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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
18 January 2007 (18.01.2007)

PCT

(10) International Publication Number
WO 2007/008976 A1

(51) International Patent Classification:
H04L 29/06 (2006.01) *H04L 12/22* (2006.01)

(21) International Application Number:
PCT/US2006/027037

(22) International Filing Date: 11 July 2006 (11.07.2006)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/698,053 11 July 2005 (11.07.2005) US
11/320,593 30 December 2005 (30.12.2005) US

(71) Applicant *(for all designated States except US)*: **NORTEL NETWORKS LIMITED** [CA/CA]; 2351 Boulevard Alfred-Novel, St. Laurent, Quebec H4S 2A9 (CA).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **KUMAR, Ravi, Chakravarthi** [US/US]; 7957 Folkstone Drive, Cupertino, CA 95014 (US). **LAVIAN, Tal, I.** [IL/US]; 1294 Caldwell Court, Sunnyvale, CA 94087 (US). **SAHAY, Vasant** [US/US]; 1546 Grackle Way, Sunnyvale, CA 94087 (US). **DAS, Nirmalendu** [IN/US]; 20364 Gillick Way, Cupertino, CA 95014 (US). **KUNJUKUNJU, Biju, Sajibahavan** [IN/US]; 10639 Maplewood Road #D, Cupertino, CA 95014 (US). **LEVI, David, Burton** [US/US]; 3501 Kesterwood Drive, Knoxville, TN 37918 (US). **MICHELET, Philippe** [FR/US]; 972 Courtland Court, Milpitas, CA 95035 (US).

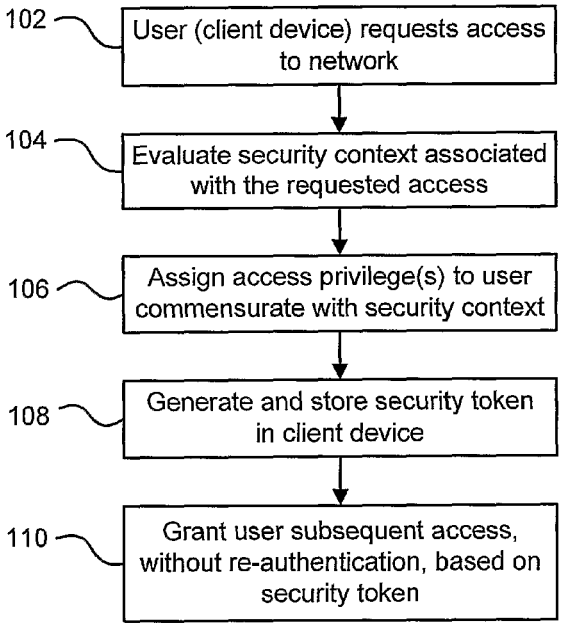
(74) Agents: **ANDERSON, Thomas, E.** et al.; Hunton & Williams LLP, Intellectual Property Department, 1900 K Street, N.w., Suite 1200, Washington, DC 20006-1109 (US).

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

[Continued on next page]

(54) Title: TECHNIQUE FOR AUTHENTICATING NETWORK USERS



(57) Abstract: A technique for authenticating network users is disclosed. In one particular exemplary embodiment, the technique may be realized as a method for authenticating network users. The method may comprise receiving, from a client device, a request for connection to a network. The method may also comprise evaluating a security context associated with the requested connection. The method may further comprise assigning the client device one or more access privileges based at least in part on the evaluation of the security context.

WO 2007/008976 A1

European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, LV, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report*

TECHNIQUE FOR AUTHENTICATING NETWORK USERS

CROSS-REFERENCE TO RELATED APPLICATIONS

This patent application claims priority to U.S. Provisional Patent Application No. 60/698,053, filed July 11, 2005, which is hereby incorporated by reference herein in its entirety.

This patent application is related to U.S. Patent Application No. 11/320,603, entitled "Technique for Providing Secure Network Access," filed December 30, 2005, which is incorporated herein in its entirety.

FIELD OF THE DISCLOSURE

The present disclosure relates generally to network security and, more particularly, to a technique for authenticating network users.

BACKGROUND OF THE DISCLOSURE

To prevent unauthorized access, it is often necessary for a network to authenticate its users to ensure that each user is who he or she claims to be. Conventional user authentication methods typically involve a brief interaction between a user and a network, wherein the user provides to the network a security identifier such as a secret password, a token device, a digital certificate, a biometric key, or a combination thereof. The network then verifies the security identifier against records of authorized users.

Conventional user authentication methods only produce a binary result - pass or fail. That is, if a user provides a security identifier that cannot be verified by the network, the user will be denied access completely. If the user's security identifier can be successfully verified, the user is often granted full access to the network. In some networks,

each authorized user may have predetermined access privileges also known as a "role." In this type of network, conventional user authentication methods still produce a binary result. That is, if the user is authenticated, he or she is assigned a predetermined role in the network. If the user is not authenticated, he or she will be completely locked out.

Except for a user-provided security identifier, conventional user authentication methods typically do not take into account any other factors in its decision to grant or deny access. That is, as long as a user enters a correct set of username and password, the user will be granted full access or a predetermined access privilege. In other words, conventional user authentication methods only care about who the user is, and do not pay attention to the circumstances in which the user accesses the network. Such conventional user authentication methods may make the network vulnerable to virus infections and/or malicious attacks. For example, a client device infected with virus may easily gain access to the network and put other devices at a greater risk of infection.

In addition, it is generally assumed that a network cannot trust client devices from which end-users access the network. Therefore, once a user disconnects from the network, the user's authentication with the network expires. The next time the user attempts to access the network, the user has to be re-authenticated. Even if the user does not leave the network but simply moves from one part of the network to another, the user may also have to go through a re-authentication process. To a network user, re-authentication can be inconvenient and sometimes annoying. For example, when roaming within a network, in each new location, a user may have to close some networked applications, get re-authenticated, and then restart the networked applications. As a result, in-network mobility may be burdened even for a

legitimate user of the network.

