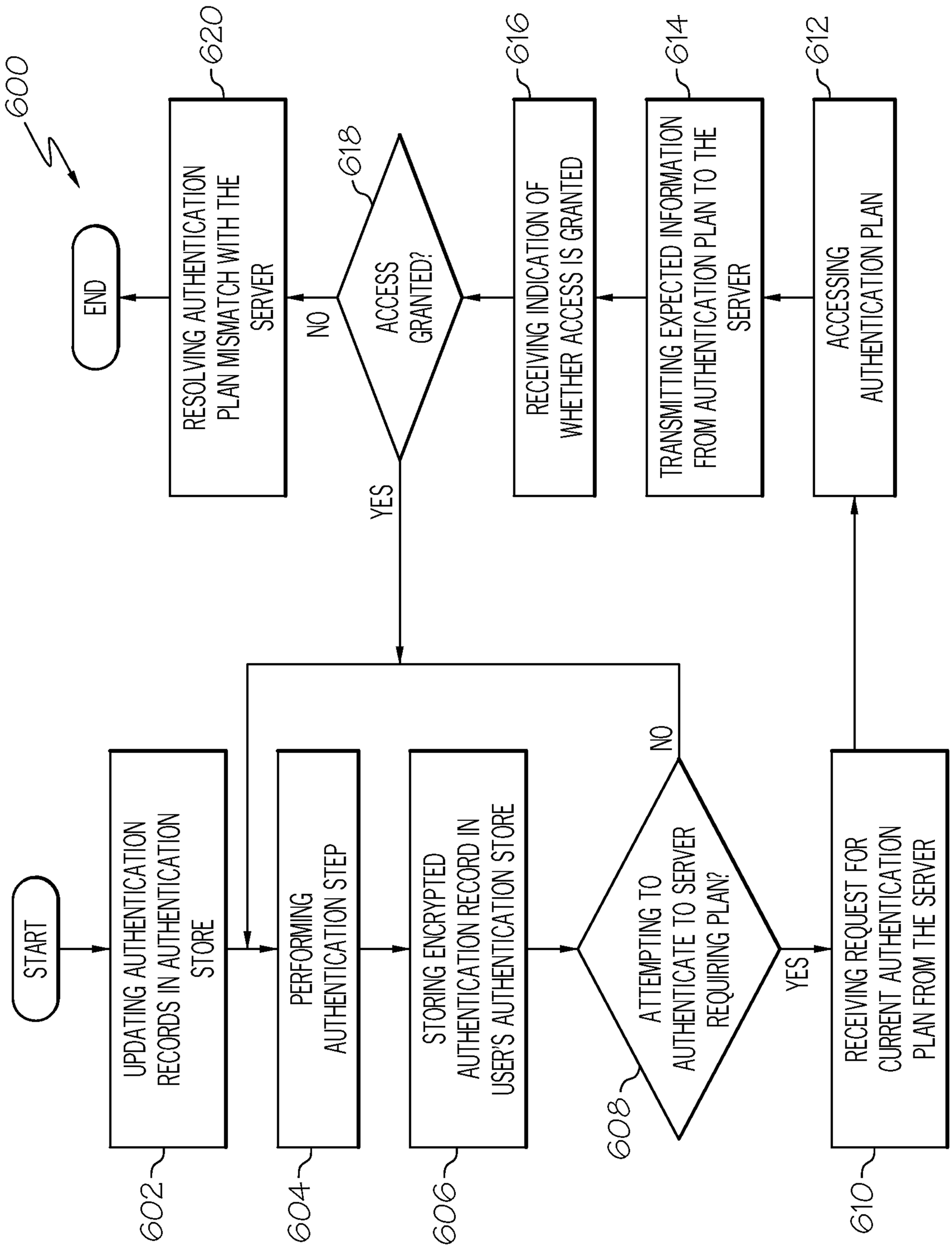
600
END
RESOLVING AUTHENTICATION PLAN MISMATCH WITH THE SERVER
618
ACCESS GRANTED?
NO
YES
RECEIVING INDICATION OF WHETHER ACCESS IS GRANTED
TRANSMITTING EXPECTED INFORMATION FROM AUTHENTICATION PLAN TO THE SERVER
ACCESSING AUTHENTICATION PLAN
START
UPDATING AUTHENTICATION RECORDS IN AUTHENTICATION STORE
PERFORMING AUTHENTICATION STEP
STORING ENCRYPTED AUTHENTICATION RECORD IN USER'S AUTHENTICATION STORE
ATTEMPTING TO AUTHENTICATE TO SERVER REQUIRING PLAN?
NO
YES
RECEIVING REQUEST FOR CURRENT AUTHENTICATION PLAN FROM THE SERVER
602
604
606
608
610

