

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

CISCO SYSTEMS, INC.,
Petitioner,

v.

LIONRA TECHNOLOGIES LTD.,
Patent Owner.

IPR2024-00734
Patent 7,623,518 B2

Before WILLIAM V. SAINDON, PATRICK R. SCANLON, and
STEVEN M. AMUNDSON, *Administrative Patent Judges*.

AMUNDSON, *Administrative Patent Judge*.

DECISION
Granting Institution of *Inter Partes* Review
35 U.S.C. § 314

I. INTRODUCTION

Cisco Systems, Inc. (“Petitioner”) filed a Petition requesting an *inter partes* review of claims 1, 3, 15, 17, and 19–21 in U.S. Patent No. 7,623,518 B2 (Exhibit 1001, “the ’518 patent”) under 35 U.S.C. §§ 311–319. Paper 2 (“Pet.”). Lionra Technologies Ltd. (“Patent Owner”) did not file a Preliminary Response.

Under 37 C.F.R. § 42.4(a), we have authority to determine whether to institute an *inter partes* review. We may institute an *inter partes* review only if “the information presented in the petition filed under section 311 and any response filed under section 313 shows that there is a reasonable likelihood that the petitioner would prevail with respect to at least 1 of the claims challenged in the petition.” 35 U.S.C. § 314(a) (2018). The “reasonable likelihood” standard is “a higher standard than mere notice pleading” but “lower than the ‘preponderance’ standard to prevail in a final written decision.” *Hulu, LLC v. Sound View Innovations, LLC*, IPR2018-01039, Paper 29 at 13 (PTAB Dec. 20, 2019) (precedential).

Based on the current record and for the reasons explained below, Petitioner has shown that there is a reasonable likelihood that it would prevail with respect to at least one of the challenged claims. Thus, we institute an *inter partes* review of claims 1, 3, 15, 17, and 19–21 in the ’518 patent on all challenges included in the Petition.

II. BACKGROUND

A. *Real Parties in Interest*

Petitioner identifies itself as the sole real party in interest. Pet. 88. Patent Owner identifies itself as the sole real party in interest. Paper 5, 2. The parties do not raise any issue about real parties in interest.

B. Related Matters

Petitioner and Patent Owner identify the following civil actions as related matters involving the '518 patent:

- *Lionra Technologies Ltd. v. Fortinet, Inc.*, No. 2:23-cv-00206 (E.D. Tex. filed May 9, 2023);
- *Lionra Technologies Ltd. v. Cisco Systems, Inc.*, No. 2:23-cv-00207 (E.D. Tex. filed May 9, 2023) (the “Texas case”);
- *Lionra Technologies Ltd. v. Palo Alto Networks, Inc.*, No. 2:23-cv-00214 (E.D. Tex. filed May 11, 2023);
- *Lionra Technologies Ltd. v. CrowdStrike Holdings, Inc. et al.*, No. 1:23-cv-00928 (W.D. Tex. filed Aug. 7, 2023); and
- *Lionra Technologies Ltd. v. VMware, Inc.*, No. 1:23-cv-00929 (W.D. Tex. filed Aug. 7, 2023).

Pet. 88; Paper 5, 2.

C. The '518 Patent (Exhibit 1001)

The '518 patent, titled “Dynamic Access Control Lists,” issued on November 24, 2009, from an application filed on April 8, 2004. Ex. 1001, codes (22), (45), (54). The patent explains that in “an Ethernet local area network (LAN), computers or hosts are attached to the network and each host is uniquely identified by a physical address which is designated a media access control (MAC) address in an Ethernet network.” *Id.* at 1:5–8. The patent also explains that domain name system (DNS) servers “contained in the Internet convert the domain name into a corresponding IP address which allows a host to communicate with a desired host corresponding to that IP address.” *Id.* at 2:61–64.

The '518 patent identifies “a need for access control lists that allows [sic] access of hosts in a computer network such as an Ethernet network to be restricted or controlled based upon user names, MAC addresses, domain names, and DNS names instead of merely IP address information.”

Ex. 1001, 3:12–16. The patent endeavors to address that need by providing a method for controlling a user’s access to a network that includes:

- (1) storing in a network switch “an enhanced access control list containing data related to at least one of user names, DNS names, domain names, and physical addresses”; and
- (2) generating a dynamic access control from “the enhanced access control list, with the dynamic access control list containing a plurality of IP addresses that restrict access of the user to the network.”

Id. at 3:20–30, code (57).

The '518 patent’s Figure 1 (reproduced below) depicts an Ethernet network with an Ethernet switch:

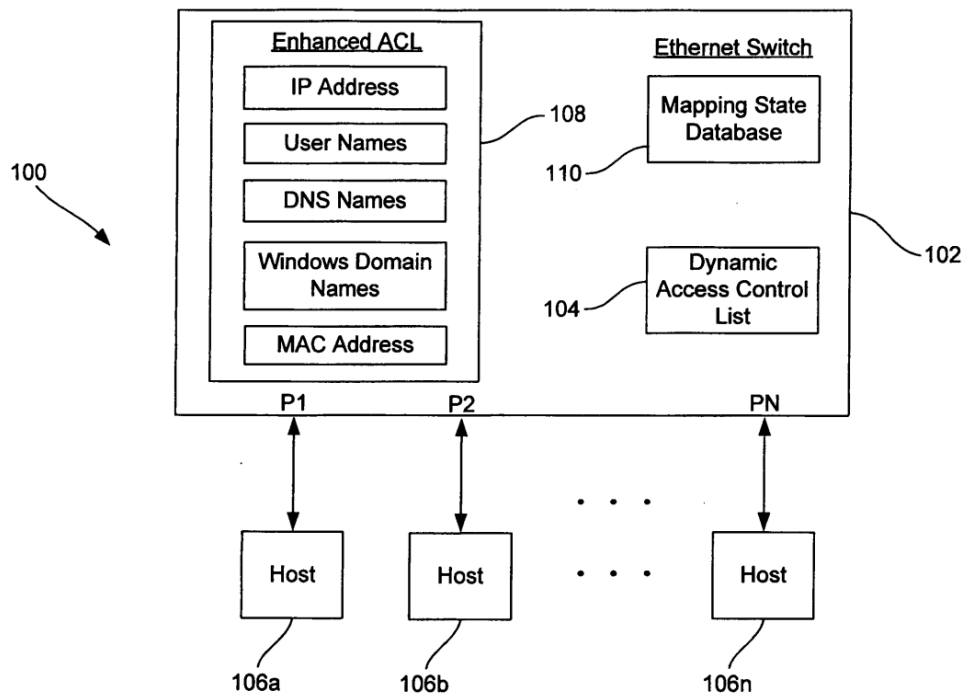


FIG. 1

Figure 1 illustrates Ethernet network 100 with Ethernet switch 102 connected through ports P1–PN to hosts 106a–106n. Ex. 1001, 3:49–53, 4:44–46, Fig. 1.

As Figure 1 shows, Ethernet switch 102 includes (1) “dynamic access control list 104 containing IP addresses currently being utilized by various users of the network 100,” (2) “enhanced access control list 108 containing IP address, user name, DNS name, Windows domain name, and MAC address information for the users and hosts 106a–n of the network 100,” and (3) mapping state database 110 storing records that associate “the user name with the MAC address of the host 106a–n the user is utilizing.” Ex. 1001, 3:53–55, 3:57–62, 4:61–5:3, Fig. 1.