Another problem with conventional user authentication methods lies in a general requirement that a client device requesting access to a network must be compatible with the authentication scheme supported by the network. A traditional network typically supports only one particular authentication scheme, which may be based on, for example, IEEE 802.1x standard, a Media Access Control (MAC) or Internet Protocol (IP) database, or Remote Authentication Dial In User Service (RADIUS) protocol. Such a network can only authenticate a client device that is pre-configured to work with the network's chosen authentication scheme. For example, a network that only supports the IEEE 802.1x standard may not be able to authenticate a client device that employs the RADIUS protocol. Some networks go even further by requiring trusted, proprietary client software to be pre-installed in client devices. These compatibility requirements tend to block otherwise legitimate users with incompatible devices and may cause frustration or dissatisfaction in network users.

In view of the foregoing, it would be desirable to provide a technique for authenticating network users which overcomes the above-described inadequacies and shortcomings.

SUMMARY OF THE DISCLOSURE

A technique for authenticating network users is disclosed. In one particular exemplary embodiment, the technique may be realized as a method for authenticating network users. The method may comprise receiving, from a client device, a request for connection to a network. The method may also comprise evaluating a security context associated with the requested connection. The method may further comprise assigning the client device one or more access privileges based at least in part on the evaluation of the security context.

In accordance with other aspects of this particular exemplary embodiment, the security context may be evaluated at least in part by an agent program in the client device. The agent program may interact with the network to evaluate the security context. At least a portion of the security context may be evaluated prior to the request for connection. The agent program may comprise a JAVA applet. The agent program may be automatically downloaded to the client device upon receipt of the request for connection. In addition, the agent program may remain in the client device, after the client device disconnects from the network, in preparation for a subsequent connection to the network.

In accordance with further aspects of this particular exemplary embodiment, the security context may comprise one or more factors selected from a group consisting of: a user login mechanism employed by the client device, a threat level associated with the network, vulnerabilities of an access medium with which the client device connects to the network, and a security level associated with the client device.

In accordance with additional aspects of this particular exemplary embodiment, the method may further comprise generating a security token that records the one or more access privileges assigned to the client device and storing the security token in the client device. The method may also comprise detecting the security token in the client device when the client device, after ending a first connection to the network, attempts a second connection to the network and granting the client device access to the network based on the one or more recorded access privileges if the security token is detected and verified. The first and the second may connection to the network are through different ports.

In accordance with a further aspect of this particular exemplary embodiment, the method may further comprise generating a security token that records at least a portion of

the security context and storing the security token in the client device. The method may also comprise: detecting the security token in the client device when the client device, after ending a first connection to the network, attempts a second connection to the network, and granting the client device access to the network based at least in part on the recorded security context if the security token is detected and verified. The recorded security context may be updated prior to the client device's attempt of the second connection to the network.

In accordance with a yet further aspect of this particular exemplary embodiment, the method may comprise configuring a connection between the client device and the network based at least in part on the evaluation of the security context. The method may also comprise re-configuring the connection between the client device and the network based at least in part on a security token stored in the client device.

In another particular exemplary embodiment, the technique may be realized as at least one signal embodied in at least one carrier wave for transmitting a computer program of instructions configured to be readable by at least one processor for instructing the at least one processor to execute a computer process for performing the method as recited above.

In yet another particular exemplary embodiment, the technique may be realized as at least one processor readable carrier for storing a computer program of instructions configured to be readable by at least one processor for instructing the at least one processor to execute a computer process for performing the method as recited above.

In still another particular exemplary embodiment, the technique may be realized as a system for authenticating network users. The system may comprise a network interface

that facilitates communications between a client device and a network. The system may also comprise at least one processor that receives, from a client device, a request for connection to the network, causes a security context associated with the requested connection to be evaluated, and assigns the client device one or more access privileges based at least in part on the evaluation of the security context.

In another particular exemplary embodiment, the technique may be realized as a method for authenticating network users. The method may comprise receiving, from a client device, a request for connection to a network. The method may also comprise identifying a communication protocol employed by the client device. The method may further comprise adopting an authentication scheme that is compatible with the communication protocol, if the compatible authentication scheme is available for use by the network to authenticate the client device. The method may additionally comprise downloading an agent program to the client device if the compatible authentication scheme is not available, wherein the agent program interacts with the network to authenticate the client device.

In accordance with other aspects of this particular exemplary embodiment, the compatible authentication scheme may be selected from a group consisting of: authentication schemes associated with IEEE 802.1x standard, authentication schemes based on one or more Media Access Control (MAC) address lists, authentication schemes based on one or more Internet Protocol (IP) address lists, and authentication schemes based on Remote Authentication Dial In User Server (RADIUS) protocol.

The present disclosure will now be described in more detail with reference to exemplary embodiments thereof as shown in the accompanying drawings. While the present disclosure is described below with reference to exemplary embodiments, it should be understood that the present

disclosure is not limited thereto. Those of ordinary skill in the art having access to the teachings herein will recognize additional implementations, modifications, and embodiments, as well as other fields of use, which are within the scope of the present disclosure as described herein, and with respect to which the present disclosure may be of significant utility.

BRIEF DESCRIPTION OF THE DRAWINGS

In order to facilitate a fuller understanding of the present disclosure, reference is now made to the accompanying drawings, in which like elements are referenced with like numerals. These drawings should not be construed as limiting the present disclosure, but are intended to be exemplary only.

Figure 1 shows a flow chart outlining an exemplary method for authenticating network users in accordance with an embodiment of the present disclosure.

Figure 2 shows a flow chart illustrating an exemplary method for authenticating network users in accordance with an embodiment of the present disclosure.

Figure 3 shows a flow chart illustrating another exemplary method for authenticating network users in accordance with an embodiment of the present disclosure.

Figure 4 shows a flow chart illustrating an exemplary method for enhancing network security in accordance with an embodiment of the present disclosure.

Figure 5 shows a block diagram illustrating an exemplary system for authenticating network users in accordance with an embodiment of the present disclosure.