FIG. 6

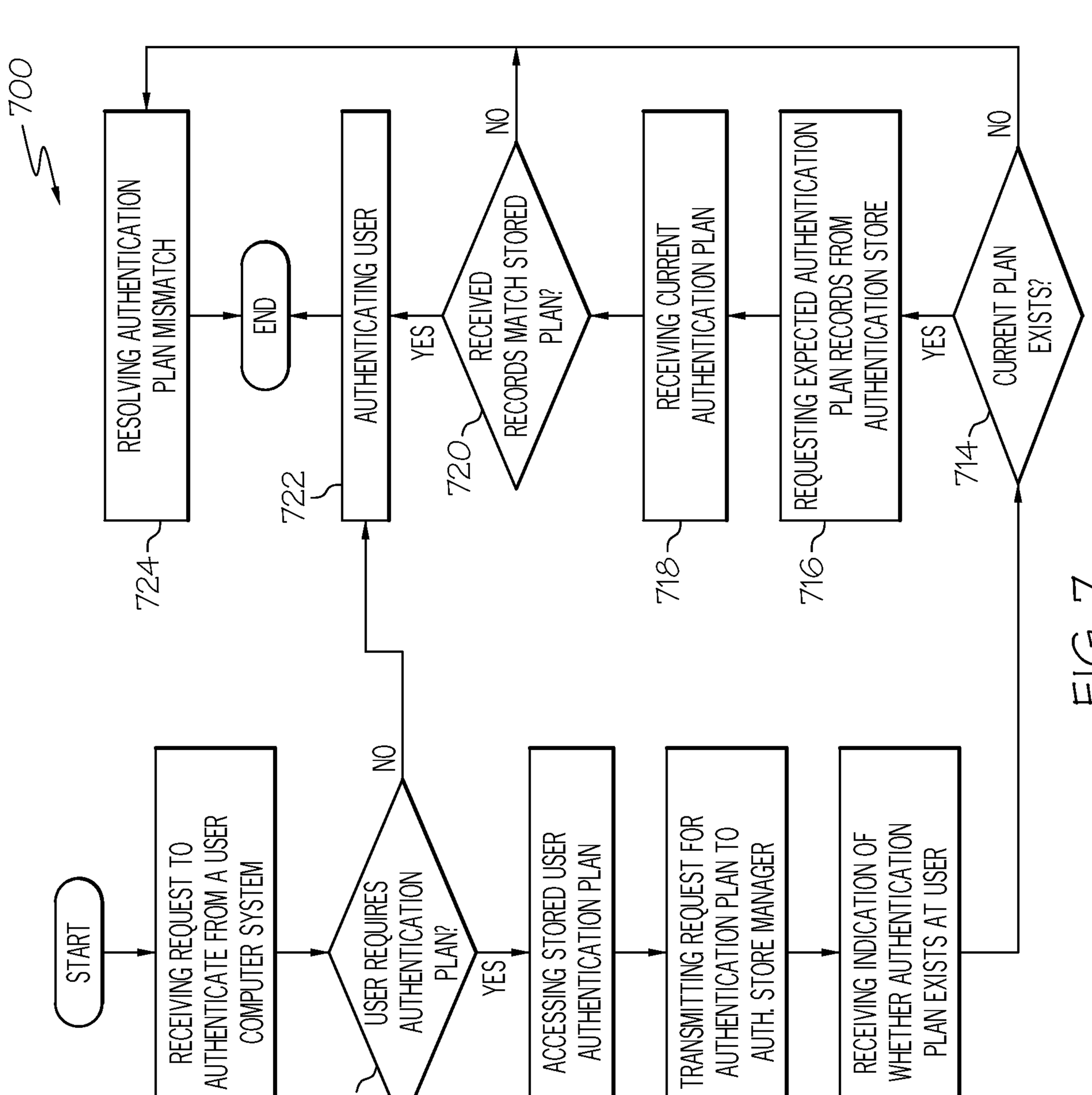
700
START
RECEIVING REQUEST TO AUTHENTICATE FROM A USER COMPUTER SYSTEM
702
USER REQUIRES AUTHENTICATION PLAN?
704
NO
YES
ACCESSING STORED USER AUTHENTICATION PLAN
706
TRANSMITTING REQUEST FOR AUTHENTICATION PLAN TO AUTH. STORE MANAGER
708
RECEIVING INDICATION OF WHETHER AUTHENTICATION PLAN EXISTS AT USER
710
CURRENT PLAN EXISTS?
714
NO
YES
REQUESTING EXPECTED AUTHENTICATION PLAN RECORDS FROM AUTHENTICATION STORE
716
RECEIVING CURRENT AUTHENTICATION PLAN
718
RECEIVED RECORDS MATCH STORED PLAN?
720
NO
YES
AUTHENTICATING USER
722
END
RESOLVING AUTHENTICATION PLAN MISMATCH
724