Ethernet switch 102 “detects all dynamic host configuration protocol (DHCP) packets sent by any of the hosts” to a DHCP server. Ex. 1001, 5:4–6. A DHCP packet includes the source MAC address of the host sending the packet to the DHCP server and, therefore, identifies the host requesting the IP address from the DHCP server. *Id.* at 5:9–12. Ethernet switch 102 “then monitors the return DHCP packets from the DHCP server sent in response to the DHCP packets, and from these return DHCP packets determines the IP address assigned to” the host requesting the IP address from the DHCP server. *Id.* at 5:12–16. “At this point, the switch knows the IP address assigned to a particular MAC address.” *Id.* at 5:16–17.

The '518 patent's Figure 2 (reproduced below) is a functional flow diagram depicting the process executed by Ethernet switch 102 in generating a dynamic access control list:

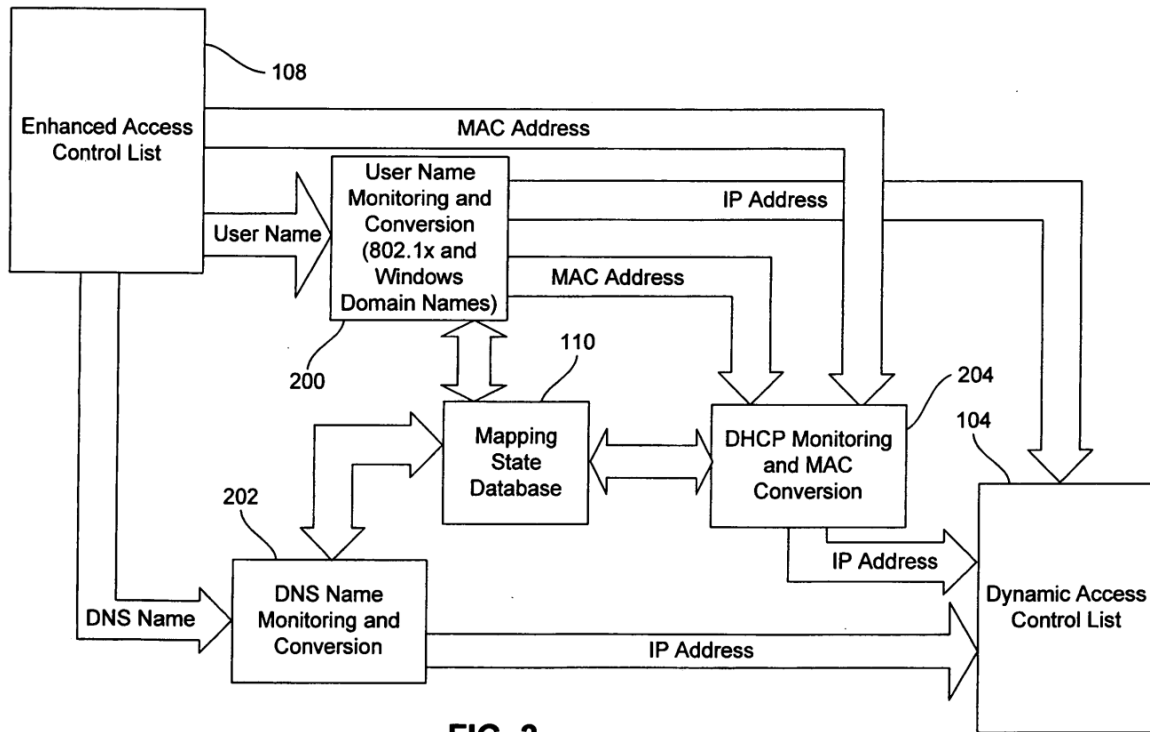


FIG. 2

Figure 2 illustrates dynamic access control list 104, enhanced access control list 108, and mapping state database 110 along with three subprocesses as follows:

- (1) “a user name monitoring and conversion subprocess 200”;
- (2) “a DNS name monitoring and conversion subprocess 202”; and
- (3) “a DHCP monitoring and MAC conversion subprocess 204.”

Ex. 1001, 5:56–66, Fig. 2. These “subprocesses operate in combination to apply the rules contained in the enhanced access control list 108 to generate”

dynamic access control list 104 and mapping state database 110. *Id.* at 5:67–6:3.

Subprocess 200 “converts user names into either MAC or IP addresses depending upon the manner in which a user is attempting to log into the network 100.” Ex. 1001, 6:4–7. Subprocess 202 “detects all Ethernet packets being communicated through the switch 102 (FIG. 1) having a source IP address which does not yet exist in the mapping state database 110.” *Id.* at 7:10–13. When subprocess 202 “detects an unknown source IP address, the subprocess generates a reverse DNS lookup query containing the source IP address, and sends this query to a DNS server.” *Id.* at 7:24–27. Subprocess 204 “monitors DHCP return packets from a DHCP server, which corresponds to one of the hosts 106 (FIG. 1), and from these return packets determines the IP address assigned by the DHCP server to a given MAC address.” *Id.* at 7:54–58.

The '518 patent's Figure 3 (reproduced below) depicts Ethernet switch 102 in more detail:

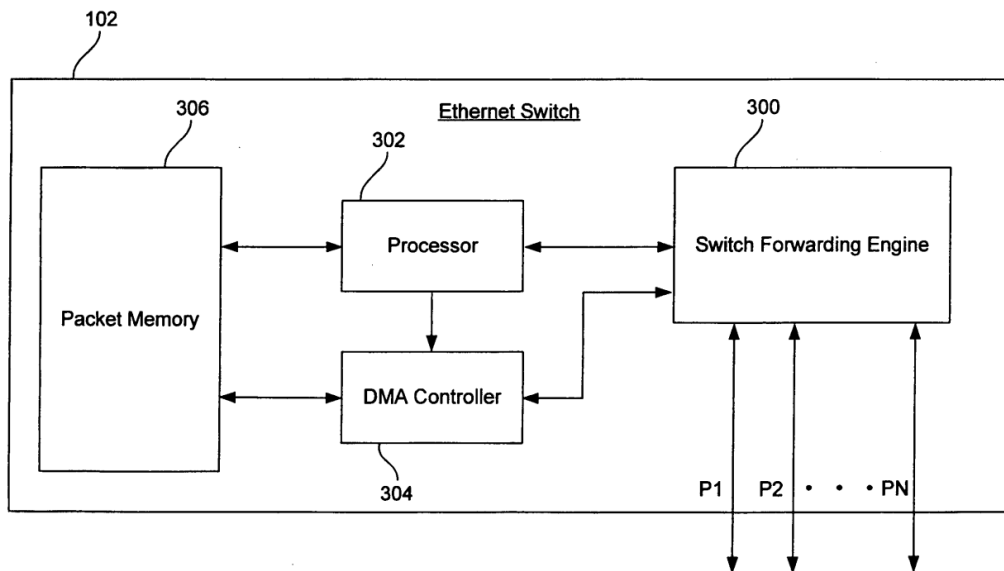


FIG. 3

Figure 3 illustrates Ethernet switch 102 including the following components:

- (1) switch forwarding engine 300 that “examines Ethernet packets received on one of a plurality of ports P1–PN from a host (not shown) coupled to each port and forwards each received Ethernet packet to the proper host through the corresponding port”;
- (2) processor 302 that “programs the switch forwarding engine 300 to execute a desired set of rules to provide selected Ethernet packets received by the forwarding engine to the processor”; and
- (3) direct memory access (DMA) controller 304 that
 - (i) “transfers the packets selected by the switch forwarding engine 300 to a packet memory 306” and
 - (ii) reads “packets from the packet memory 306” so that “the switch forwarding engine 300” may forward “the packets to the required destination host.”

Ex. 1001, 9:9–19, 9:28–30, 9:51–55, Fig. 3.