DETAILED DESCRIPTION OF EXEMPLARY EMBODIMENTS

Exemplary embodiments of the present disclosure are described below. In one particular exemplary embodiment, an authentication technique may take into account a security context associated with a requested connection between a

client device and a network. The security context may be evaluated through an interaction between the network and an agent program on the client device. Based on the evaluation of the security context, one or more access privileges (or "roles") may be assigned to the client device. The security context and/or the assigned access privilege(s) may be recorded in a security token which may be stored in the client device for use in a subsequent access to the network. The client device may later be permitted to re-connect to the network, without re-authentication, if the security token in the client device can be detected and verified.

As used herein, the term "network" refers to one or more interconnected devices, such as routers, switches and servers. In most instances, a network includes a plurality of interconnected devices that form, for example, one or more local area networks (LANs) and/or wide area networks (WANs). In some instances, however, a network may include a single device, such as a single computer as a host or a server. Many of the authentication functions as described below may be performed either by a central or dedicated device (e.g., a central or dedicated computer) or by a plurality of devices (e.g., access-point switches).

Referring to Figure 1, there is shown a flow chart outlining an exemplary method for authenticating network users in accordance with an embodiment of the present disclosure.

In step 102, a user or a client device may request access to a network. The user may request access to the network by logging in with, for example, a secret password or other security identifier. Typically, the user may request access from a client device such as, for example, a personal computer (PC), a UNIX terminal, a wireless personal digital assistant (PDA) device, or even a mobile telephone. Alternatively, the access request may be made by a client device on its own, that is, without intervention of a user. In either case, from the

network's perspective, it is the client device that is directly communicating with the network.

If the user or client device provides login information (e.g., a username and password) in the access request, the network may perform some initial verification of the user or client device identity. Such identification may provide a basis for subsequent processes. For example, if the user or client device cannot be verified at all, access to the network may be immediately denied. According to some embodiments, however, the access request may not necessarily involve a login mechanism or any input of security identifiers. The client device may simply be plugged into a port of the network, be detected inside the network, or transmit a message to a network element requesting connection.

In step 104, a security context associated with the requested access or connection may be evaluated. The security context may comprise information related to potential security issues that might arise out of a connection between the client device and the network. For example, the security context may include information related to one or more of the following factors: a user login mechanism, current or projected network threat levels, vulnerabilities of the access medium used in the proposed connection, and a security level of the client device. The security context may indicate how big a threat to the network the client device might be if it is allowed to connect to the network.

The evaluation of the security context may be carried out in a number of ways. Most preferably, the evaluation of the security context may be performed through an interaction between the network and the client device so that security information of both the network and the client device may be covered. For example, an agent program may be downloaded to the client device, if such a program is not yet available therein, and the agent program may interact with the network

to evaluate the security context. Alternatively, the evaluation of the security context may be performed by either the network or the client device alone. Timing of the evaluation of the security context may also be flexible. For example, an agent program may scan the client device for potential security threats even before the client device requests to access the network. Thus, at least a portion of the security context may be available when a new network connection is attempted. Time-sensitive security information may be obtained at or near the time when a new connection is attempted.

In step 106, the user or client device may be assigned one or more access privileges that are commensurate with the current security context. That is, a role may be assigned to the user or client device according to a potential threat level estimated for the requested connection. Generally, a higher potential threat level may require a more restrictive role for the client device. For example, if it is determined that the client device does not have the most up-to-date client software, or if viruses are found in the client device, a "read-only" or "browse-only" restriction may be imposed on the client device's access privileges. If the user logged in with both a token device and a secret password, the user may be given more access rights than if he or she only provided the secret password. If the client device is plugged into a physical port in a known secure location, the client device may be assigned more access rights than if it were roaming in a non-secure wireless area. From these examples, it may be understood that a role assigned to a network user or a client device may not be a predetermined one. Rather, the role may vary based on the security context in which the user or the client device connects to the network. Since the assignment of access privileges may be adaptive to the current security conditions, it may provide a better protection against virus

infections and malicious attacks.

In step 108, a security token may be generated for storage in the client device. The security token may be an electronic record, preferably encrypted, that comprises the security context and/or the access privileges assigned to the client device. The security token may be generated by the network and downloaded to the client device. Alternatively, the security token may be generated by the agent program in the client device and stored therein. A replicate of the security token or its content may also be stored in the network.

In step 110, if, after disconnecting from the network, the client device attempts another connection, the network may grant the client device access privileges, without re-authentication, based on the security token. For example, if the network detects the security token and verifies its content, the network may assign the client device access privileges that are the same as or similar to those assigned in a previous connection. The security token may or may not be the same as the one downloaded to the client device in its previous connection. According to one embodiment, the agent program may continue to monitor the client device and update the security token. Therefore, in a subsequent connection, the network may assign a new role to the client device based on the updated security context in the security token.

Figure 2 shows a flow chart illustrating an exemplary method for authenticating network users in accordance with an embodiment of the present disclosure.

In step 202, a user may log into a network from a client device. Upon the user's successful login, an agent program, such as a JAVA applet, may be automatically downloaded to the client device in step 204.

In step 206, the agent program may interact with the network to evaluate a security context associated with the

requested access. For example, the agent program may scan the client device for viruses, spywares or other malicious programs. The agent programs may also verify the hardware and software environment of the client device (e.g., trusted operation system, updated security patches, etc.). Coordinating with the network, the agent program may also identify vulnerabilities of the access medium via which the client device connects to the network. In addition, the network may report its current security conditions to the agent program. Based on the security information, the agent program may generate a threat level for the proposed connection. The threat level may be simple pass-or-fail assessment. Preferably, the threat level may be presented on a scale of multiple values to describe the security context with more granularity.

In step 208, access privileges may be assigned to the client device based at least in part on the threat level generated in step 206. Some embodiments of the present disclosure do not exclude the possibility of predetermined access privileges or roles. For example, a predetermined role (e.g., "administrator," "general user," or "guest") may be associated with a username. However, after a user with this username logs in, the access privileges associated with the predetermined role may be either maintained or modified depending on the threat level. Alternatively, there may be no predetermined roles and the access privileges may be assigned on an ad hoc basis (i.e., user-by-user and connection-by-connection). In addition to the assignment of access privileges, the connection between the client device and the network may be configured based on the evaluation of the security context. For example, a client device from a high-risk location may be required to communicate with the network through heavily encrypted messages. Conversely, for a client device connected to a trusted port of the network, encryption

requirement may be less stringent. Further, a rate limit may be imposed on traffic between the client device and the network based on the potential threat level. Access privilege for a specific user may be enforced from a policy manager (or server) based on the user authentication status and security context of the client device.

