

EX 1025

EXHIBIT 4

UNITED STATES DISTRICT COURT
NORTHERN DISTRICT OF CALIFORNIA
OAKLAND DIVISION

NETSKOPE, INC.,

Plaintiff,

v.

FORTINET, INC.,

Defendant.

Case No. 4:25-cv-02360-HSG

EXPERT REPORT OF AARON D. STRIEGEL

Aaron D. Striegel, hereby declare as follows:

I. INTRODUCTION

1. I am currently a Professor in the Department of Computer Science and Engineering at the University of Notre Dame. With respect to this report, I have been retained as a technical expert by counsel for Netskope, Inc. (“Netskope”) to address claim construction issues regarding U.S. Patent Nos. 8,356,336 (the “’336 Patent”), 8,543,710 (the “’710 Patent”), 8,117,639 (the “’639 Patent”), 8,224,983 (the “’983 Patent”), 8,327,426 (the “’426 Patent”), 7,593,936 (the “’936 Patent”), 8,397,282 (the “’282 Patent”), 8,661,153 (the “’153 Patent”), and 8,635,697 (the “’697 Patent”) in the above captioned case.

2. I am being compensated for my time in this case at the rate of \$875 per hour (plus expenses) for services rendered as an expert witness in this case. My compensation for this matter is not determined by or contingent on the outcome of this case.

3. I am willing to testify at the hearing set by the Court regarding this declaration if asked about these matters by the Court or the parties' attorneys.

4. I reserve the right to supplement my opinions, as well as the bases for my opinions, expressed herein in light of any additional materials, including opinions expressed by other witnesses and/or other evidence that may be provided to me after submission of this declaration. This includes any opinions offered by Defendants’ experts on the subject of invalidity.

II. QUALIFICATIONS

5. I am currently a Professor in the Department of Computer Science and Engineering at the University of Notre Dame. I received my Ph.D. in Electrical and Computer Engineering in 2002 from Iowa State University in Ames, Iowa. I received my Bachelor’s degree in Computer Engineering in 1998 from Iowa State University in Ames, Iowa. I have served in my current role as Professor in the Department of Computer Science and Engineering at the University of Notre Dame in Notre Dame since May 2018. Prior to that period, I served as an Associate Professor from May 2009 to May 2018, and prior to that, as an Assistant Professor (tenure track) from January 2003 through May 2009.

6. From 2012 through 2018, I served as the Associate Chair in the Department of Computer Science and Engineering overseeing junior faculty mentoring, external degree program accreditation via ABET (the professional accreditation body for engineering, including Computer Science via CSAB), and other administrative duties for the department. In 2011 and 2017, I led the reaccreditation of the Computer Science and Computer Engineering programs for the Department of Computer Science and Engineering at the University of Notre Dame with ABET. From 2020 through 2025, I served as the Program Director for the Bachelor of Arts in Computer Science which is a joint program between the Department of Computer Science and Engineering and the College of Arts and Letters at the University of Notre Dame. I also served on the organizing committee establishing a joint Business / Computer Science degree between the College of Business and the College of Engineering at the University of Notre Dame.

7. I have supervised seventeen students through completion of their Ph.D. degrees, eight students through completion of their Masters degrees, and numerous undergraduates conducting research. Primary focus areas of the various dissertations and theses are outlined in my CV (Ex. 1) and are oriented around my primary areas of research (computer networking, computer security).

8. I have instructed over thirty courses across my time as an instructor between Iowa State University and the University of Notre Dame. Courses have ranged from an introductory course for freshmen on engineering concepts to graduate level courses on operating systems and security. Additionally, I co-taught an MOOC (Massive Online Open Course) to over eleven thousand students entitled “Understanding Wireless” offered as a joint effort between edX and the University of Notre Dame. I have participated in multiple curriculum revisions including a recasting of the lab of the embedded system course at Iowa State University (C, Assembly), a conversion of the introductory programming sequence at the University of Notre Dame (from Scheme to C/C++), and piloted new courses on computer security, human-computer interaction, graduate networks—also at the University of Notre Dame.

9. I have served on numerous Technical Program Committees (TPCs) for top conferences in my field of research. For example, I served as the general chair for ICCCN on its 25th International Conference on Computer Communications and Networks, which was held in August 2016. For the past several years, I have served amongst a select group of nearly 40 individuals serving as Area Chairs for IEEE INFOCOM which is one of the premier research venues in the area of computer networking. Notably, I recently served as one of the Technical Program Committee (TPC) co-chairs for INFOCOM 2025, one of the flagship conferences in computer networking and will be serving as General Chair of IEEE MASS 2026. I have also served on numerous panels and federal grant review efforts. Recently, I served on a panel on the Internet of Things at MASS 2025. Previously, I have served on a panel at INFOCOM 2018 concerning the impact of artificial intelligence (AI) on networking research. Prior examples include serving on a panel regarding large-scale instrumentation of mobile devices at CSET 15 (8th Workshop on Cyber Security and Test). In the past few years, I have given a keynote regarding data collection and smart device challenges at the SmartWork workshop in October 2021 and a keynote on network measurement at UEMCON 2022.

10. I have been a senior member of the Institute of Electrical and Electronics Engineers (IEEE) since 2017 and a full member since 2002. I am also a full member of both the Association of Computing Machinery (ACM) and the American Society for Engineering Education (ASEE), which I joined in 2006 and 2011, respectively.

11. I served as a co-PI (Principal Investigator) of the Industrial University Cooperative Research Center (IU/CRC) sponsored by the National Science Foundation at the University of Notre Dame, of which Notre Dame is a site under the Broadband Wireless Applications Center (BWAC). I have participated in considerable industrial outreach including funding from Sprint, Google, Alcatel-Lucent, Hewlett-Packard, and Sun Microsystems. I was a founding member of the Wireless Institute at the University of Notre Dame and serve as part of the Executive Committee. Past and current collaborators for the Wireless Institute include Sprint, Alcatel-

Lucent, InterDigital, National Instruments, the Office of Naval Research, and the Lab for Telecom Sciences.

12. I have been involved in the submission of two provisional patents, both of which involved my students as the co-inventors. Provisional patent #60/752,945 generally related to automatic detection at the network level of rogue wireless access points at the edge of the enterprise network. Provisional patent #61/628,883 generally related to techniques for the transparent extraction of redundancy in wireless communications for the purpose of improving wireless (specifically WiFi) performance.

13. I am also credited as one of two inventors on U.S. Patent No. 10,383,002, titled, *"Systems and Methods for Rapidly Estimating Available Bandwidth in a WiFi link,"* awarded May 2019. The patent generally pertains to techniques for determining the available bandwidth of a WiFi link using frame aggregation between an access point and an electronic communication device.

14. I have published thirty-eight peer-reviewed journal papers and over one hundred conference papers in the areas of computer networking, computer security, engineering education, and real-time systems. Out of those papers, three papers have received best paper awards at conferences. My work has been cited by research papers and other scholarly materials over 4500 times.

15. My research group has produced numerous, large-software artifacts available to researchers and interested parties including code bases across C, C++, C#, Java, MATLAB, Python, Swift, Go, and Tcl. Notable software products produced by our research group include a fully functional agent for Android proximity / communication monitoring (PhoneMonitor, Java), a full visualization suite for network security context (ENAVis, Java, C++), improved rehabilitation utilizing a combination of the Nintendo Wii balance board and Microsoft Kinect (WeHab, C#), an integrated software suite for capturing smartphone, wearable, and beacon sensing streams (Tesseract, multiple languages), implementations for video streaming utilizing QUIC (Go),

and a full network programming suite for layer two to layer seven interactions (ScaleBox, C++). All of the efforts involve multiple years of development and range on the order of thousands to hundreds of thousands of lines of code. I have supervised / am in the process of supervising millions of dollars (over \$17M) of funding during my time at the University of Notre Dame for research purposes relating generally to networks, network security, and sensing devices. Funding sources include government agencies (National Science Foundation (NSF), National Institute of Health (NIH), DARPA, and IARPA), private foundations (National Academies / Keck Foundation), and industrial support.

16. I completed several industrial internships during my graduate and undergraduate work. In 1996 (January through August), I worked in a cooperative work program with Fisher Controls (now Emerson Process) in Marshalltown, IA. In 1997 and 1998 (both May to August), I continued with additional summer internships also at Fisher Controls. The work at Fisher Controls included development in C++ and C supporting GUI development and communications tool development (HART, Fieldbus). In 2000 (May to August), I worked for Architecture Technology Corporation, a small research firm in Eden Prairie, MN. My tasks included development in C++ on a fully functional suite of a network gateway capable of providing DHCP and MobileIPv4 support. In 2002 (January to May), I worked for the Universite de Technologie Compiegne in Compiegne, France in partnership with Motorola. My duties there included the development of a specification for distributed admission control support with C++ and Tcl support for simulation in ns2 for the purposes of mobile resource management. In 2002 (May to August), I worked for Sun Microsystems in Mountain View, CA. My duties included design document creation (Rational Rose) and the creation of a fully functional emulation testbed for the evaluation of network processors via Layer 2 through Layer 4 rule bases with all code written in Java.