D. The Challenged Claims

Petitioner challenges the following claims in the ’518 patent:

- independent claim 1 for a “method of developing an access control list”;
- claim 3 that depends directly from claim 1;
- independent claim 15 for a “network switching circuit”;
- claim 17 that depends directly from claim 15;
- independent claim 19 for a “computer network”; and
- claims 20 and 21 that depend directly from claim 19.

Pet. 9–85.

Claims 1 and 15 exemplify the challenged claims and read as follows (with formatting added for clarity):

1. A method of developing an access control list, comprising:

developing an enhanced access control list including data related to at least one of user names, DNS names, Windows domain names, and physical addresses;

converting, in a network switch, at least one of:

user names into corresponding IP and physical addresses according to data in the enhanced access control list, wherein converting user names into corresponding IP and physical addresses according to data in the enhanced access control list includes: detecting login packets being communicated over the network; determining a MAC address from the login packets; detecting server message block login packets being communicated over the network; determining an IP address from the server message block login packets; and developing records in the access control list using the obtained IP address for the respective user name;

DNS names into corresponding IP addresses according to data in the enhanced access control list; and

physical addresses into IP addresses according to data in the enhanced access control list; and

developing, in the network switch, the access control list from each of the operations of converting.

15. A network switching circuit, comprising:

a forwarding circuit

operable to detect specific received packets and to provide the specific packets on a processor port, and

further operable to receive packets on one of a plurality of ports including the processor port and to forward each received packet to a port corresponding to a destination address contained in the packet subject to access restrictions contained in a dynamic access control list;

a memory circuit coupled to the forwarding circuit, the memory circuit

operable to store packets and

operable to store an enhanced access control list and a dynamic access control list; and

a processor coupled to the forwarding circuit and to the memory circuit, the processor

operable to define the specific packets detected by the forwarding circuit and

operable to process the specific packets stored in the memory circuit using the enhanced access control list to generate the dynamic access control list and store the dynamic access control list in the memory circuit, and

further operable to provide the specific packets to the processor port of the forwarding circuit after processing the packets.

Ex. 1001, 10:12–37, 11:57–12:17.

E. The Asserted References

For its challenges, Petitioner relies on the following references:

Name	Reference	Exhibit
Gai	US 6,651,096 B1, issued November 18, 2003 (based on an application filed April 20, 1999)	1006
Kwan	US 7,523,485 B1, issued April 21, 2009 (based on an application filed July 31, 2003)	1007
Georgiou	US 7,003,597 B2, issued February 21, 2006 (based on an application filed July 9, 2003)	1008
Yip	US 8,751,647 B1, issued June 10, 2014 (based on an application filed June 30, 2001)	1013

Pet. 9–85.

Petitioner asserts that Gai, Kwan, Georgiou, and Yip qualify as prior art “under Pre-AIA 35 U.S.C. §§ 102(a) and 102(e).” Pet. 9–10; *see* 35 U.S.C. § 102(a), (e) (2006).¹

F. The Asserted Challenges to Patentability

Petitioner asserts the following challenges to patentability:

Claims Challenged	35 U.S.C. §	References/Basis
1, 3	103(a)	Gai, Yip, Kwan
15, 17, 19–21	103(a)	Gai, Yip, Kwan, Georgiou

Pet. 9–85.

G. Testimonial Evidence

To support its challenges, Petitioner relies on the declaration of Mark E. Crovella, Ph.D. (Exhibit 1003). Dr. Crovella states that he has been “asked to provide my opinions regarding whether the subject matter of claims 1, 3, 15, 17, and 19–21” in the ’518 patent “would have been obvious to a person having ordinary skill in the art” at “the time of the alleged invention, in light of the prior art.” Ex. 1003 ¶ 3. Dr. Crovella also states that he (1) “earned an undergraduate degree in Biology from Cornell University” in 1982, (2) “received a master’s degree in Computer Science from the University of Buffalo in 1989,” and (3) “received a Ph.D. in Computer Science from the University of Rochester in 1994.” *Id.* ¶ 8.

At this stage of the proceeding, Patent Owner does not introduce testimonial evidence concerning the patentability issues.

¹ The Leahy-Smith America Invents Act (“AIA”), Pub. L. No. 112-29, 125 Stat. 284 (2011), amended 35 U.S.C. § 102 and § 103 effective March 16, 2013. Because the effective filing date of the challenged claims predates the AIA’s amendments to § 102 and § 103, this decision refers to the pre-AIA versions of § 102 and § 103.

III. PATENTABILITY ANALYSIS

A. Legal Principles: Obviousness

A patent may not be obtained “if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains.” 35 U.S.C. § 103(a) (2006). An obviousness analysis involves underlying factual inquiries including (1) the scope and content of the prior art; (2) differences between the claimed invention and the prior art; (3) the level of ordinary skill in the art; and (4) where in evidence, objective indicia of nonobviousness, such as commercial success, long-felt but unsolved needs, and failure of others. *Graham v. John Deere Co.*, 383 U.S. 1, 17–18, 35–36 (1966); *Apple Inc. v. Samsung Elecs. Co.*, 839 F.3d 1034, 1047–48 (Fed. Cir. 2016) (en banc). When evaluating a combination of references, an obviousness analysis should address “whether there was an apparent reason to combine the known elements in the fashion claimed by the patent at issue.” *KSR Int’l Co. v. Teleflex Inc.*, 550 U.S. 398, 418 (2007).

We analyze the obviousness issues according to these principles.

B. Level of Ordinary Skill in the Art

Factors pertinent to determining the level of ordinary skill in the art include (1) the educational level of the inventor; (2) the type of problems encountered in the art; (3) prior-art solutions to those problems; (4) the rapidity with which innovations are made; (5) the sophistication of the technology; and (6) the educational level of workers active in the field. *Envtl. Designs, Ltd. v. Union Oil Co.*, 713 F.2d 693, 696–97 (Fed. Cir. 1983). Not all factors may exist in every case, and one or more of these or

other factors may predominate in a particular case. *Id.* These factors are not exhaustive, but merely a guide to determining the level of ordinary skill in the art. *Daiichi Sankyo Co. v. Apotex, Inc.*, 501 F.3d 1254, 1256 (Fed. Cir. 2007). Moreover, the prior art itself may reflect an appropriate skill level. *Okajima v. Bourdeau*, 261 F.3d 1350, 1355 (Fed. Cir. 2001).

Petitioner asserts that a person of ordinary skill in the art at the time of the alleged invention “would have been someone”:

- (1) “knowledgeable with the computer networking techniques available at the time”; and
- (2) “who had a bachelor’s degree in computer science, computer engineering, electrical engineering, or an equivalent, and two years of experience working with computer networks, including network security.”

Pet. 7. Petitioner also asserts that “[l]ack of work experience can be remedied by additional education, and vice versa.” *Id.* Dr. Crovella’s testimony supports Petitioner’s assertions. *See* Ex. 1003 ¶¶ 15–16.

Based on the current record and for purposes of this decision, we accept Petitioner’s description of an ordinarily skilled artisan because it comports with the technology and claims of the ’518 patent as well as the asserted references. *See* Ex. 1003 ¶ 16.

C. Claim Construction

We construe claim terms “using the same claim construction standard” that district courts use to construe claim terms in civil actions under 35 U.S.C. § 282(b). *See* 37 C.F.R. § 42.100(b) (2024). Under that standard, claim terms “are given their ordinary and customary meaning, which is the meaning the term would have to a person of ordinary skill in the art at the time of the invention.” *Power Integrations, Inc. v. Fairchild*

Semiconductor Int'l, Inc., 904 F.3d 965, 971 (Fed. Cir. 2018) (citing *Phillips v. AWH Corp.*, 415 F.3d 1303, 1312–13 (Fed. Cir. 2005) (en banc)). The meaning of claim terms may be determined by “look[ing] principally to the intrinsic evidence of record, examining the claim language itself, the written description, and the prosecution history, if in evidence.” *DePuy Spine, Inc. v. Medtronic Sofamor Danek, Inc.*, 469 F.3d 1005, 1014 (Fed. Cir. 2006) (citing *Phillips*, 415 F.3d at 1312–17).