In step 210, a security token may be generated. The security token may record at least a portion of the security context and the threat level, the access privileges assigned to the client device, and/or other information related to the current connection (e.g., encryption mechanism, transfer rate limit, etc.).

In step 212, the security token may be stored in the client device. For example, the security token may have been generated in the network and then downloaded to the client device. While stored in the client device, the security token is preferably protected from unauthorized tampering. Alternatively, or in addition, the network may maintain a copy or a digital signature of the security token's content so that any unauthorized changes to the security token may be detected.

In step 214, the agent program downloaded to the client device may remain therein in preparation for a subsequent connection to the network. That is, the agent program may become persistent in the memory or other storage of the client device even if it is disconnected from the network. The agent program may continue to monitor the client device or otherwise perform security scan(s) before or at its next attempt to access the network. Security scans such as virus or spyware scans are typically time-consuming processes. Therefore, it may be desirable to perform these scans prior to a subsequent connection to the network so that at least that portion of the security context will be available for evaluation when needed.

Referring to Figure 3, there is shown a flow chart

illustrating another exemplary method for authenticating network users in accordance with an embodiment of the present disclosure.

In step 302, a user or a client device may attempt to connect to a network. In step 304, it may be determined whether a security token is present in the client device.

If a security token is present in the client device, then, in step 306, the network may verify the security token, for example, against a digital signature or a copy of a valid security token. If the security token is successfully verified, its contents may be extracted. For example, a security context (related to last connection or subsequently updated) may be among the security token's contents. Based on the security context, the network may, in step 308, instantly assign access privileges to the client device without re-authentication. The process may then continue in step 318.

If a security token is not present in the client device, it may be determined, in step 310, whether an agent program (known as a "Tunnel Guard") is present in the client device. If a Tunnel Guard is not present, one copy of the agent program may be downloaded to the client device in step 312, and the process may then continue in step 316. If a Tunnel Guard is present in the client device, then, in step 314, the agent program may report its pre-scanned security context information to the network.

In step 316, the agent program may interact with the network to evaluate a security context associated with the current connection, and access privileges may be assigned to the client device based on the evaluation.

In step 318, a new security token may be generated and downloaded to the client device. Alternatively, an existing security token in the client device may be updated with the new security context or access privileges.

In addition to the convenience of re-admitting network

users or client devices without re-authentication, embodiments of the present disclosure may also help improve network security when the users or devices are connected to the network. Figure 4 shows a flow chart illustrating an exemplary method for enhancing network security in accordance with an embodiment of the present disclosure.

In step 402, one or more security threats, such as virus attacks or denial of service (DoS) attacks, may be detected in a network that employs the authentication technique according to embodiments of the present disclosure. There may be one or more client devices connected to the network, wherein the client devices have been authenticated and have security tokens stored therein.

In step 404, the network may start to implement a heightened security policy in light of the security events. According to one embodiment, a policy server may be employed for security management. The policy server may be triggered by one or more security events detected by the agent program in the client device or by one or more network-level alerts. The policy server may then enforce specialized policies designed for a specific client, port, or network segment.

In step 406, client devices in the network may be polled for their respectively updated security context. Each client device may have a persistent agent program that keeps updating its security token with most up-to-date security information. Upon receipt of a polling instruction, the agent program may report a current security context to the network.

Based on the security context collected from the client devices, it may be determined which client devices, network devices/ports, or portions of the network are more vulnerable to the detected security threats or more critical to the spread of the security threats. Accordingly, the network may take coordinated measures to battle or mitigate the security threats. For example, in step 408, those client devices at a

high risk may be temporarily blocked from accessing the network or may be required to go through re-authentication. Certain critical devices or portions of the network may be quarantined. In step 410, network traffic on certain high-risk ports or devices may be rate-limited. In step 412, encryption mechanisms for some devices may be changed from light encryption to heavy encryption.

Figure 5 shows a block diagram illustrating an exemplary system 500 for authenticating network users in accordance with an embodiment of the present disclosure. The system 500 may comprise a network having a switch 504, a first server 506, a WAN 508, a second server 510, an Ethernet medium 512, and wireless routers 514. The switch 504 and the first server 506 may provide physical network connections for a client device 502 in a first area 516. The first server 506 may be coupled to the second server 510 via the WAN 508. The second server 510 may be coupled to the wireless routers 514 via the Ethernet medium 512. The wireless routers 514 may provide network access to the client device 502 in a second area 518.

A user working on the client device 502 may establish a first connection 51 to the network via a first port on the switch 504. For the connection 51, the user may be authenticated and assigned access privileges based on a security context associated with the client device 502 and the connection 51. A security token containing the security context and/or the assigned access privileges may be downloaded to the client device 502.

Then, if the user decides to move from location A to location B inside the first area 516, for example, from the user's office to a conference room, the connection 51 may have to be terminated. According to embodiments of the present disclosure, when the client device 502 is unplugged from the switch 504, there may be no need to terminate any networked applications on the client device 502. In location B, the

user may plug the client device 502 back to the network, e.g., on a different port of the switch 504. The switch 504 (or the server 506) may recognize the security token in the client device 502. Therefore, the network may re-admit the user and establish a new connection 52 without re-authenticating the user. The networked applications may continue running on the client device 502 without restarting.

Later on, if the user decides to roam even further on the network, the user may unplug the client device 502 to terminate the connection 52 and move to the second area 518. The client device 502 may establish another connection 53 with the wireless routers 514, again without re-authentication or restarting the networked applications. The routers 514 may detect and verify the security token in the client device 502. For example, a copy or a digital signature of the security token generated earlier may be stored in the server 506 or another central storage device. The routers 514 (or the server 510) may retrieve the stored security token (or its digital signature) and use it to verify the security token found in the client device 502.