17. During my time as an instructor (both as a Ph.D. student and as a professor), I have drawn extensively on those industrial experiences in teaching the various code and system-oriented classes. I have taught 11 courses that involved significant aspects of C programming (CSE 20211, CPRE 211, CSE 30341, CSE 60641, over 800 students), four courses that involved significant aspects of object-oriented (CSE 20211, CSE C++, C#) programming (over 150 students), six courses that had significant systems components with considerable C programming (over 225 students, CSE 30341, CSE 30342, CSE 60641), two courses with considerable introductory programming (MATLAB) (800 students, EG 10111), and four courses with robust assembly components (nearly 500 students, CPRE 211). Furthermore, my programming and system notes were published as an open note set. Finally, I have taught three courses centered on networking (CSE 30289, CSE 60744, CSE 60884— one at the undergraduate level, two at the graduate level) and one course on security (CSE 498U— cross-listed between undergraduate / graduate). Cumulatively, I have over 20 years of coding experience (industry, graduate school, faculty member) in C and C++ programming. I have over 20 years of experience with computer networks and over 18 years of experience with computer security topics.

18. My curriculum vitae is attached as Exhibit A.

III. SUMMARY OF OPINIONS

19. It is my opinion that a person of ordinary skilled in the art would understand that the term “subordinate to” is not indefinite in light of the intrinsic evidence, including the claims and the specification. In light of the claims and the specification of the ’710 patent, a person of ordinary skilled in the art would understand that the term “subordinate to” has a definite meaning within the context of the ’710 patent.

20. It is my opinion that a person of ordinary skilled in the art would understand that the term "principal believing interactions are with the external service" is not indefinite in light of the intrinsic evidence, including the claims and the specification. In light of the claims and the specification of the ’426 patent, a person of ordinary skilled in the art would understand that the

term “principal believing interactions are with the external service” has a definite meaning in the context of the ’426 patent.

21. It is my opinion that a person of ordinary skilled in the art would understand that the term "generic response" is not indefinite in light of the intrinsic evidence, including the claims and the specification. In light of the claims and the specification of the ’936 patent, a person of ordinary skilled in the art would understand that the term “generic response” has a definite meaning within the context of the ’936 patent.

22. It is my opinion that a person of ordinary skilled in the art would understand that the term "arbitrary number of attributes" is not indefinite in light of the intrinsic evidence, including the claims and the specification. In light of the claims and the specification of the ’153 patent, a person of ordinary skilled in the art would understand that the term “arbitrary number of attributes” has a definite meaning in the context of the ’153 patent.

23. It is my opinion that a person of ordinary skilled in the art would understand that the term "a more reliable OS ID" is not indefinite in light of the intrinsic evidence, including the claims and the specification. In light of the claims and the specification of the ’697 patent, a person of ordinary skilled in the art would understand that the term “a more reliable OS ID” has a definite meaning in the context of the ’697 patent.

IV. LEVEL OF ORDINARY SKILL IN THE ART

24. In determining who would be a person of ordinary skill, I considered at least the following criteria: (a) the type of problems encountered in the art; (b) prior art solutions to those problems; (c) the rapidity with which innovations are made; (d) the sophistication of the technology; and (e) the education level of active workers in the field.

25. In my opinion, a person of ordinary skill in the art relevant to the ’710 patent, ’426 patent, ’936 patent, ’153 patent, and ’697 patent would possess at least the equivalent of a college degree in electrical engineering, telecommunication, computer engineering, computer science, or a related technical degree, as well as 2-3 years of experience in engineering in related fields.

V. LEGAL STANDARD

26. I have been informed and understand that the words of a claim are generally given their ordinary and customary meaning to a person of ordinary skill in the art in question at the time of the invention. The person of ordinary skill in the art is deemed to read the claim term not only in the context of the particular claim in which the disputed term appears, but in the context of the entire patent, including the specification. Hence, patent claims must be read in view of the specification.

27. I have been informed and understand that the presence of modifiers may also be taken into account in evaluating whether a claim term has sufficient structural meaning.

28. I understand that claim terms should usually be read and understood in a manner that gives meaning to all of the words in the claim and, further, gives each independent claim a distinct meaning and scope from its dependent claims. Thus, other claims (and claim elements) of the patent in question, both asserted and unasserted, can inform the meaning of a claim term. For example, because claim terms are normally used consistently through the patent, the usage of a term in one claim can often illuminate the meaning of the same term in other claims. Differences among claims can also be a useful guide in understanding the meaning of particular claim terms.

29. I understand that, under some circumstances, an inventor may clearly define a claim term in the specification or the prosecution history to mean something other than its ordinary meaning.

30. I understand that each claim of an issued patent is presumed to be valid and definite, and a party challenging the validity of a patent claim must present clear-and-convincing evidence to overcome that presumption. A patent claim is invalid for indefiniteness only if the challenging party can demonstrate by clear-and-convincing evidence that the claim, read in light of the patent's specification and prosecution history, does not inform with reasonable certainty those skilled in the art about the scope of the invention.

VI. THE '710 PATENT

A. OVERVIEW OF THE TECHNOLOGY

31. The '710 Patent addresses challenges in network access control systems, in which, based on security levels, a device either had full access to a network or was completely blocked. This framework not only lacked flexibility, such as by constraining clients exhibiting abnormal behavior, the user of the infected client device may also be completely unaware of the type of malicious infection or how to remediate it.

32. The '710 Patent presents embodiments that solve these challenges by allowing a “service provider to use the network access gateway to constrain a network client’s level of network access gateway to constrain a network client’s level of network access in a way that can inform the user of the problem and still allow the user access to a limited set of destinations that may be helpful in resolving the problem.” Ex. 2 at 1:61-66.

33. Specifically, the '710 Patent discloses systems and methods for a rules-based access system which uses controlled environments, known as “walled gardens,” which “limit the network segments or routes available, as well as the network protocol traffic permitted.” *Id.* at 3:65. By implementing traffic shaping discrimination techniques, “traffic from a client can then be subjected to a particular list of access rules that specify a ‘walled garden’ which define where network access is allowed.” *Id.* at 3:67-4:3.

34. For example, Figure 1 demonstrates an embodiment of the '710 Patent:

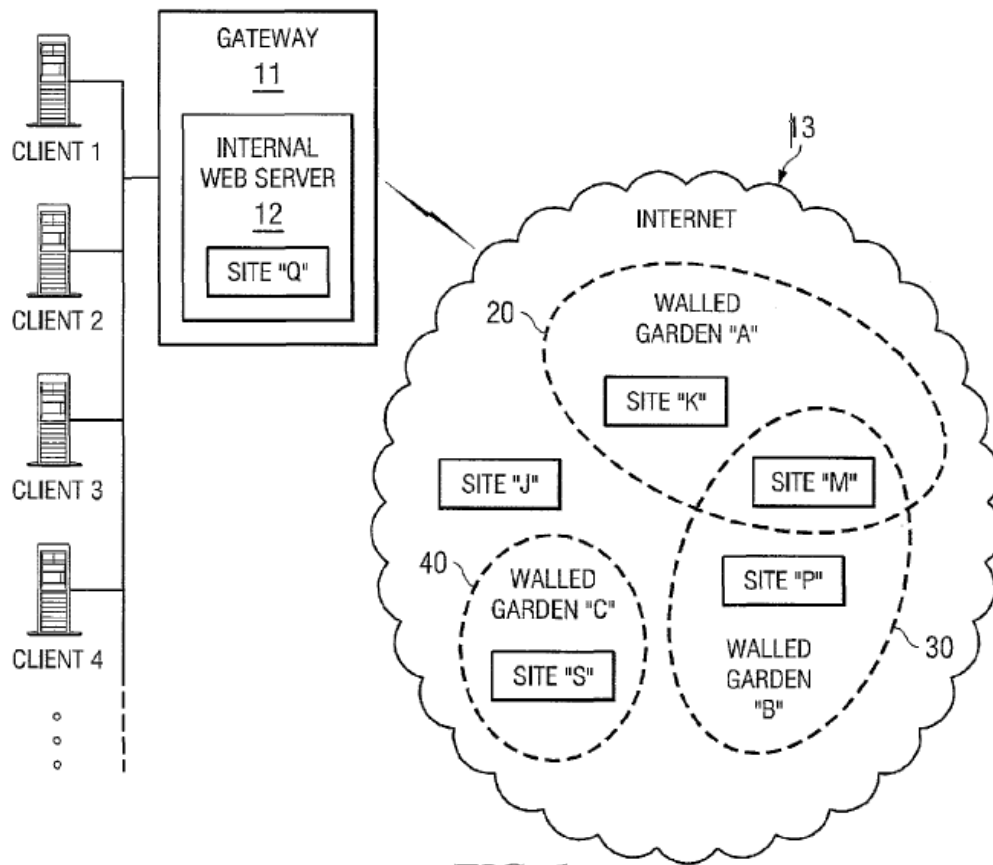


FIG. 1

Ex. 2, Fig. 1.