For the phrase “at least one of” claim 1’s first “developing” step and claim 1’s “converting” step, Petitioner contends that in the Texas case Patent Owner interpreted that phrase as requiring “only one of the listed items,” e.g., “only one” of “user names, DNS names, Windows domain names, and physical addresses” as recited in the first “developing” step. Pet. 8; *see* Ex. 1001, 10:14–16. Here, Petitioner “applies Patent Owner’s apparent district court interpretation of the claim term ‘at least one of’” in claim 1. Pet. 8; *see id.* at 25, 33–34, 39. Further, because the phrase “each of” in claim 1’s second “developing” step “relates back to the ‘at least one of’ converting operations” in claim 1’s “converting” step, Petitioner “applies Patent Owner’s apparent district court interpretation to mean developing from the at least one of the listed items of converting.” *Id.* at 8 (emphasis omitted); *see id.* at 39.

Petitioner does not propose an explicit construction for any other claim term. *See, e.g.*, Pet. 7–85.

“[O]nly those terms need be construed that are in controversy, and only to the extent necessary to resolve the controversy.” *Vivid Techs., Inc. v. Am. Sci. & Eng’g, Inc.*, 200 F.3d 795, 803 (Fed. Cir. 1999); *see Nidec Motor Corp. v. Zhongshan Broad Ocean Motor Co.*, 868 F.3d 1013, 1017

(Fed. Cir. 2017). Based on the current record, we determine that no claim term requires an explicit construction to decide whether Petitioner satisfies the “reasonable likelihood” standard for instituting trial.

D. Alleged Obviousness over Gai, Yip, and Kwan: Claims 1 and 3

Petitioner contends that claims 1 and 3 are unpatentable under § 103(a) because the claims would have been obvious over Gai, Yip, and Kwan. *See* Pet. 10–42. Below, we provide overviews of Gai, Yip, and Kwan. Then, we consider obviousness issues. For the reasons explained below, Petitioner establishes sufficiently for purposes of institution that the combined disclosures in Gai, Yip, and Kwan teach the subject matter in claims 1 and 3.

1. OVERVIEW OF GAI (EXHIBIT 1006)

Gai is a U.S. patent titled “Method and Apparatus for Organizing, Storing and Evaluating Access Control Lists,” filed on April 20, 1999, and issued on November 18, 2003. Ex. 1006, codes (12), (22), (45), (54). Gai states that the invention “relates generally to computer networks, and more specifically, to a method and apparatus for manipulating access control lists in order to optimize their evaluation by computer network devices.” *Id.* at 1:15–18; *see id.* at code (57).

Gai explains that certain devices on a network employ access control lists (ACLs) to “provide security” by determining whether “messages should be forwarded or filtered (i.e., dropped) based on certain predefined criteria,” such as “IP source address, IP destination address, or upper-layer application based on TCP/UDP port numbers.” Ex. 1006, 2:46–51, 3:17. “Access control lists may be established for both inbound and outbound traffic and

are most commonly configured at border devices (i.e., gateways or firewalls) to provide security to the network.” *Id.* at 2:53–57.

Gai identifies a need to “optimize the creation and evaluation of multiple access control lists so as to maintain, if not improve, packet processing speeds.” Ex. 1006, 3:34–36. Gai endeavors to address that need by providing “a method and apparatus for efficiently organizing, storing and evaluating access control lists” for “use by an intermediate network device of a computer network.” *Id.* at 3:55–57, code (57).

More specifically, the “intermediate network device includes an ACL converter which, in turn, includes a boolean transformation engine that is operatively coupled to a boolean manipulation engine.” Ex. 1006, code (57). The “boolean transformation engine is configured to access the ACLs in [a] first format and to translate them into a first boolean representation, such as binary decision diagram (BDD) format.” *Id.* The “boolean manipulation engine is configured to perform one or more operations on the ACLs specified for a given interface, including a merge operation, so as to generate a single, unified ACL for the given interface.” *Id.*

With “a single, unified ACL defined per interface per direction” and stored in a content addressable memory (CAM), the “intermediate network device is able to rapidly evaluate network messages.” Ex. 1006, 4:26–29. In particular, “upon receipt of a packet at a first interface, a forwarding entity at the intermediate network device tests the packet against the single, unified ACL stored in the corresponding portion of the CAM.” *Id.* at 4:29–32. “When a match is obtained, the corresponding decision is returned to the forwarding entity, which then takes the appropriate action (e.g., forward,

discard, log and forward, transfer to CPU for additional processing, etc.).”
Id. at 4:32–36.

Gai’s Figure 3 (reproduced below) depicts a computer network:

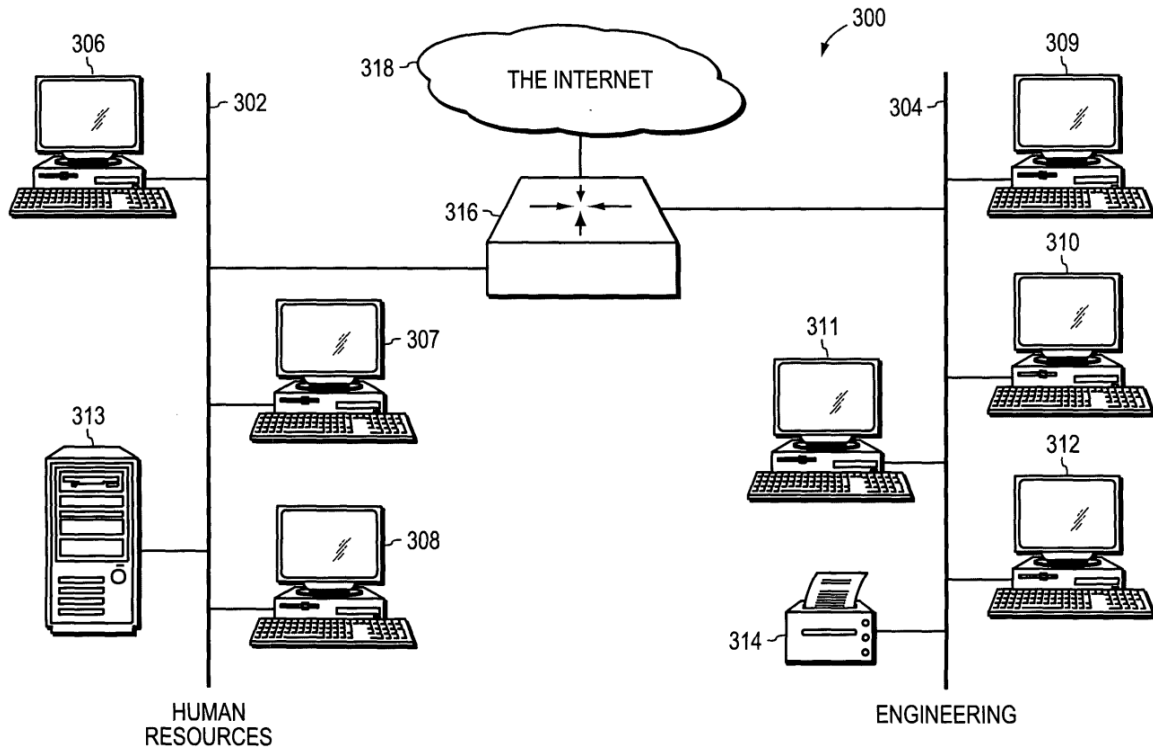


FIG. 3

Figure 3 illustrates computer network 300 including LAN 302, LAN 304, end stations 306–312, server 313, server 314, and intermediate network device 316 connected via the Internet 318. Ex. 1006, 5:3–14, Fig. 3.