According to some embodiments of the present disclosure, a network or a network element therein may be adapted to support multiple authentication schemes. The multiple authentication schemes may include one or more schemes based on the IEEE 802.1x standard. The multiple authentication schemes may also include one or more schemes based on a local or centralized MAC or IP address list. That is, a network element such as an edge switch or an access controller/server may maintain one or more lists of trusted MAC or IP addresses. A client device may be authenticated by verifying its MAC or IP information against the trusted list(s). Further, the multiple authentication schemes may include one or more schemes in which a Tunnel Guard (TG) may be dynamically downloaded to a client device to facilitate authentication

and/or other functions. When a client device requests access to the network, one of the multiple authentication schemes may be dynamically adopted based on a determination of the client device's type and/or behavior. For example, if it is determined that the client device supports a particular authentication scheme that is also supported by the network, this particular authentication scheme may be adopted. Otherwise, the network may adopt a fallback scheme to authenticate the client device. Table 1 lists exemplary authentication schemes that may be adopted according to the client device's type and behavior.

Table 1. Exemplary Authentication Schemes Based On Client Device Type and Behavior

Client Device Type & Behavior	Adopted Authentication Scheme	Fallback Authentication Scheme
802.1x Device (Dynamic IP i.e. device does DHCP)	If the device starts speaking 802.1x, authentication will be done by 802.1x. If the device does not starts speaking 802.1x, authentication will be done by TG.	When the 802.1x or TG authentication fails the switch may still try to authenticate the client device using a local or centralized MAC database.
Non 802.1x Device (Dynamic IP)	Authentication will be done by TG.	When the TG authentication fails the switch may still try to authenticate the client device using a local or centralized MAC database.
802.1x Device (Static IP i.e. device does not do DHCP)	Authentication will be done by 802.1x.	When the 802.1x authentication fails the switch will still try to authenticate the client device using a local or centralized MAC+IP database.

Non 802.1x Device (Static IP)	Authenticate the client device using a local or centralized MAC+IP database.	Block access.

At this point it should be noted that the technique for authenticating network users in accordance with the present disclosure as described above typically involves the processing of input data and the generation of output data to some extent. This input data processing and output data generation may be implemented in hardware or software. For example, specific electronic components may be employed in a computer and/or communications network or similar or related circuitry for implementing the functions associated with authentication of network users in accordance with the present disclosure as described above. Alternatively, one or more processors operating in accordance with stored instructions may implement the functions associated with authentication of network users in accordance with the present disclosure as described above. If such is the case, it is within the scope of the present disclosure that such instructions may be stored on one or more processor readable carriers (e.g., a magnetic disk), or transmitted to one or more processors via one or more signals.

The present disclosure is not to be limited in scope by the specific embodiments described herein. Indeed, other various embodiments of and modifications to the present disclosure, in addition to those described herein, will be apparent to those of ordinary skill in the art from the foregoing description and accompanying drawings. Thus, such other embodiments and modifications are intended to fall within the scope of the present disclosure. Further, although

the present disclosure has been described herein in the context of a particular implementation in a particular environment for a particular purpose, those of ordinary skill in the art will recognize that its usefulness is not limited thereto and that the present disclosure may be beneficially implemented in any number of environments for any number of purposes. Accordingly, the claims set forth below should be construed in view of the full breadth and spirit of the present disclosure as described herein.

CLAIMS

1. A method for authenticating network users comprising the steps of:

receiving, from a client device, a request for connection to a network;

evaluating a security context associated with the requested connection; and

assigning the client device one or more access privileges based at least in part on the evaluation of the security context.

2. The method according to claim 1, wherein the security context is evaluated at least in part by an agent program in the client device.

3. The method according to claim 2, wherein the agent program interacts with the network to evaluate the security context.

4. The method according to claim 2, wherein at least a portion of the security context is evaluated prior to the request for connection.

5. The method according to claim 2, wherein the agent program comprises a JAVA applet.

6. The method according to claim 2, wherein the agent program is automatically downloaded to the client device upon receipt of the request for connection.

7. The method according to claim 6, wherein:

the agent program remains in the client device, after the client device disconnects from the network, in preparation for a subsequent connection to the network.

8. The method according to claim 1, wherein the security context comprises one or more factors selected from a group consisting of:

a user login mechanism employed by the client device;

a threat level associated with the network;

vulnerabilities of an access medium with which the client device connects to the network; and

a security level associated with the client device.

9. The method according to claim 1, further comprising:

generating a security token that records the one or more access privileges assigned to the client device; and

storing the security token in the client device.

10. The method according to claim 9, further comprising:

detecting the security token in the client device when the client device, after ending a first connection to the network, attempts a second connection to the network; and

granting the client device access to the network based on the one or more recorded access privileges if the security token is detected and verified.

11. The method according to claim 10, wherein the first and the second connections to the network are through different ports.

12. The method according to claim 1, further comprising:

generating a security token that records at least a portion of the security context; and

storing the security token in the client device.

13. The method according to claim 11, further comprising:

detecting the security token in the client device when

the client device, after ending a first connection to the network, attempts a second connection to the network; and

granting the client device access to the network based at least in part on the recorded security context if the security token is detected and verified.

14. The method according to claim 13, wherein the recorded security context is updated prior to the client device's attempt of the second connection to the network.

15. The method according to claim 1, further comprising:

configuring a connection between the client device and the network based at least in part on the evaluation of the security context.

16. The method according to claim 15, further comprising:

re-configuring the connection between the client device and the network based at least in part on a security token stored in the client device.

17. At least one signal embodied in at least one carrier wave for transmitting a computer program of instructions configured to be readable by at least one processor for instructing the at least one processor to execute a computer process for performing the method as recited in claim 1.

18. At least one processor readable carrier for storing a computer program of instructions configured to be readable by at least one processor for instructing the at least one processor to execute a computer process for performing the method as recited in claim 1.

19. A system for authenticating network users, the system comprising:

a network interface that facilitates communications between a client device and a network; and

at least one processor that

receives, from a client device, a request for connection to the network;

causes a security context associated with the requested connection to be evaluated; and

assigns the client device one or more access privileges based at least in part on the evaluation of the security context.