35. For example, Figure 1 demonstrates an embodiment of the '710 Patent comprising a network topology with a LAN and gateway connected to the Internet. "Gateway 11 may be operable to support any number of walled gardens 20, 30, 40 and any number of client classifications." *Id.* at 4:40-42. "Clients 1, 2, 3, 4 on LAN network 10 are connected to Internet 13 via gateway 11." *Id.* at 4:31-32. "Gateway 11 can assign clients 1, 2, 3, 4 to walled gardens 20, 30, 40 automatically based on facts known or learned at gateway 11, or the assignment can be done in response to external configuration or commands received at gateway 11." *Id.* at 4:42-46. Specifically, network access gateway 11 inspects incoming network traffic for attributes that associate the packet with a particular client. *Id.* at 4:47-49. Once the "incoming traffic is

discriminated (and/or marked) on a per-client basis, traffic can be directed to, assigned to, or associated with, one or more sets of client-specific (or group-specific) access roles. *Id.* at 4:55-59.

36. Once a client is assigned to a specific walled garden, any traffic emanating from that client that is not addressed to a location in the walled garden “may have the traffic dropped, denied, or redirected by gateway 11.” *Id.* at 4:66-5:3. The ’710 Patent also discloses that in other embodiments, “when this errant traffic is a web browser request, gateway 11 can redirect the request thus causing display of an informational page in the requesting user’s browser.” *Id.* at 5:3-6. This informational page can “inform the user of his status in the access gateway and offer choices of action.” *Id.* at 5:9-10. This process allows the user to seek remediation of the computer infection “without requiring direct support from the service provider.”

VII. CONSTRUCTION OF THE ’710 PATENT TERMS

A. “subordinate to”

37. I understand that the parties have proposed the following constructions for this term:

Netskope	Fortinet
Plain and ordinary meaning	Indefinite

38. The term “subordinate to” appears in claims 1, 6, 8, 13, 15 and 20 of the ’710 Patent. The term appears in claim 1 as follows:

1. A method of network traffic quarantine control, comprising:
 - at a network access gateway device between a local network and the Internet, selecting a client device in a first network segment of the network;
 - at the network access gateway device, performing a plurality of quarantine control functions over the client device, wherein the plurality of quarantine control functions comprises:
 - a) restricting all network traffic emanating from the client device to one or more network destination addresses that are not in or ***subordinate to*** the first network segment;

b) restricting all network traffic emanating from the client device to an allowed network destination address to selected one or more network protocols; and

rendering a web page to display on the client device from the network access gateway device, wherein the web page contains an offer for a user of the client device to perform an action in order to obtain unrestricted access to the Internet responsive to implementation of one of the plurality of quarantine control function of the client device.

39. I understand that Fortinet argues that the term “subordinate to” is indefinite. However, I disagree. It is my opinion that a person of ordinary skill in the art would understand that the term “subordinate to” is not indefinite in light of the intrinsic evidence, including the claims and specification.

40. The '710 Patent's specification describes portions of a network (e.g., “network segments”) which can be used to confine a network user's access to a specific region of the network. Ex. 2 at 1:58-60. This confinement can be used as a way to quarantine a client with suspected malicious behavior to inform the user of the detected anomaly and direct the user with actionable steps to resolve the problem. *Id.* at 1:61-67. In other embodiments, the confinement may be repurposed into a paywall or a barrier requiring additional authentication before gaining full access to the network. *Id.* at 2:1-4.

41. Figure 1 demonstrates how various portions of the network are segmented from each other:

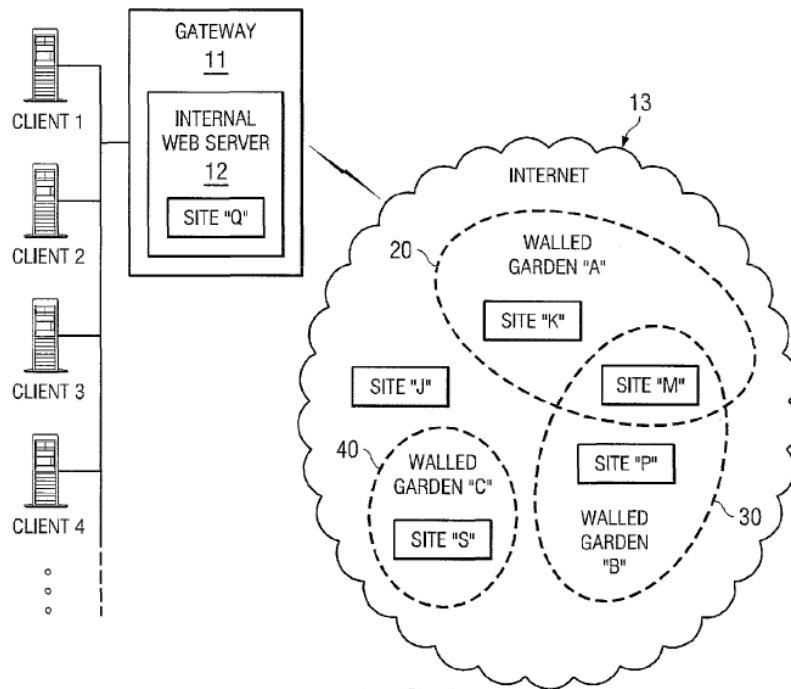


FIG. 1

Ex. 2, Fig. 1.

42. As shown in Figure 1, “Gateway 11 may be operable to support any number of walled gardens 20, 30, 40 and any number of client classifications.” *Id.* at 4:40-42. “Gateway 11 can assign clients 1, 2, 3, 4 to walled gardens 20, 30, 40 automatically based on facts known or learned at gateway 11, or the assignment can be done in response to external configuration or commands received at gateway 11.” *Id.* at 4:42-46.

43. It is my opinion that Figure 1 provides sufficient support for a person of ordinary skill in the art to understand “subordinate to.” For example, Figure 1 shows Internet 13, comprising walled garden “A” with corresponding Site K and Site M destinations, walled garden “B” with corresponding Site P and Site M destinations—wherein Site M overlaps with walled gardens “A” and “B.” *Id.* at Fig. 1. Internet 13 also includes a standalone destination Site “J” and

a separate, unrelated walled garden, titled walled garden “C,” which contains destination Site “S.”
Id.

44. In the context of the specification and claims, a person of ordinary skill in the art would understand walled garden “C” to contain a network destination address that is not in or “subordinate to” walled gardens “A” and “B,” because, for example, these network segments are not hierarchically related, managed by, or logically dependent upon each other. In view of the specification, a network segment may be considered subordinate if it operates within the administrative, routing, or policy domain of another segment—regardless of physical or topological configuration. Applying this context to the figures, a person of ordinary skill in the art would understand Site “J” not to be in or “subordinate to” walled gardens “A,” “B,” and “C” or the sites located within those walled gardens because Site “J” is located outside of those network destinations. Accordingly, the plain language of the ’710 Patent provides sufficient information to inform a person of ordinary skill in the art on the meaning of “subordinate to.”

45. Further, the specification explains the reasoning behind the need for network destination addresses that are not in or subordinate to other network segments. The ’710 Patent informs a scenario in which client 1 becomes infected with malware. Ex. 2 at 5:54-57. Gateway 11 notices the abnormal behavior from client 1, and quarantines the client by “assigning it to a walled garden reserved for infected users,” allowing “traffic from this client to destinations outside the walled garden [to] become[] restricted.” *Id.* at 5:58-64. A person of ordinary skill in the art would understand that constraining the infected client to a walled garden that is not in or subordinate to another network segment to ensure the security of other devices on the network. Further, “these network regions may contain content or data intended to repair or enhance a network client’s network access and permit renewed access to the other portions of a network by the client.” *Id.* at 6:24-27.

46. Accordingly, it is my opinion that a person of ordinary skill in the art would understand the meaning of “subordinate to” in the context of the ’710 Patent’s specification, figures, and claims.