Network device 316 may include “transmitting and receiving circuitry and components” and “may also implement network services such as route processing, path determination and path switching functions.” *Id.* at 5:27–32.

Gai's Figure 4 (reproduced below) depicts network device 316 in more detail:

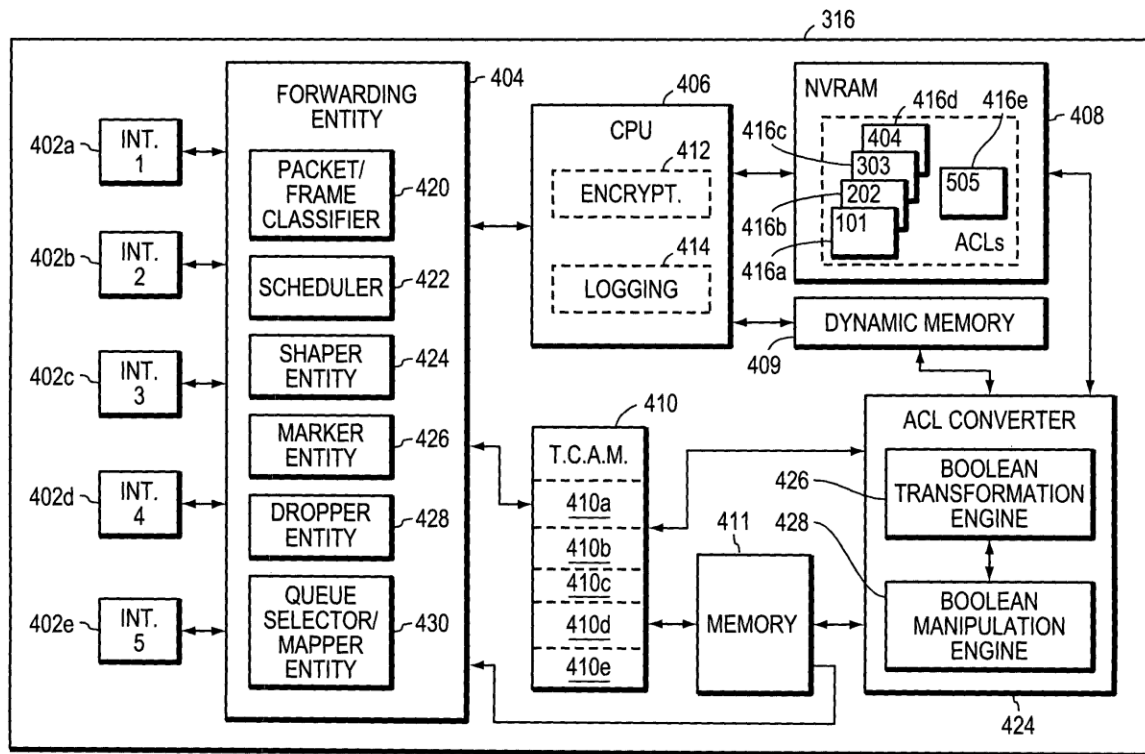


FIG. 4

Figure 4 illustrates network device 316 including the following components:

- a plurality of interfaces 402a–402e “associated with one or more physical ports (not shown)”;
- forwarding entity 404;
- central processing unit (CPU) 406;
- non-volatile random access memory (NVRAM) 408, e.g., for storing text-based ACLs 416a–416e;
- dynamic memory 409;
- content addressable memory 410, preferably “a ternary content addressable memory (TCAM),” apportioned into segments 410a–410e where each of segments 410a–410e corresponds to a particular interface 402a–402e;

- memory 411 coupled to TCAM 410, preferably including “a plurality of spaces or locations, such that each entry or row of the TCAM 410 has a corresponding location or space in memory 411”; and
- ACL converter 424 with boolean transformation engine 426 and boolean manipulation engine 428 “arranged in communicating relationship by communications means.”

Ex. 1006, 5:60–6:12, 6:36–51, Fig. 4.

ACL converter 424 “transforms the text-based ACLs” into corresponding BDD representations and “then merges all of the ACLs assigned to each interface into a single, unified ACL.” Ex. 1006, 9:25–29. ACL converter 424 “stores these single, unified ACLs in the TCAM 410 for subsequent use by the forwarding entity 404.” *Id.* at 9:29–31. “The final or merged actions to be implemented by a given forwarding entity may include permit, deny, forward to CPU for additional processing (e.g., encryption, logging, etc.), forward to an alternative forwarding entity, etc.” *Id.* at 13:41–45.

“ACL converter 424, including boolean transformation engine 426 and boolean manipulation engine 428, may run on or otherwise be executed by CPU 406 or some other processing element (not shown).” Ex. 1006, 6:52–55.

2. OVERVIEW OF YIP (EXHIBIT 1013)

Yip is a U.S. patent titled “Method and Apparatus for Network Login Authorization,” filed on June 30, 2001, and issued on June 10, 2014. Ex. 1013, codes (12), (22), (45), (54). Yip states that the invention “relates to the field of network security.” *Id.* at 1:6–7; *see id.* at code (57).

Yip explains that previous “network security techniques have been implemented at a connection point (i.e. at the LAN switch level) to prevent intruder and hacker attacks.” Ex. 1013, 2:33–35. For instance, techniques “have been designed and implemented in LAN switches to prohibit cyber-attacks, such as access control lists (ACLs).” *Id.* at 2:35–38. Yip also explains that “access lists were developed to combat cyber-attacks on the LAN switch by providing an administrator-controlled list of Internet Protocol (IP) addresses or media access control (MAC) addresses that were authorized to access the network.” *Id.* at 2:43–47.

Yip endeavors to improve network security by providing a method “to control the admission of a user to a network by preventing a port through which the user connects to the network from forwarding data packets until the user is authorized.” Ex. 1013, 3:39–43, code (57). According to the method, a “network login controller operates in conjunction with”:

- (1) “a user interface to receive a user identification data from the port user”; and
- (2) “an authorization server to authenticate the user by sending a user authentication request containing the user identification data to the authentication server.”

Id. at 3:43–48, code (57).

Based on “the user authentication response from the authentication server,” the “network login controller grants or denies permission to the user to access the network.” Ex. 1013, 3:48–51, code (57). “If permission is granted, then the network login controller unblocks the port through which the user is connected to place it in packet-forwarding mode.” *Id.* at 3:51–54, code (57). “If permission is denied, then the port remains in packet non-forwarding mode (i.e. it remains blocked).” *Id.* at 3:54–56, code (57).

3. OVERVIEW OF KWAN (EXHIBIT 1007)

Kwan is a U.S. patent titled “System and Method for Source IP Anti-Spoofing Security,” filed on July 31, 2003, and issued on April 21, 2009. Ex. 1007, codes (12), (22), (45), (54). Kwan states that the invention relates to “a method of providing for enhanced security on a computer network to reduce the risk created by the spoofing of IP addresses.” *Id.* at 1:15–17; *see id.* at code (57). According to Kwan, “source IP address spoofing attacks are widely known, and include attacks such as distributed denial of service attacks (DDoS), Worm attacks, and Man In the Middle attacks.” *Id.* at 1:22–26.