20. A method for authenticating network users, the method comprising the steps of:

receiving, from a client device, a request for connection to a network;

identifying a communication protocol employed by the client device;

adopting an authentication scheme that is compatible with the communication protocol, if the compatible authentication scheme is available for use by the network to authenticate the client device; and

downloading an agent program to the client device if the compatible authentication scheme is not available, wherein the agent program interacts with the network to authenticate the client device.

21. The method according to claim 20, wherein the compatible authentication scheme is selected from a group consisting of:

authentication schemes associated with IEEE 802.1x standard;

authentication schemes based on one or more Media Access Control (MAC) address lists;

authentication schemes based on one or more Internet Protocol (IP) address lists; and

authentication schemes based on Remote Authentication Dial In User Server (RADIUS) protocol.

Figure 1

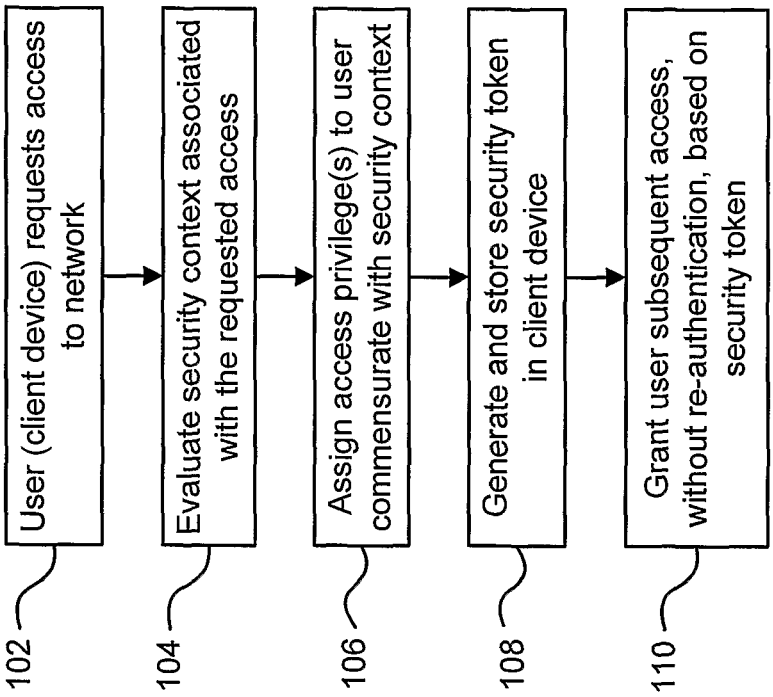

User (client device) requests access to network
Evaluate security context associated with the requested access
Assign access privilege(s) to user commensurate with security context
Generate and store security token in client device
Grant user subsequent access, without re-authentication, based on security token
102
104
106
108
110

Figure 2

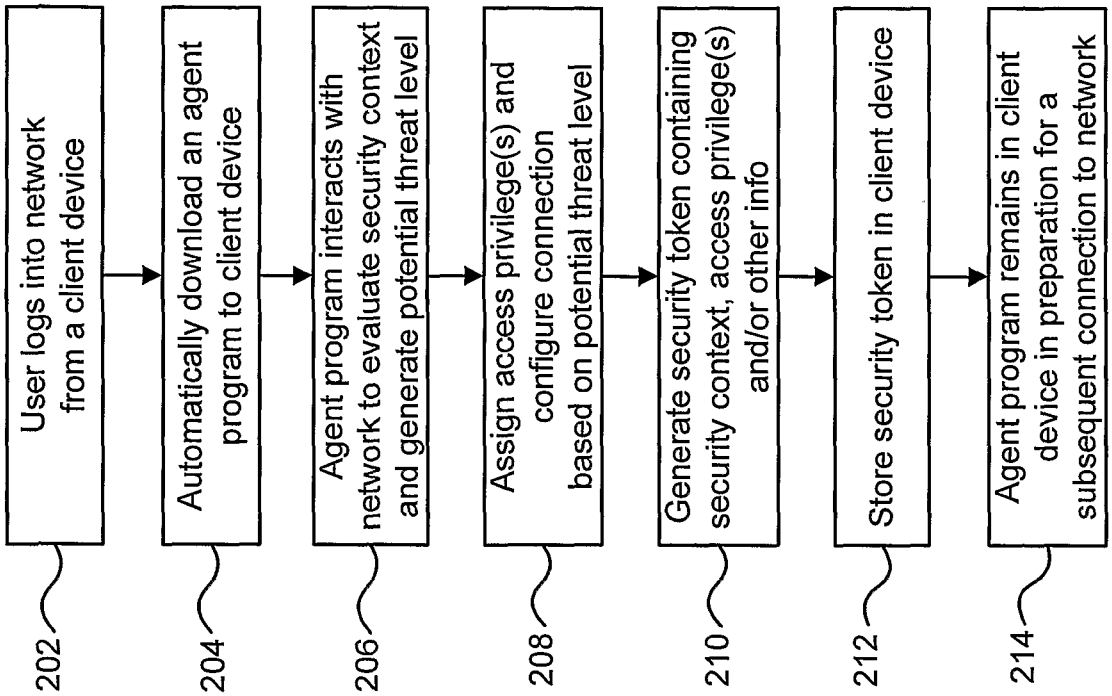

User logs into network from a client device
Automatically download an agent program to client device
Agent program interacts with network to evaluate security context and generate potential threat level
Assign access privilege(s) and configure connection based on potential threat level
Generate security token containing security context, access privilege(s) and/or other info
Store security token in client device
Agent program remains in client device in preparation for a subsequent connection to network
202
204
206
208
210
212
214