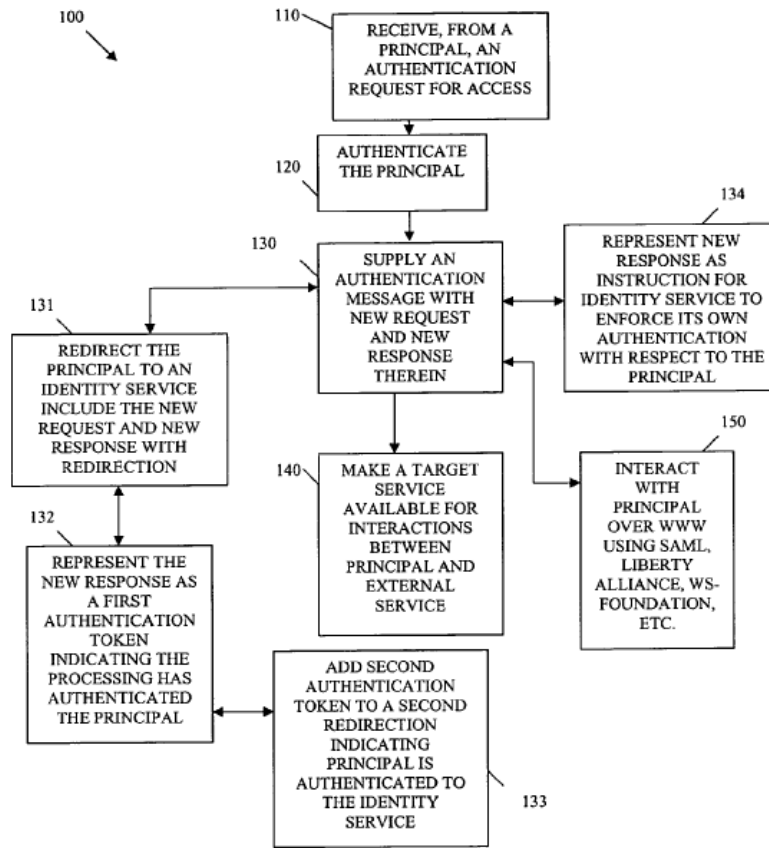
VIII. THE ’426 PATENT

A. OVERVIEW OF THE TECHNOLOGY

47. The ’426 Patent addresses challenges in networks when a user is attempting to access multiple services. Under traditional forms of authentication, a user often must authenticate separately to each online service because the user maintains separate credentials with each party. So, if a user were attempting to make a purchase with a vendor, the transaction would fail as the user’s banking credentials are distinct from the vendor. This method was cumbersome, as it required the user to maintain a variety of—even within the same transaction, and despite the fact that the user may already be authenticated by one service. This method was cumbersome and often resulted in friction and inconvenience for the user when attempting to complete a single transaction.

48. The ’426 Patent introduces techniques for single sign-on that facilitates seamless authentication across multiple services by using proxying techniques. For example, a user signs in to a primary identity service, which then generates an authentication statement or token that is trusted by a secondary identity service. The secondary identity service uses that token to authenticate the user without requiring additional credentials from the user. The outcome of this process is the user can access multiple trusted services without needing to re-enter credentials to access each subsequent identity service.

49. Fig. 1 illustrates an embodiment of the invention, describing a method for single sign-on with proxy services:



Ex. 3, Fig. 1.

50. As shown in Figure 1 at 110, the single sign-on service receives an authentication request for access from a principal; the request may be to access the network or to access a site within an identity service. Ex. 3 at 4:4-8. In other words, the principal may believe it is attempting to access a site within an identity service, and “the single sign-on intercepts the communication and attempts to authenticate the principal.” *Id.* at 4:9-14. At step 120, the single sign-on service authenticates the principal using a variety of authentication methods (e.g. user identification and password, certificate, biometrics, etc.). *Id.* at 4:17-23.

51. After the principal is authenticated, the single sign-on service provides an authentication message to the principal, comprising an authentication request and an authentication

response to satisfy the authentication request. *Id.* at 4:30-36. This authentication request is made on behalf of the principal that is targeted to another identity service, and allows the principal to be automatically authenticated to the identity service. *Id.* at 4:30-48.

52. Figure 1 also includes additional embodiments of the '426 patent. For example at 131, “the single sign-on service may facilitate the authentication of the principal to another identity service in response to successfully authenticating the principal using WWW browser interactions and interfaces.” *Id.* at 4:49-53. Under this embodiment, the principal’s browser is automatically redirected to the secondary identity service after the principal is authenticated. *Id.* at 4:53-57. And, as shown in Figure 1, that redirection also includes the new authentication request made on behalf of the principal and a new authentication response, created by the single sign-on service. *Id.* at 4:57-60. This results in the principal accessing the secondary identity service unbeknownst to the proxying techniques performed by the single sign-on service. “The targeted or secondary identity service, in response to the authentication response, may assume that the principal is to be automatically presumed authenticated for access to the targeted identity service.” *Id.* at 5:4-7. “This presumption can occur because the single sign-on service is trusted with and by the targeted identity service and if the single sign-on service vouches for the principal and indicates via an authentication message or via the authentication response that the principal is authenticated then the targeted identity service can assume or presume the principal to be authenticated to the targeted identity service.” *Id.* at 5:7-14.

B. CONSTRUCTION OF THE '426 PATENT TERMS

I. “principal believing interactions are with the external service”

53. I understand that the parties have proposed the following constructions for this term:

Netskope	Fortinet
Plain and ordinary meaning	Indefinite

54. The term “principal believing interactions are with the external service” appears in claim 1 of the ’426 Patent. The term appears in claim 1 as follows:

1. A machine-implemented method to execute on a machine, comprising:
 receiving, by the machine, an authentication request from a principal, the request directed by the principal to an external service and intercepted by the method for receipt;
 authenticating, by the machine, the principal; and
 supplying, by the machine, an authentication message for use by an identity service on behalf of the principal, the authentication message serves as a new authentication request and as a new authentication response for single sign-on access of the principal to the identity service and other services external or internal to the identity service, the identity service acts as a proxy for access sessions to the other services on behalf of the principal, the principal's access sessions occur indirectly through the identity service and transparently to the principal, wherein the authentication message includes the new authentication request made on behalf of the principal and the authentication message also includes a new authentication response that satisfies the new authentication request, that response vouches for authentication of the principal to the identity service for the single sign-on access of the principal, the **principal believing interactions are with the external service**, which is one of the other services that the identity service controls access to, and a determination as to whether to use a single interaction or multiple interactions for authentication of the principal to the other services is automatically communicated in the new authentication response.

55. I understand that Fortinet argues that the term “principal believing interactions are with the external service” is indefinite. I disagree. It is my opinion that a person of ordinary skill in the art would understand that the term “principal believing interactions are with the external service” is not indefinite in light of the intrinsic evidence, including the claims and specification.

56. Specifically, the specification describes that a single sign-on service performs proxying techniques in which a user (e.g., principal) could access a secondary identity service based on authentication performed by accessing a first identity service. Ex. 3 at 10:17-18. “In an

example embodiment, these interactions occur via a WWW browser interface...[t]he first identity service 502 authenticates the principal 501, perhaps through an interactive challenge and response dialogue with the principal 501, and automatically redirects the browser of the principal 501 to the second identity service 503 via B.” *Id.* at 10:25-32.

57. This process is further detailed in Figure 5:

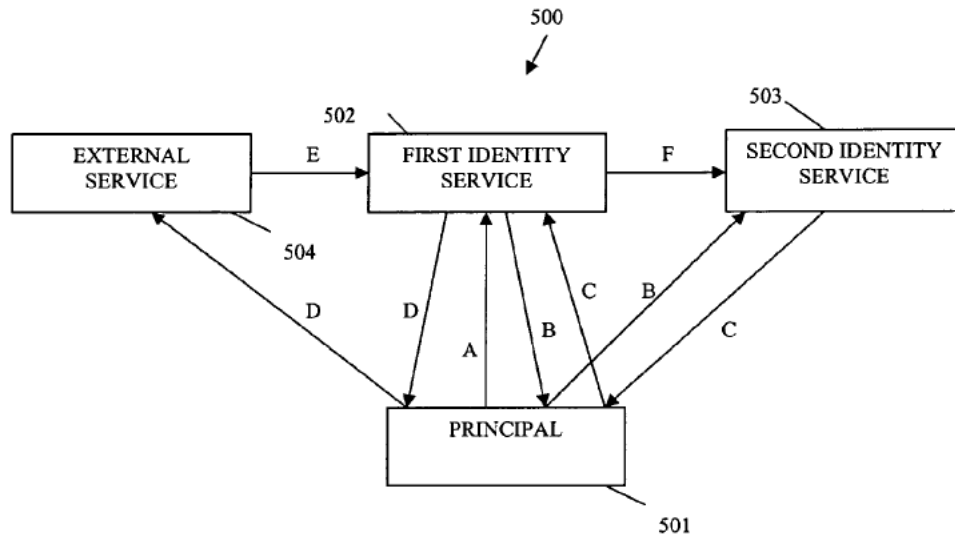


FIG. 5

Ex. 3, Fig. 5.