Kwan explains that “MAC addresses are used to identify the devices” connected to a network because “a MAC address is a unique address” corresponding to “a device connected to a network.” Ex. 1007, 1:38–41. A data packet transmitted over the network includes “the MAC address for the device to which the data packet is to be sent (the destination MAC address) and the MAC address for the host which is sending the packet (the source MAC address).” *Id.* at 1:55–61; *see id.* at 3:63–66.

Additionally, each host device, such as an end-user computer or a server, “will be assigned an IP address” used “to determine routing for data packets” transmitted over the network. Ex. 1007, 1:61–66; *see id.* at 3:66–67. “The IP address is often assigned using the Dynamic Host Configuration Protocol (DHCP).” *Id.* at 1:66–2:1. A data packet includes “a source IP address and a destination IP address” as well as the source MAC address and the destination MAC address. *Id.* at 2:2–7.

Kwan explains that source IP address spoofing “occurs when an attacker host uses a source IP address, which does not correspond, or is not

assigned, to its MAC address, in a transmitted data packet.” Ex. 1007, 2:8–10. To thwart source IP address spoofing, Kwan provides a system and method for (1) “analyzing MAC addresses and source IP addresses at the datalink (layer 2) level” and (2) using “the information derived from such analysis to block access through a port where a host device is using a false, or spoofed, source IP address in transmitted data packets.” *Id.* at code (57).

Kwan’s Figure 2 (reproduced below) depicts a network device providing layer 2 switching functions:

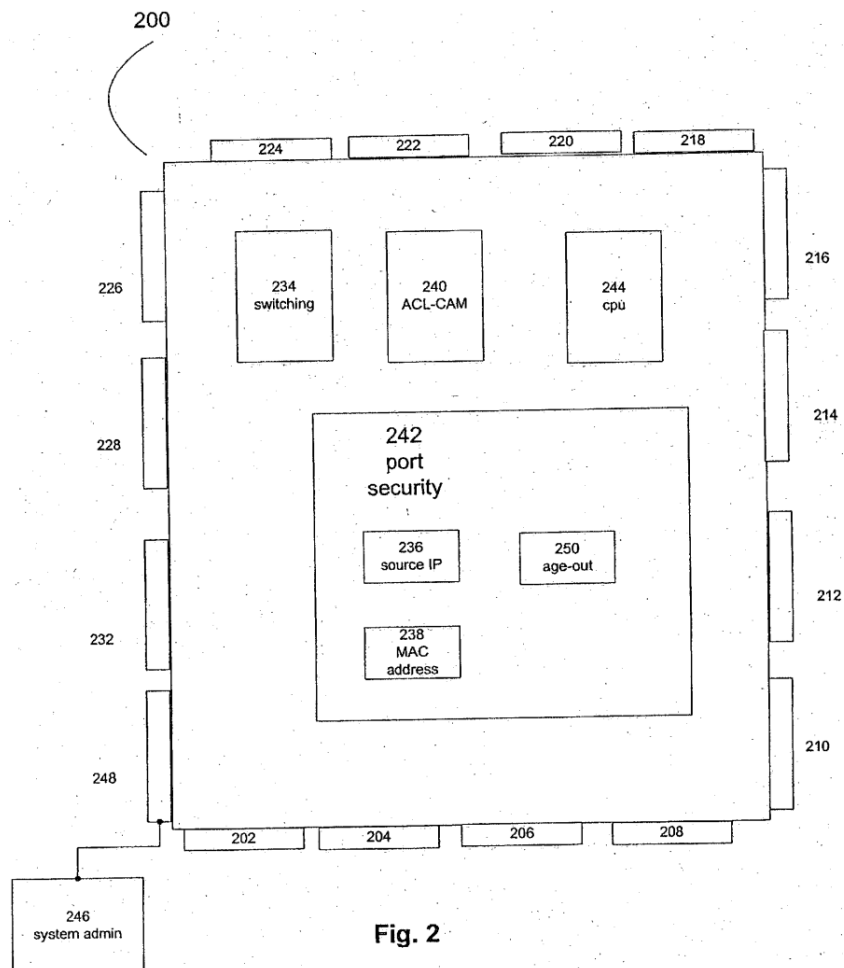


Fig. 2

Figure 2 illustrates network device 200 including the following components:

- a plurality of ports 202–232 and 248;
- network device switching 234;

- content addressable memory 240 containing a table that stores IP address/MAC address pairs “embodied as an access control list, which are data fields,” and shown as ACL-CAM 240;
- port security processor 242 with source IP address detector 236, MAC address detector 238, and source IP age-out timer 250; and
- central processing unit (CPU) 244.

Ex. 1007, 3:43–46, 3:35–55, 4:19–29, 4:34–39, 4:56–58, 4:66, 7:35–36, Fig. 2.

MAC address detector 238 “detects when a new host has been coupled to one of the ports 202–232” of network device 200. Ex. 1007, 4:19–22. After detecting when a new host has been coupled to a port, network device 200 stores the “MAC address for the new host” in a table that “correlates the MAC address for the new host with the port to which it is coupled,” e.g., a table stored in ACL-CAM 240. *Id.* at 4:22–26.

Source IP address detector 236 “automatically learns the source IP address for each MAC address entering a port” of network device 200. Ex. 1007, 4:56–58; *see id.* at 5:23–26. To “determine if the data packets received at a port contain a new source IP address that has not been learned, the ACL-CAM compares the source IP address and the MAC address in a received data packet, with the table of IP address/MAC address pairs.” *Id.* at 5:26–30. After learning a new IP address/MAC address pair, “ACL-CAM 240 is programmed with the information.” *Id.* at 5:35–36; *see id.* at 7:61–66, 8:43–48. After storing a new IP address/MAC pair in ACL-CAM 240, “blocking procedures are applied to the port.” *Id.* at 6:28–30.

Source IP address detector 236 can “learn the source IP address for a host in a number of different ways.” Ex. 1007, 5:38–39. As an example,

“the source IP address can be learned by using a reverse address resolution protocol (RARP) which provides for sending out the MAC address on the subnet, and in response the host having the MAC address responds with its IP address.” *Id.* at 5:39–43. As another example, source IP address detector 236 can “learn the IP address for a host by listening to the DHCP response packet being returned to the host” because that response packet contains the source IP address for the host. *Id.* at 5:44–47. When “a DHCP packet is detected, the entry in the table for the MAC address receiving the DHCP packet with a source IP address is cleared and the source IP address provided in the DHCP packet is loaded into the table.” *Id.* at 5:47–50.

Further, to reduce “the risk of possibly learning the wrong IP address,” a system administrator may “seed the IP learning process with the IP address properties that are consistent with the subnet to which the host is coupled.” Ex. 1007, 6:14–18. For instance, “if the IP Subnet is 10.32.1.0/24, then this information can be used to seed” the “learning process and only match a MAC address with an IP addresses consistent with the subnet 10.32.1.0/24 addresses,” thus eliminating the false learning of certain default addresses assigned “when a DHCP server is not initially located.” *Id.* at 6:18–25.