FIG. 7

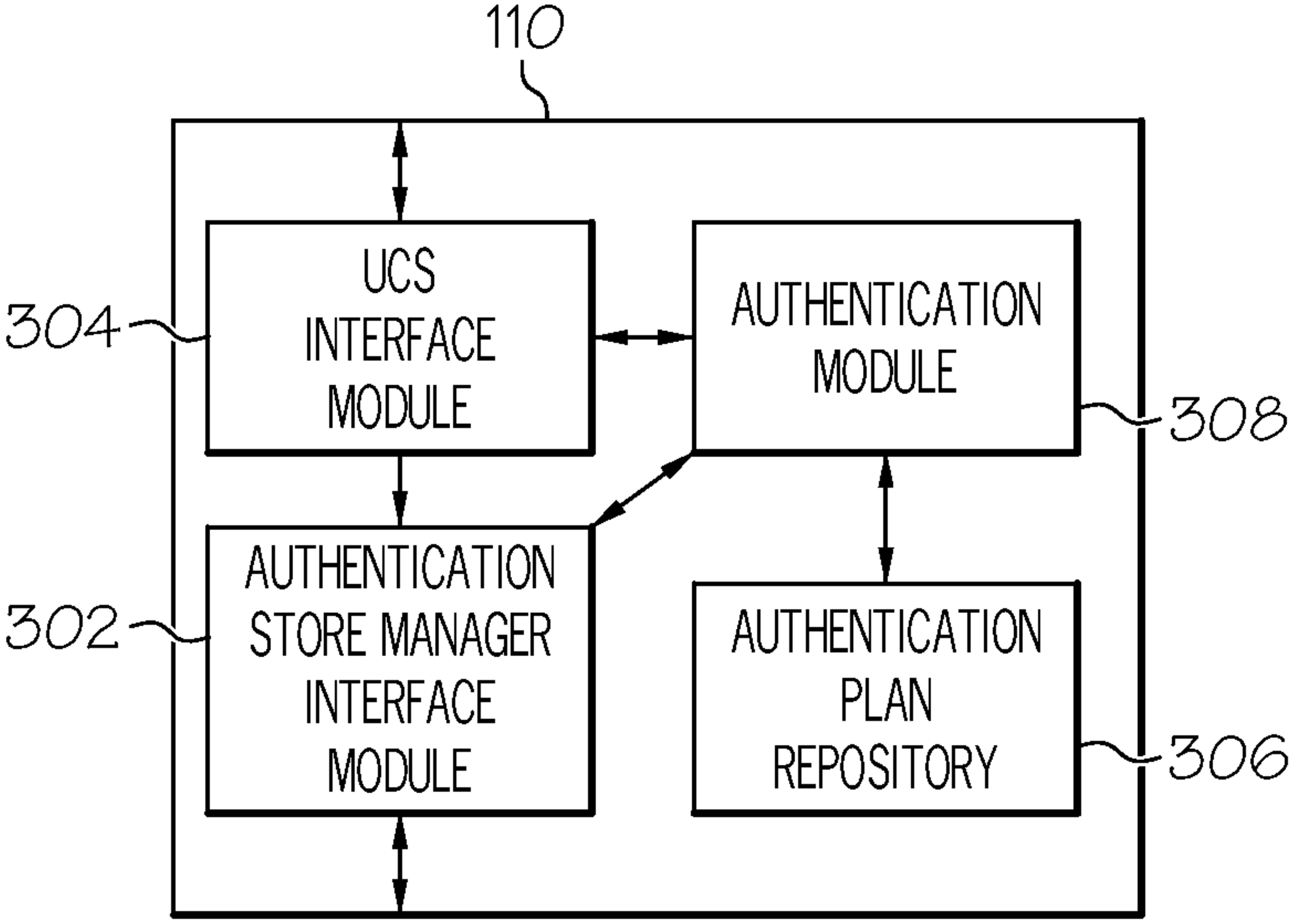
110
304
UCS INTERFACE MODULE
AUTHENTICATION MODULE
308
302
AUTHENTICATION STORE MANAGER INTERFACE MODULE
AUTHENTICATION PLAN REPOSITORY
306

FIG. 3