58. The second identity service “receives an authentication request formulated by the first identity service 502 on behalf of the principal 501.” *Id.* at 10:33-35. “That authentication request also includes a response from the first identity service 502 for the principal 501” which indicates that the “principal 501 is currently already authenticated to the first identity service 502.” *Id.* at 10:37-40. From this established authentication, the “second identity service 503 considers the principal 501 to be authenticated with the second identity service 501.” *Id.* at 10:40-42. “In response to the automatic authentication of the principal 501 to the second identity service 503, the second identity service 503 supplies a new authentication token to the principal 501 and redirects the browser of the principal 501 back to the first identity service 502 via C.” *Id.* 10:45-

49. These techniques are performed in a “seamless manner from the perspective of the parties involved.”

59. It is my opinion that the specification provides sufficient support for a person of ordinary skill in the art to understand “principal believing interactions are with the external service.” An advantage of the ’426 Patent is it allows users to authenticate once, while simultaneously gaining access to multiple services without requiring additional steps. This is further emphasized by the specification noting that “[t]he presence of the single sign-on service may be transparent or unknown to the principal” and “the principal may believe it is interacting or attempting to interact with an external service and the single sign-on intercepts the communication...” *Id.* at 4:9-13. Accordingly, this term describes how the identity service acts as a transparent proxy between the principal and an external service. In other words, the system is designed so that the principal perceives no difference between interacting directly with the external service and interacting through the proxy; the identity service handles requests and responses in a way that makes its mediation invisible. This enables a single sign-on framework across multiple services without the need for additional user logins or awareness of the proxying services occurring behind the scenes.

IX. THE ’936 PATENT

A. OVERVIEW OF THE TECHNOLOGY

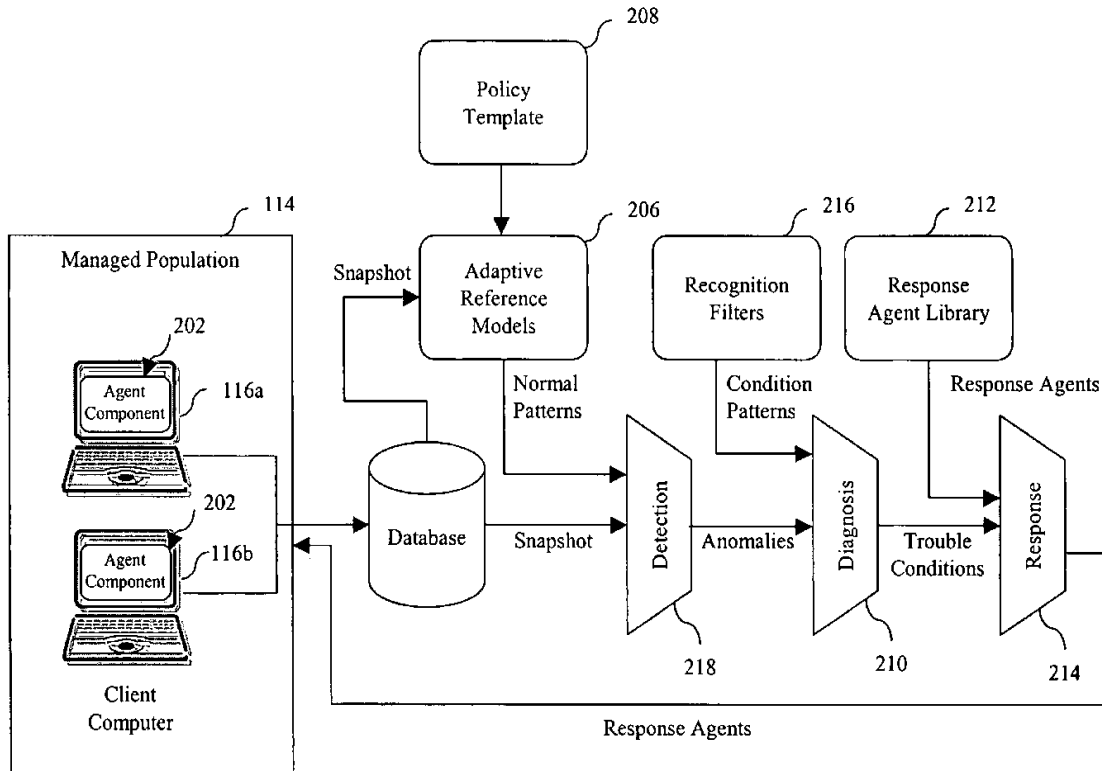
60. At a basic level, the ’936 Patent describes systems and methods for automated computer support that utilize computer snapshots and an adaptive reference model to detect abnormal system states in a population of computers. Ex. 4 at 3:41-51. The patent explains how expensive labor costs are for incident resolution, and how automated systems can reduce the costs by automating the discovery and correction of a problem before the user is even aware that a problem exists. *Id.* at 1:32-2:56.

61. The ’936 Patent also describes the prior art’s shortcomings. Conventional products like software distribution tools and configuration management tools attempt to mitigate incidents by limiting the managed population to known configurations and eliminating any signatures that

deviate from this configuration. *Id.* at 2:14-19. However, this approach is flawed because not all software is static, and certain products may update themselves automatically, which may trigger a response from conventional tools. *Id.* at 2:18-31. A fundamental challenge that these conventional tools consistently fail at is determining the scope and granularity of the allowed configurations. The conventional tools are generally incapable of being flexible enough to allow for updates to software located on the computer without activating a system response, while simultaneously being able to reliably distinguish between “normal” and “abnormal” system states on each computer in the population. *Id.* at 2:57-64. With there being a nearly endless number of possible software combinations and version permutations on any given computer, manually setting rules for “normal” or “abnormal” states will be close to impossible on conventional tools.

62. Additionally, the '936 Patent explains how self-service tools, such as knowledge bases and software solutions in the prior art, designed to assist the user with low-risk support functions, generally fail to resolve issues effectively and, as a result, fail at reducing the labor costs associated with computer repair. *Id.* at 2:65-3:29 These conventional tools routinely fall short in diagnosing issues across the computer population, both in the types of issues that can be identified and the accuracy of their diagnosis. *Id.* at 3:15-22. When these tools fail, resolving the issue often requires substantial service time from a skilled analyst, which would be costly. *Id.* at 3:18-29.

63. Fig. 2 illustrates a block diagram highlighting the process of an anomaly being detected through snapshots, diagnosed by the recognition filters, and the system sending a response agent to the computer to solve the anomaly according to one embodiment.



Ex. 4, Fig. 2.

64. Specifically, snapshots are analyzed by the adaptive reference model in step 206 to determine if there are any anomalies on the machine. The adaptive reference model is generated automatically using a plurality of snapshots from the plurality of computers in the population. *Id.* at 3:41-51. The adaptive reference model then uses data mining algorithms on snapshots that are generated periodically to determine if there is an anomaly on any computers within the population. *Id.* at 7:23-35. If there are, the recognition filters diagnose the anomalies by searching through known conditions and classifying them in step 210. *Id.* at 9:31-36. If the anomaly is recognized, then an associated response can be administered to the computer in step 214.

65. Anomalies are analyzed by grouping and may be related because they arise from a single root cause. *Id.* at 15:35-37. Anomalies that are grouped may be matched to a plurality of recognition filters that contain pairs of asset names and values that can help the user recognize threats like the presence of a virus on the computer. “A generic recognition filter may provide a template for creating more specific filters. For example, a recognition filter that is adapted to search for worms in general may be adapted to search for a specific worm.” *Id.* at 16:1-5.

66. After the condition has been identified, response agents execute scripts to correct the condition and remove the anomalies in step 214. *Id.* at 8:53-62.

B. CONSTRUCTION OF THE '936 PATENT TERMS

I. “generic response”

67. I understand that the parties have proposed the following constructions for this term:

Netskope	Fortinet
Plain and ordinary meaning	Indefinite

68. The term “generic response” appears in claims 3, 4, 14, and 15 of the '936 Patent. The term appears in claim 3 as follows:

3. The method of claim 2, wherein the automated response is a ***generic response*** not specific to a particular asset of the at least one computer, and wherein the method further comprises:
 sending the ***generic response*** and a set of anomalies found in the snapshot to the at least one computer, the set of anomalies indicating assets of the at least one computer whose states are anomalous; and
 applying the ***generic response*** to the anomalous assets to correct the trouble condition.

69. It is my belief that the term “generic response” has a plain and ordinary meaning within the context of the '936 Patent that would have been readily understood by one of ordinary skill in the art at the time of filing. A generic response provides flexibility by accommodating multiple computer configurations within a single system, while still enabling the troubleshooting

of anomalies common across entire population. Ex. 4 at 6:64-7:13. The automated response is considered “generic” because it can address multiple distinct anomalies that share a common characteristic—similar to how rebooting a computer can resolve various software issues that keep applications stuck mid-process.

70. I understand that Fortinet argues “generic response” is indefinite to a person of ordinary skill in the art. I disagree and believe that one of ordinary skill in the art would understand “generic response” in light of the specification, claim language, and knowledge of the field would have understood the scope and meaning of this term with reasonable certainty.