4. INDEPENDENT CLAIM 1

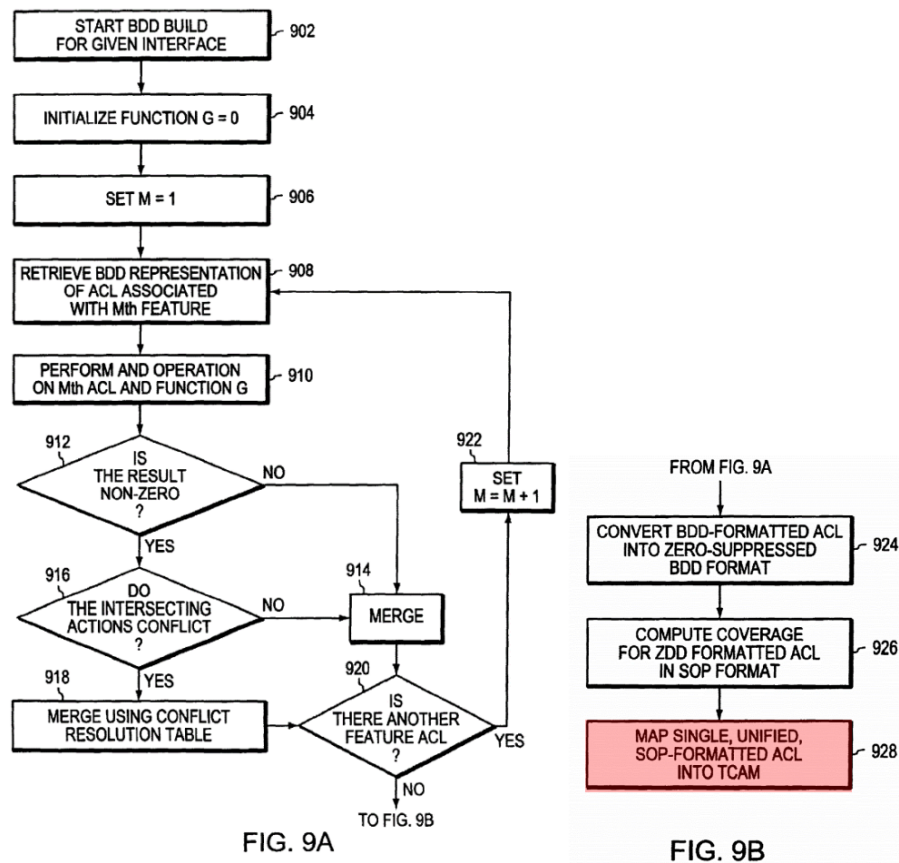
(a) Preamble

Claim 1 recites in its preamble “[a] method of developing an access control list.” Ex. 1001, 10:12.

Petitioner contends that the combined disclosures in Gai, Yip, and Kwan teach claim 1’s preamble for several related reasons. *See* Pet. 22–25. First, Petitioner asserts that Gai discloses “a method and apparatus for

efficiently organizing . . . access control lists (ACLs).” *Id.* at 22 (alteration by Petitioner) (emphasis omitted) (quoting Ex. 1006, 3:55–57). Petitioner also asserts that Gai’s Figures 9A and 9B illustrate “a method for dynamically generating optimized ACLs” stored in a ternary content addressable memory (TCAM). *Id.* at 23 (citing Ex. 1003 ¶ 79).

To support its assertions, Petitioner provides a highlighted version of Gai’s Figures 9A and 9B as reproduced below (Pet. 24):



Figures 9A and 9B depict steps performed by an ACL converter in a network switch to generate “a single, unified ACL for a given interface.” Ex. 1006, 12:20–22, Figs. 9A–9B. The above highlighted version of Figures 9A and 9B includes pink highlighting over step 928 for “map single,

unified SOP-formatted ACL into TCAM” in Figure 9B.² *See* Pet. 24; Ex. 1003 ¶ 79.

Second, Petitioner asserts that the combination with Yip modifies Gai’s method “such that the TCAM ACL is updated to include MAC addresses.” Pet. 24 (citing Ex. 1003 ¶ 81); *see id.* at 15–16, 20.

Third, Petitioner asserts that the combination with Kwan modifies Gai’s method “such that the TCAM ACL is updated to include . . . newly learned IP addresses.” Pet. 24 (citing Ex. 1003 ¶ 81); *see id.* at 18–20.

Therefore, according to Petitioner, “Gai in combination with Yip and Kwan discloses a method of dynamically generating an optimized TCAM ACL, which renders obvious the preamble.” Pet. 24–25 (citing Ex. 1003 ¶ 82).

As noted above, Patent Owner did not file a Preliminary Response. *See supra* § I. Nonetheless, the burden remains on Petitioner to demonstrate a reasonable likelihood of success. 35 U.S.C. § 314(a).

Generally, a preamble does not limit a claim. *Allen Eng’g Corp. v. Bartell Indus., Inc.*, 299 F.3d 1336, 1346 (Fed. Cir. 2002). We need not decide whether claim 1’s preamble limits the claim because, based on the current record and for the reasons stated by Petitioner and supported by Dr. Crovella’s testimony, Petitioner establishes sufficiently for purposes of institution that the combined disclosures in Gai, Yip, and Kwan teach claim 1’s preamble. *See* Pet. 22–25; Ex. 1003 ¶¶ 76–82.

² The acronym “SOP” stands for “sum of products.” Ex. 1006, 4:22–23, code (57).

(b) The First “Developing” Step

Claim 1 recites “developing an enhanced access control list including data related to at least one of user names, DNS names, Windows domain names, and physical addresses.” Ex. 1001, 10:14–16 (the first “developing” step).

Petitioner contends that the combined disclosures in Gai, Yip, and Kwan teach the first “developing” step for several related reasons. *See* Pet. 25–32. First, Petitioner asserts that Gai teaches “developing an enhanced access control list” because Gai discloses that an administrator creates one or more text-based ACLs used “for security” and to “implement additional features,” such as “packet filtering based on source and destination IP addresses, source and destination port, protocol, QoS, encryption, NAT, etc.” *Id.* at 25 (citing Ex. 1003 ¶ 85; Ex. 1006, 2:46–57, 3:24–30, 7:29–9:21, Figs. 5A–5E, 10A–10B); *see id.* at 28.

Second, Petitioner asserts that Yip teaches including “data related to . . . physical addresses” in an ACL because “Yip discloses that ‘access lists were developed to combat cyber-attacks on the LAN Switch by providing an administrator-controlled list of Internet Protocol (IP) addresses or media access control (MAC) addresses that were authorized to access the network.’” Pet. 26 (emphasis omitted) (quoting Ex. 1013, 2:43–47) (citing Ex. 1003 ¶¶ 89–90).

Additionally, Petitioner contends that Gai “contemplates that additional ‘message criteria’ may be included in the ACLs.” Pet. 26–27 (citing Ex. 1006, 7:32–34). According to Petitioner, an ordinarily skilled artisan “would have understood that MAC addresses are message criteria (since the Ethernet messages sent and received by Gai’s switch would have

contained source and destination MAC addresses) and also that the MAC addresses may be used to provide additional security.” *Id.* at 27.

Third, Petitioner asserts that Kwan teaches “how to automatically learn new IP addresses based on known MAC addresses” because “Kwan discloses that ‘the source IP address can be learned by using a reverse address resolution protocol (RARP) which provides for sending out the MAC address on the subnet, and in response the host having the MAC address responds with its IP address.’” Pet. 28–29 (quoting Ex. 1007, 5:40–43). Petitioner asserts that Kwan also discloses learning a new IP address “by listening to the DHCP response packet being returned to the host . . . [which] contains the source IP Address for the host.” *Id.* at 29 (alterations by Petitioner) (quoting Ex. 1007, 5:45–46).

Fourth, Petitioner asserts that “because Gai’s TCAM ACL is generated based on the text-based ACLs (which in the proposed combination would include MAC addresses), the TCAM ACL would also include both MAC and IP addresses, analogous to Kwan.” Pet. 30 (citing Ex. 1007, 4:34–39, 5:64–67).

Fifth, Petitioner asserts that “Kwan discloses that the system administrator provides seed IP address properties” representing “data (text data such as IP subnet) that is used to restrict access to the network such that only IP addresses consistent with the seed IP address properties will be learned and permitted to communicate on the network.” Pet. 30 (citing Ex. 1007, 6:14–25).