Figure 3

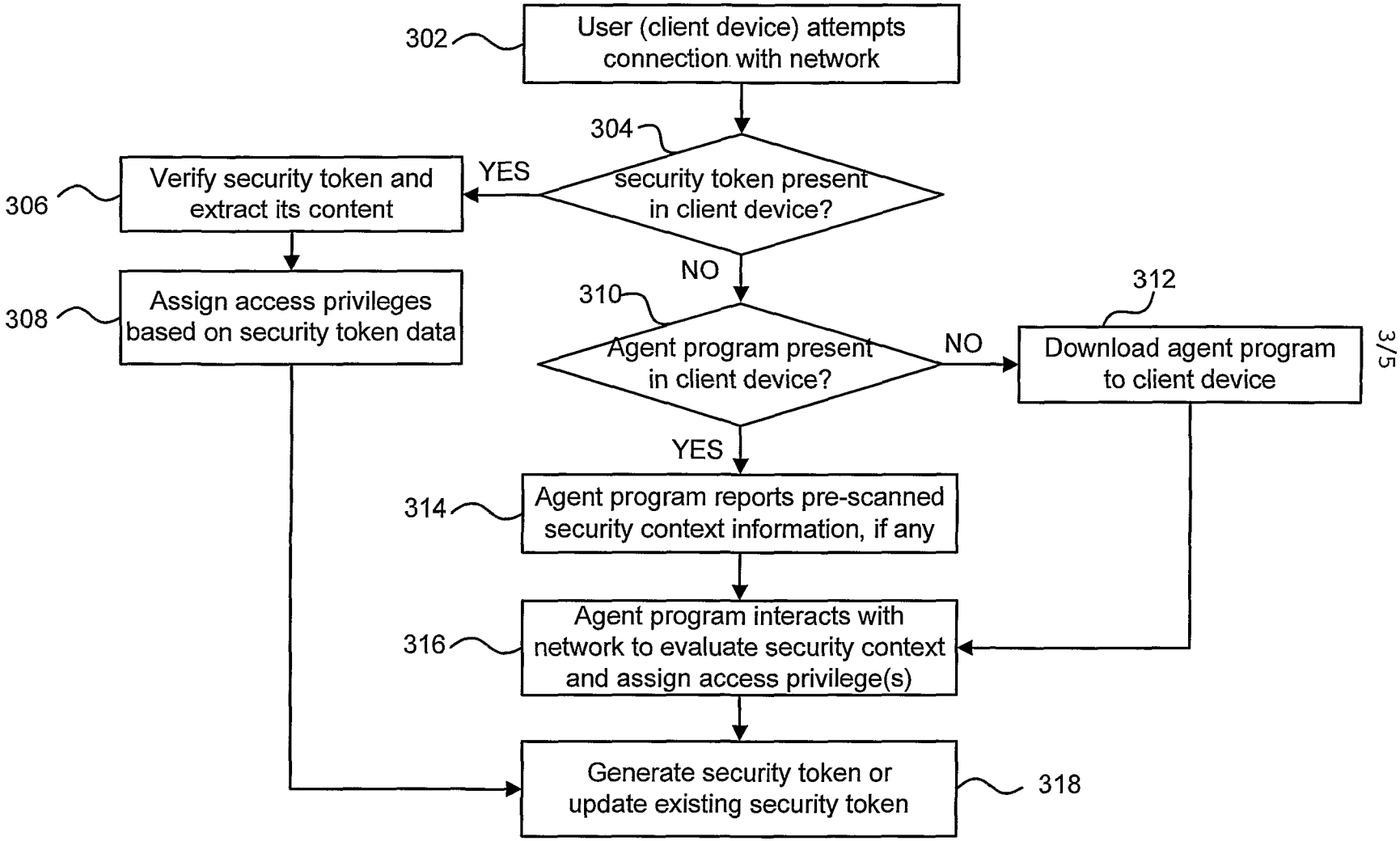
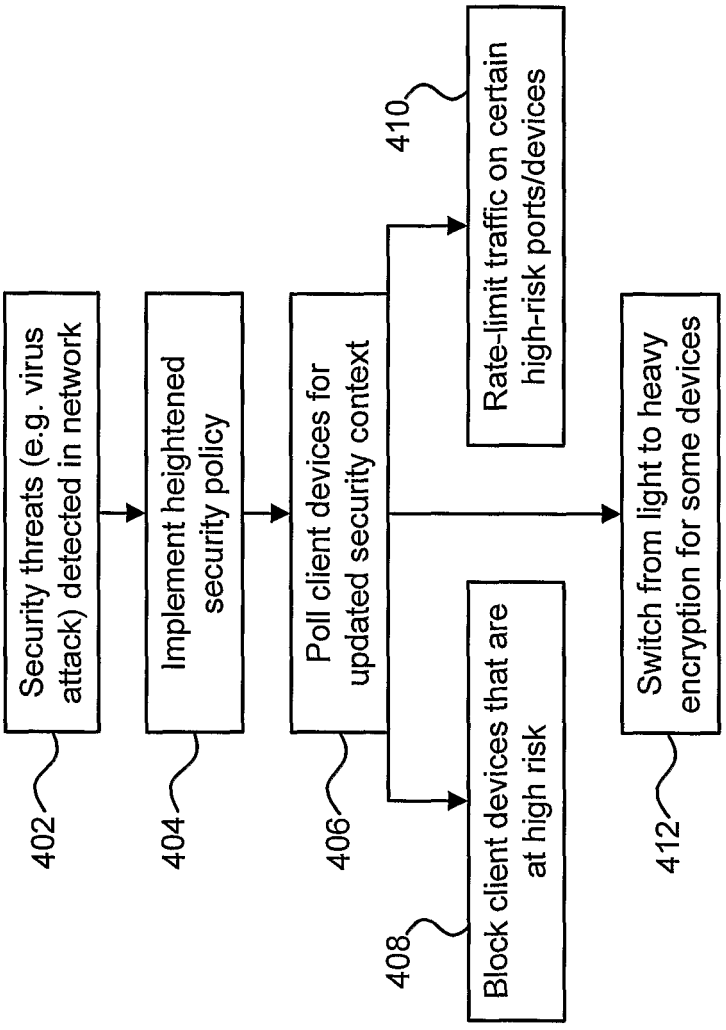

302
User (client device) attempts connection with network
304
security token present in client device?
YES
306
Verify security token and extract its content
NO
308
Assign access privileges based on security token data
310
Agent program present in client device?
NO
312
Download agent program to client device
YES
314
Agent program reports pre-scanned security context information, if any
316
Agent program interacts with network to evaluate security context and assign access privilege(s)
Generate security token or update existing security token
318