71. The '936 Patent illustrates the concept of a generic class or trait when it explains how anomalies are detected and removed in an embodiment of the invention as seen below:

“An anomaly is defined as an unexpectedly present asset, an unexpectedly absent asset, or an asset that has an unknown value. Anomalies are matched against a library of Recognition Filters 216. A Recognition Filter 216 comprises a particular pattern of anomalies that indicates the presence of a particular root cause condition **or a generic class of conditions**. Recognition Filters 216 also associate conditions with a severity indication, a textual description, and a link to a response agent. In another embodiment, a Recognition Filter 216 can be used to identify and interpret benign anomalies. For example, if a user adds a new application that the administrator is confident will not cause any problems, the system according to the present invention will still report the new application as a set of anomalies. If the application is new, then reporting the assets that it adds as anomalies is correct. However, the administrator can use a Recognition Filter 216 to interpret the anomalies produced by adding the application as benign.”

[8:12-29] (emphasis added)

“A recognition filter comprises a signature of a condition or of a class of conditions. For example a recognition filter may comprise a collection of pairs of asset names and values that, when taken together, represent the signature of a condition that is desirable to recognize, such as the presence of a worm. A generic recognition filter may provide a template for creating more specific filters. For example, a recognition filter that is adapted to search for worms in general may be adapted to search for a specific worm.”

[15:63-16:5]

72. Those generic class or conditions may have a root cause that can be identifiable. Specifically, the process includes comparing anomalies to recognition filters to determine whether the cause of the anomaly is known or unknown and subsequently dispatching an appropriate action. Within this process, the invention distinguishes between particular and generic causes of anomalies. The specification illustrates that when a generic cause is identified, a generic response may be dispatched as seen below.

“If a match is found between a subset of anomalies and a Recognition Filter, the name of the Recognition Filter will be associated with the subset of anomalies in the output stream. For example, the presence of a virus might generate a set of file anomalies, process anomalies, and registry anomalies. A Recognition Filter could be used to consolidate these anomalies so that the user would simply see a descriptive name relating all the anomalies to a likely common cause, i.e. a virus.

If automated response has been authorized, then the response agent library (212) downloads the appropriate response agents to the affected machine 320. The Agent component 202 in the affected machine then executes the sequence of scripts needed to correct the trouble condition 322.”

Id. at 9:28-42

73. As a multitude of different viruses are capable of doing something like corrupting a file, altering registry values, or changing computer processes, the generic responses may reverse those changes to correct the conditions. “The response agent library 212 allows the service provider to author and store automated responses for specific trouble conditions. These automated responses are constructed from a collection of scripts that can be dispatched to a managed machine to perform actions like replacing a file or changing a registry value.” *Id.* at 8:53-63

74. These possible generic responses are found not only in the specification but in Claim 4, which relies on Claim 3 and states:

4. The method of claim 3, wherein the generic response includes at least one of: installing a missing software component, removing an undesirable software component, and restoring an incorrect registry setting.

75. One of ordinary skill in the art would understand that these steps are helpful in a broad variety of troubleshooting issues and can solve many different conditions a computer may have. “Generic response” indicates breadth of possible steps the system can perform rather than indefiniteness.

76. For the reasons stated above, it is my opinion that a person of ordinary skill in the art would understand the plain meaning of “generic response.” The specification of the ’936 Patent describes this as a response that can be used to cure anomalies that may have many distinct root causes. Thus, the plain language of the ’936 Patent provides sufficient detail to inform a person of ordinary skill in the art on the meaning of this term.

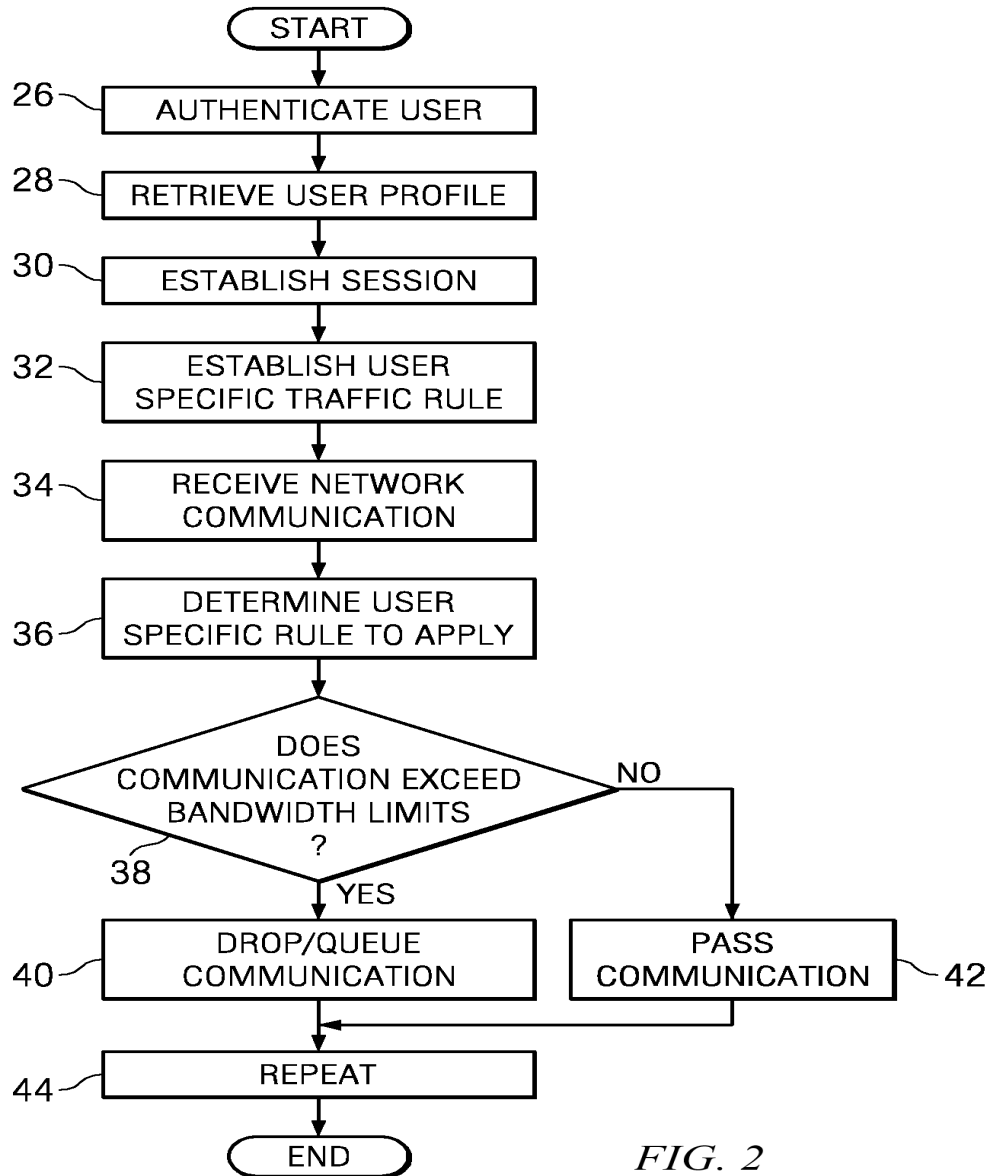
X. THE ’153 PATENT

A. OVERVIEW OF THE TECHNOLOGY

77. The ’153 Patent describes a method, control device, and system for controlling network access and allocating bandwidth on a per user basis. Ex. 5 at 2:15-20. Unlike prior art systems, which allocated bandwidth per port and lacked the ability to dynamically reprovision bandwidth to individual users, the patented system introduces bandwidth shaping, the process of allocating bandwidth, that calculates and adjusts bandwidth allocations dynamically for each user. This dynamic bandwidth shaping enables the system to reprovision bandwidth as additional users are added to the same port, ensuring consistent performance across the network. In contrast, earlier systems could not adapt bandwidth distribution at the user level, resulting in degraded performance for all users when network demand increased.

78. The ’153 Patent utilizes user profiles that are assigned to the user upon connection to the network and govern bandwidth allocation. Each user’s profile contains attributes that determine that user’s bandwidth allocation. Based on these user profiles, the user may have limitations on how long they may be connected to the network, specify maximum upload and download bandwidths, and when network communications are received. As network conditions or the user’s attributes change, the system can dynamically update the bandwidth allocation assigned to that profile.

79. Fig. 2 is a flow chart illustrating an embodiment of the method for providing dynamic bandwidth shaping when a user connects to a network.



Ex. 5, Fig. 2.

80. When the user has connected to the control device and been authenticated, the control device will retrieve the user's profile from either local or remote storage. *Id.* at 5:14-29. The user profile can contain an arbitrary number of attributes that govern bandwidth allocation for

that user. *Id.* at 5:24-43. These attributes can be a part of predefined set with some of the attributes having no values or a default value filled in. *Id.* at 5:24-43. At step 30, the control device establishes a control session for the user which may contain specified time limits or other session features. *Id.* at 5:44-53.

81. The control device then establishes specific rules and conditions based on the attributes in the user profile at step 32. *Id.* at 5:54-58. These rules can be represented by an IP table and each rule can reference other rules or parameters usable by any number of applications or processes. *Id.* at 5:54-6:10. From then on, when the control device receives a network communication to or from a particular user, as seen in step 34, the control device determines which user specific rule to apply to said network communication which it extracts from the Internet Protocol (“IP”) packets associated with the network communication. *Id.* at 6:31-57.

82. In step 38, the control device can determine if the network communication exceeds the bandwidth limit that was imposed by a traffic control rule. *Id.* at 6:44-7:4. If so, the control device can queue or drop the network communication. If the bandwidth limit is not exceeded, then the traffic is delivered by the control device and the process repeats for any additional network communication. *Id.*

83. As the user profile’s traffic rules that are imposed on the user are modified in response to dynamic changes in the network, the control device can swap between dropping or delivering network communication for a user based on the network’s current needs.

B. CONSTRUCTION OF THE ’153 PATENT TERMS

I. “arbitrary number of attributes”

84. I understand that the parties have proposed the following constructions for this term:

Netskope	Fortinet
Plain and ordinary meaning	Indefinite

85. The term “arbitrary number of attributes” appears in claims 1, 10, and 15 of the ’153 Patent. The term appears in claim 1 as follows:

86. A method executing on a control device for allocating network bandwidth to users accessing a first network, comprising:

in response to a first user of the users connecting to said control device using a user device on a second network, obtaining a user bandwidth allocation profile for said first user based on user credentials, wherein said first user is a human user wherein said user bandwidth allocation profile is stored local or remote to said control device, wherein the user bandwidth allocation profile contains an **arbitrary number of attributes** specifying bandwidth limitations for the first user, and wherein said control device is located between said user device and said first network, said control device capable of dynamically altering bandwidth allocations among a set of users on said second network;

based on the **arbitrary number of attributes** in each user profile, establishing user specific rules and conditions that are bound to the first user during a control session based on the user device associated with the first user and the first credentials provided by the first user for the control session wherein the user specific rules include at least one network bandwidth limit for the first user;

obtaining information identifying said user device used by said first user on said second network to connect to said control device for accessing said first network;

associating said at least one traffic control rule with said first user based on said user credentials and considering said information identifying said user device used by said first user on said second network to connect to said control device for accessing said first network; and

dynamically updating the network bandwidth for said first user utilizing said at least one traffic control rule associated with said first user.

87. It is my belief that the term “arbitrary number of attributes” has a plain and ordinary meaning within the context of the ’153 Patent that would have been readily understood by one of ordinary skill in the art at the time of filing. The specification, claim language, and prosecution history collectively indicate that the term refers to a user profile that utilizes a variable number of attributes to determine user-specific rules and conditions that control bandwidth allocation. An

arbitrary number of attributes refers to a list of possible attributes wherein some attributes are not applicable to some users. As a result, the number of attributes used to determine bandwidth allocation may vary from user to user.

88. I understand that Fortinet argues “arbitrary number of attributes” is indefinite to a person of ordinary skill in the art. I disagree and believe that one of ordinary skill in the art would understand “arbitrary number of attributes” in light of the specification, claim language, and knowledge of the field would have understood the scope and meaning of this term with reasonable certainty.

89. The specification explains that each user profile may include an arbitrary number of attributes, and the control device is capable of specifying a bandwidth limitation for the corresponding user regardless of the number or completeness of those attributes. *Id.* at 2:48:58. The control device can assign traffic rules to each user profile even when attributes are incomplete or missing, by substituting default values for the missing data. *Id.* at 5:35-43. Because the number of missing attributes may vary from user to user, the number of attributes set to default values will also vary. Importantly, each user profile is assigned a traffic rule—regardless of how many attributes are defaulted. As a result, the number and type of attributes used to determine each user’s bandwidth may differ, enabling flexible and individualized bandwidth management.

90. For the reasons stated above, it is my opinion that a person of ordinary skill in the art would understand the plain meaning of “arbitrary number of attributes.” The specification of the ’153 Patent describes the process of the control device substituting default values for empty attributes and providing a traffic rule for every user profile, regardless of how many attributes are set to default values. As the traffic rules are still able to perform the communication relay and dynamically provision bandwidth to the users based on the traffic rules, the number of attributes is arbitrary. Thus, the plain language of the ’153 Patent provides sufficient detail to inform a person of ordinary skill in the art on the meaning of this term.

XI. THE '697 PATENT

A. OVERVIEW OF THE TECHNOLOGY

91. The '697 Patent addresses challenges in residential or business networks, where multiple devices connect to a service provider (SP) network through a single WAN (Wide Area Network) IP address via a router that conducts NAT (Network Address Translation). All devices connecting through the same NAT share the same IP address, as the NAT modifies the network address information when routing traffic from the private network via the NAT gateway to the public network and vice versa. When malware is detected at the network level, this framework makes determining which specific device is infected nearly impossible, preventing an efficient and targeted remediation of the infected device.

92. The '697 Patent introduces a systematic remediation technique that helps identify computers infected with malware. Specifically, the patent focuses on implementing these techniques on devices behind a NAT gateway by combining TCP session fingerprinting and UA (user-agent) analysis to accurately determine a device's operating system identifier (OS ID). By more precisely identifying an infected device, users can avoid unnecessary cleanup on unaffected devices and reduce the time spent to investigate anomalies on the network.

93. Fig. 1 illustrates an embodiment of the invention, detailing the operating system and malware remediation architecture. Multiple devices at user location 100 connect to a service provider network 110 via a an access device, such as a NAT 102:

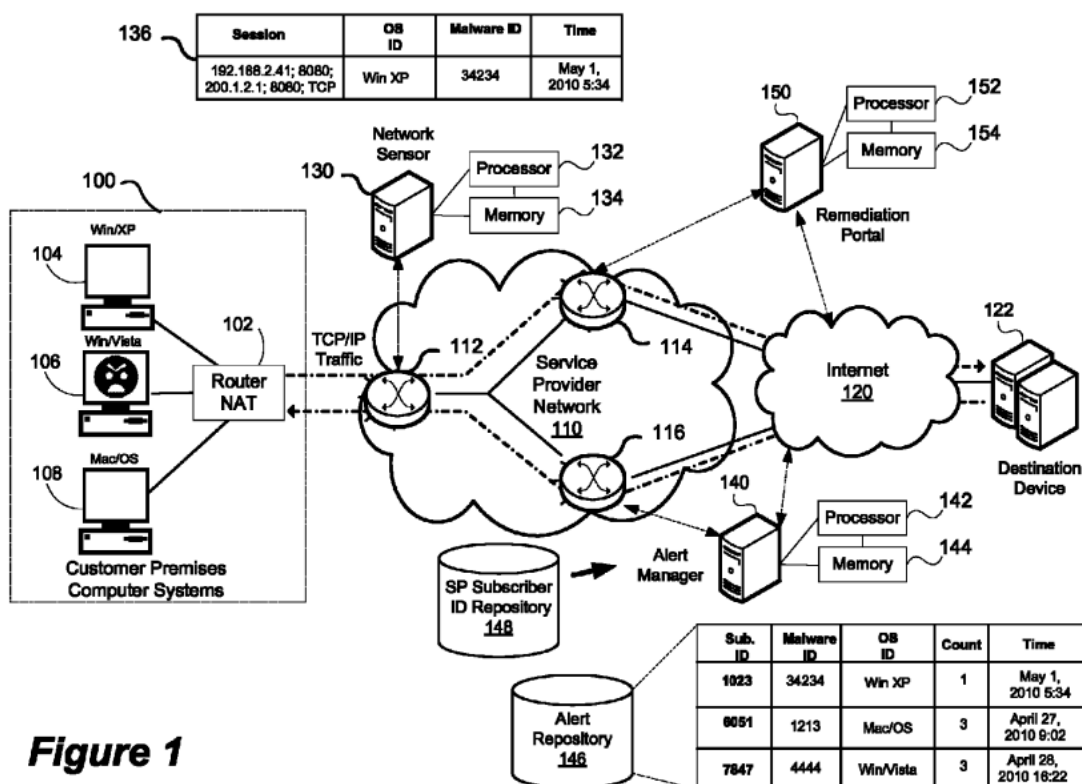
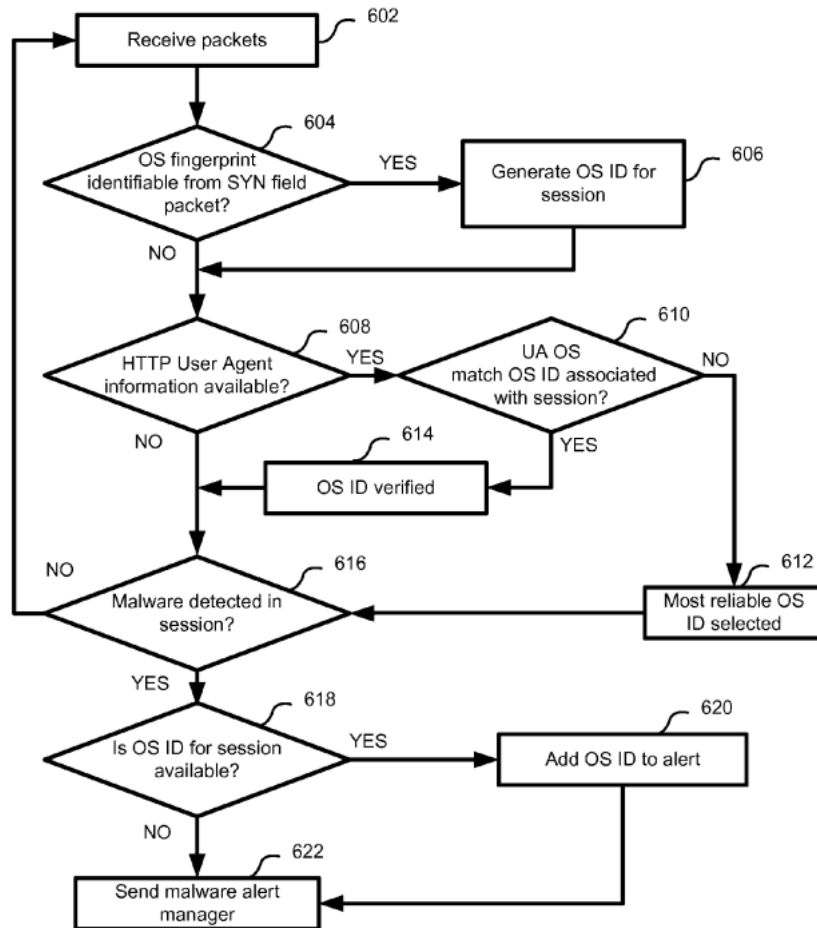