Figure 4


Security threats (e.g. virus attack) detected in network
Implement heightened security policy
Poll client devices for updated security context
Rate-limit traffic on certain high-risk ports/devices
Block client devices that are at high risk
Switch from light to heavy encryption for some devices
402
404
406
408
410
412

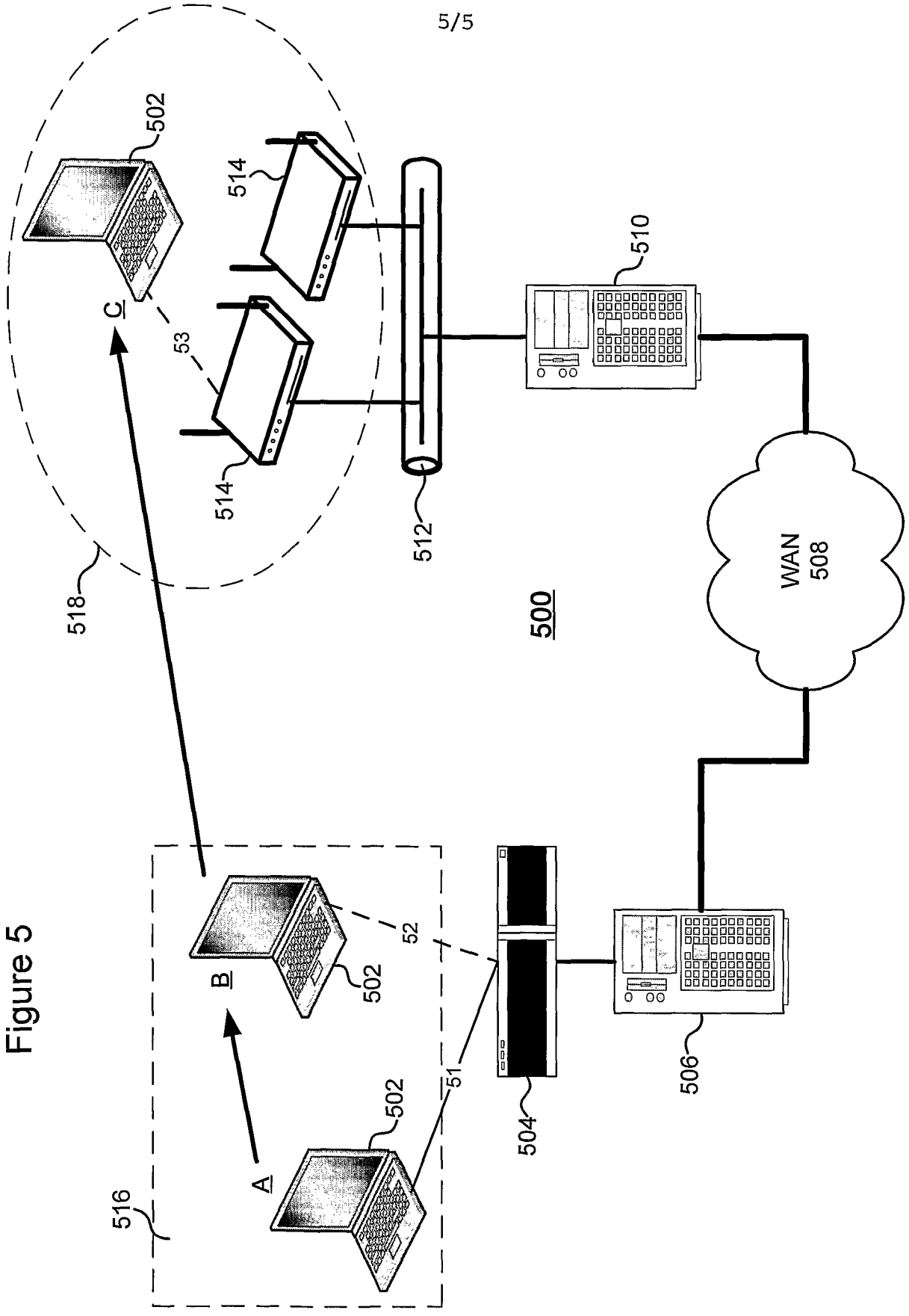

502
514
C
53
514
510
518
512
500
WAN
508
B
502
52
502
A
516
51
504
506

Figure 5

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2006/027037

A. CLASSIFICATION OF SUBJECT MATTER
INV. H04L29/06 H04L12/22

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)
EPO-Internal

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	EP 1 339 199 A (HEWLETT PACKARD CO [US]) 27 August 2003 (2003-08-27) paragraphs [0019] - [0033]	1-21
X	EP 0 465 016 A (DIGITAL EQUIPMENT CORP [US] DIGITAL EQUIPMENT CORP [DE]) 8 January 1992 (1992-01-08) column 1, lines 36-53 column 2, line 18 - column 3, line 15 column 4, line 26 - column 5, line 28	1-21
A	WO 02/03178 A2 (INTERNET SECURITY SYSTEMS INC [US]) 10 January 2002 (2002-01-10) page 3, line 27 - page 4, line 5 page 13, lines 4-18; figure 7	1-21

☐ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
17 October 2006	25/10/2006

Name and mailing address of the ISA/	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL – 2280 HV Rijswijk Tel. (+31–70) 340–2040, Tx. 31 651 epo nl, Fax: (+31–70) 340–3016	Veen, Gerardus

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/US2006/027037

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP 1339199 A	27-08-2003	US 2004083394 A1	29-04-2004
EP 0465016 A	08-01-1992	CA 2044003 A1	26-12-1991
		DE 69130657 D1	04-02-1999
		DE 69130657 T2	22-07-1999
		JP 1996980 C	08-12-1995
		JP 6095991 A	08-04-1994
		JP 7031648 B	10-04-1995
		US 5204961 A	20-04-1993
WO 0203178 A2	10-01-2002	AU 6509801 A	14-01-2002
		EP 1311921 A2	21-05-2003
		JP 2004509387 T	25-03-2004