Figure 1

94. “The system of the present disclosure, or parts thereof as described further below, are deployed within the SP network in order to capture and analyze the network traffic.” Ex. 6 at 4:33-36. “Using the analyzed traffic, the system can determine if a subscriber’s computer 104 is infected with malware.” *Id.* at 4:36-37. “If the system determines that a subscriber’s computer 104 is infected, it can provide a notification to the subscriber that a computer running a specific operating system is infected and direct them to remedy the infection.” *Id.* at 3:37-41.

95. Specifically, Fig. 6 provides another embodiment detailing OS ID detection and remediation flow:

Figure 6

96. As shown in Fig. 6, once network packets are received from the service provider network, the packets undergo multiple analyses to determine if the OS ID is known. At step 604, if an OS fingerprint is identifiable from the SYN packet, an OS identifier is generated. However, if an OS fingerprint is not identified, the packets are sent to the application protocol analysis module, which assesses the HTTP User Agent information (if present) and information about the TCP session to compare the OS associated with the packet with the OS identifier from the SYN packet—in the event of a mismatch, the most reliable OS ID is selected.

97. The malware analysis module then performs another comparison, in which “each of the received packets to a detection rule, or signature of the detection rule, for a particular piece of malware.” Ex. 6 at 9:6-8. The system then generates an alert “identifying a network address associated with the access device, the malware ID and the OS ID associated with TCP session that generated the alert.” *Id.* at 2:40-43. Overall, these techniques enable network managers to have detailed information about the malware detected on the network, which increases efficiency and effectiveness of remediation.

B. CONSTRUCTION OF THE '697 PATENT TERMS

I. “a more reliable OS ID”

98. I understand that the parties have proposed the following constructions for this term:

Netskope	Fortinet
Plain and ordinary meaning	Indefinite

99. The term “a more reliable OS ID” appears in claims 9 and 21 of the '697 Patent. The term appears in claim 9 as follows:

9. The method of claim 2 further comprising verifying the first OS ID against the second OS ID, wherein if the second OS ID does not match the first OS ID the session is identified by the OS ID identified as a ***more reliable OS ID***.

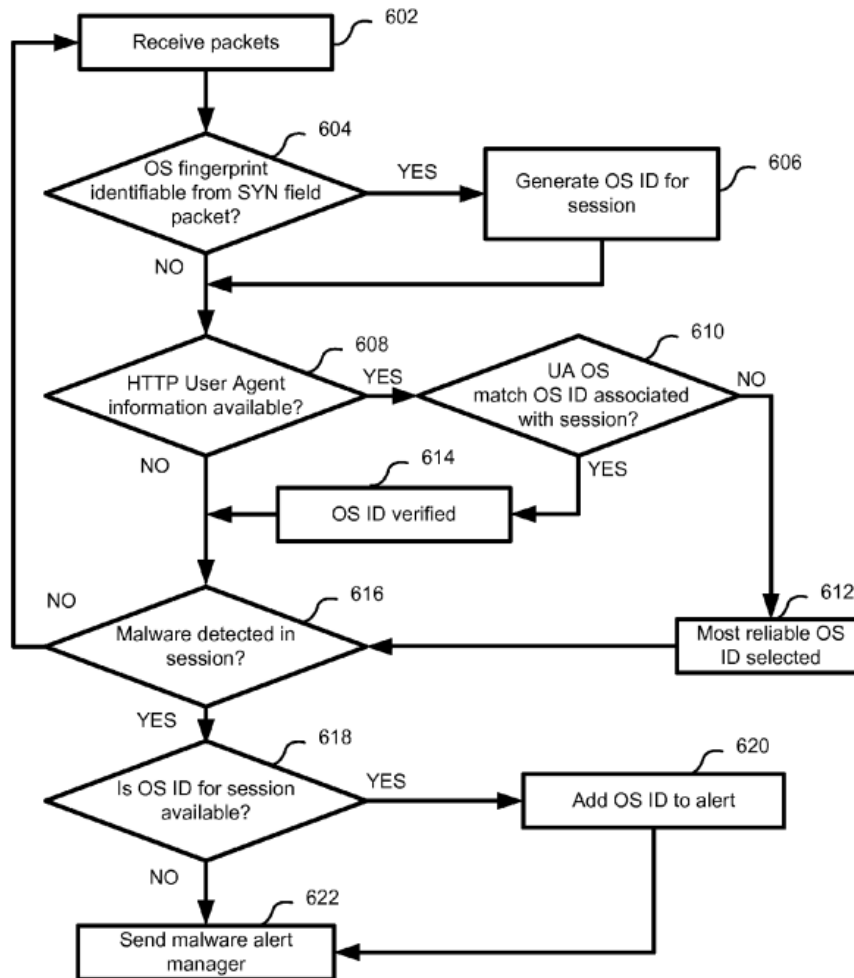
100. I understand that Fortinet argues that the term “a more reliable OS ID” is indefinite. I disagree. It is my opinion that a person of ordinary skill in the art would understand that the term “a more reliable OS ID” is not indefinite in light of the intrinsic evidence, including the claims and specification.

101. Specifically, the specification describes that the system receives network packets from the service provider network, and then analyzes these packets to determine an OS fingerprint—performed by comparing the SYN packet for the session against known SYN

fingerprints. Ex. 6 at 8:51-55. From there, the system determines whether an OS ID was identified, and if not, the packets are sent to an application protocol analysis module. *Id.* at 8:8:55-58.

102. The protocol analysis module performs a malware analysis if the HTTP UA information is not present in the packet, but if it is, then the module compares the OS specified in the UA field is compared with the OS identifier from the SYN packet. *Id.* at 8:59-62. If the OS identifiers match, then the OS identifier is verified. *Id.* at 8:62-63. However, in cases where there is a mismatch, the module chooses the most reliable one. *Id.* at 8:63-64. Once the OS identifier is verified, the UA information may be used to refine the OS identified by the SYN parameters. *Id.* at 8:65-66.

103. Figure 6 demonstrates this process. For example, Figure 6 shows the comparison step 610, wherein if the UA OS ID does not match the OS ID associated with the session, then the most reliable OS ID is selected.



Ex. 6, Fig. 6 (annotated).

104. In the context of the specification and figures, “a more reliable OS ID” refers to the OS ID that has achieved greater accuracy to trusted parameters (e.g., known SYN fingerprints). By selecting the more reliable OS ID, the OS ID is less prone to misclassification, especially in environments behind NATs where multiple devices share an address. Accordingly, “more reliable” thus means that the OS ID is validated through independent or cross-correlation of data paths, which enables the system to release more trustworthy alerts and guide remediation effectively.

105. For the reasons stated above, it is my opinion that a person of ordinary skill in the art would understand the plain meaning of “a more reliable OS ID.” The specification of the ’697

Patent describes this analysis as merely performing a comparison of OS IDs—if they match, then the OS ID is verified. If they do not match, the module selects the OS ID which is more reliable. This description in the specification is also supported in Figure 6. Thus, the plain language of the '697 Patent provides sufficient detail to inform a person of ordinary skill in the art on the meaning of this term.

I declare under penalty of perjury under the laws of the United States of America that the foregoing is true and correct.

Dated: 10/14/2025

A handwritten signature in black ink, appearing to read 'Aaron D. Striegel', is positioned above a horizontal line.

Aaron D. Striegel