Sixth, Petitioner asserts that it would have been obvious “in view of Kwan” to “include seed IP address properties in Gai’s administrator-created text-based ACLs, to beneficially reduce or ‘eliminate false learning’ in

certain instances” and “beneficially help avoid spoofing.” Pet. 31 (quoting Ex. 1007, 6:22–23).

Therefore, according to Petitioner, “Gai in combination with Yip and Kwan discloses creating one or more text-based ACLs that include MAC addresses and seed IP address properties, which renders obvious this limitation.” Pet. 32 (citing Ex. 1003 ¶ 104).

Based on the current record and for the reasons stated by Petitioner and supported by Dr. Crovella’s testimony, Petitioner establishes sufficiently for purposes of institution that the combined disclosures in Gai, Yip, and Kwan teach claim 1’s first “developing” step. *See* Pet. 25–32; Ex. 1003 ¶¶ 83–104.

(c) The “Converting” Step

Claim 1 recites:

converting, in a network switch, at least one of:

user names into corresponding IP and physical addresses according to data in the enhanced access control list, wherein converting user names into corresponding IP and physical addresses according to data in the enhanced access control list includes: detecting login packets being communicated over the network; determining a MAC address from the login packets; detecting server message block login packets being communicated over the network; determining an IP address from the server message block login packets; and developing records in the access control list using the obtained IP address for the respective user name;

DNS names into corresponding IP addresses according to data in the enhanced access control list; and

physical addresses into IP addresses according to data in the enhanced access control list.

Ex. 1001, 10:17–35 (the “converting” step).

Petitioner contends that the combined disclosures in Gai, Yip, and Kwan teach the “converting” step, i.e., “converting, in a network switch, . . . physical addresses into IP addresses according to data in the enhanced access control list,” for several related reasons. *See* Pet. 32–39. First, Petitioner asserts that Gai teaches “a network switch” because Gai discloses network device 316 with circuitry for performing “switching functions” in computer network 300. *Id.* at 32 (citing Ex. 1006, 5:22–34, 6:62–65).

Second, Petitioner asserts that Kwan discloses two techniques for “converting, in a network switch, . . . physical addresses into IP addresses” as follows: (1) converting using the DHCP technique; and (2) converting using the RARP technique. *See* Pet. 36–38. Regarding the DHCP technique, Petitioner contends that “Kwan discloses listening to DHCP response packets to learn new IP addresses according to MAC addresses.” *Id.* at 36. Regarding the RARP technique, Petitioner contends that “Kwan discloses that ‘the source IP address can be learned by using a reverse address resolution protocol (RARP) which provides for sending out the MAC address on the subnet, and in response the host having the MAC address responds with its IP address.’” *Id.* at 37 (quoting Ex. 1007, 5:40–43).

Third, Petitioner asserts that an ordinarily skilled artisan “would have implemented Gai such that its network device 316 converts physical addresses of devices on the network into new IP addresses according to data (e.g., known MAC addresses and seed IP address properties), which in the proposed combination would be included in one or more text-based ACLs” corresponding to the claimed “access control list.” Pet. 38–39 (citing Ex. 1003 ¶ 122).

Therefore, according to Petitioner, “Gai in combination with Yip and Kwan discloses converting in the network device 316 physical addresses of devices on the network into IP addresses (e.g., using a DHCP technique or a RARP technique) according to data (e.g., known MAC addresses and seed IP address) in one or more text-based ACLs, which renders obvious this limitation.” Pet. 39 (citing Ex. 1003 ¶ 123).

Based on the current record and for the reasons stated by Petitioner and supported by Dr. Crovella’s testimony, Petitioner establishes sufficiently for purposes of institution that the combined disclosures in Gai, Yip, and Kwan teach claim 1’s “converting” step. *See* Pet. 32–39; Ex. 1003 ¶¶ 105–123.

(d) The Second “Developing” Step

Claim 1 recites “developing, in the network switch, the access control list from each of the operations of converting.” Ex. 1001, 10:36–37 (the second “developing” step).

Petitioner contends that the combined disclosures in Gai, Yip, and Kwan teach the second “developing” step for several related reasons. *See* Pet. 39–42. First, Petitioner asserts that “Kwan discloses that once a new IP address has been successfully learned (e.g., using a DHCP technique or a RARP technique), the learned IP address is ‘loaded into the table.’” *Id.* at 40 (emphasis omitted) (quoting Ex. 1007, 5:50). According to Petitioner, Kwan’s “table is embodied as an access control list, which are data fields, included in a content addressable memory 240, which is referred to as an ACL-CAM.” *Id.* (emphases omitted) (quoting Ex. 1007, 4:36–39).

Second, Petitioner asserts that an ordinarily skilled artisan “would have recognized that Kwan’s CAM is a specialized memory circuit that is

designed to allow its entire contents to be searched for a matching value all at once and, consequently, very fast.” Pet. 40 (citing Ex. 1007, 4:35–39).

Third, Petitioner asserts that an ordinarily skilled artisan “implementing Kwan’s DHCP or RARP techniques in the combination with Gai” would have found it obvious to “update Gai’s TCAM ACL with the new IP address once learned, so that the network device 314 would permit packets that have the new IP address.” Pet. 41 (citing Ex. 1006, 18:3–5). According to Petitioner, “[u]pdating the TCAM ACL is consistent with Gai’s recognition that ACLs may be ‘routinely up-dated or changed.’” *Id.* (emphasis omitted) (quoting Ex. 1006, 18:47–48). Further, Petitioner asserts that “Kwan recognizes the importance of updating the ACL-CAM because ‘the source IP addresses assigned for a given host can change over time.’” *Id.* (emphasis omitted) (quoting Ex. 1007, 7:67–8:1).

Fourth, Petitioner asserts that an ordinarily skilled artisan “seeking to update Gai’s TCAM ACL to include newly learned IP addresses (per Kwan)” would have found it obvious to “first update one or more of Gai’s text-based ACLs to include the new IP addresses, and then to dynamically reoptimize the updated text-based ACLs to generate an updated TCAM ACL that includes the new IP addresses.” Pet. 41–42. According to Petitioner, “[t]his processing sequence would allow for Gai’s network device 316 to apply [the] optimized TACM ACLS with the new IP addresses to received packets.” *Id.* at 42.

Therefore, according to Petitioner, “Gai in combination with Yip and Kwan discloses developing a reoptimized TCAM ACL in the network device 314 that includes new IP addresses from the operation of converting, which renders obvious this limitation.” Pet. 42 (citing Ex. 1003 ¶ 131).

Based on the current record and for the reasons stated by Petitioner and supported by Dr. Crovella's testimony, Petitioner establishes sufficiently for purposes of institution that the combined disclosures in Gai, Yip, and Kwan teach claim 1's second "developing" step. *See* Pet. 39–42; Ex. 1003 ¶¶ 124–131.

(e) Alleged Reasons to Combine the Teachings of the References

Petitioner identifies reasons that would have prompted an ordinarily skilled artisan to combine the teachings of Gai, Yip, and Kwan in the way Petitioner proposes, e.g., to provide "additional security" and "improve security." Pet. 14–21, 27, 29–31, 38. Further, Petitioner asserts that an ordinarily skilled artisan would have had a "reasonable expectation of success" in combining the teachings of the references. *Id.* at 16–17, 22, 38.

Dr. Crovella's testimony supports Petitioner's positions. *See, e.g.,* Ex. 1003 ¶¶ 51–75, 91–92, 98, 101, 121. For instance, Dr. Crovella testifies that including the "MAC address [per Yip] in Gai's text-based ACLs would beneficially allow for dynamically generating TCAM ACLs with MAC addresses to thereby restrict or permit access based on MAC address" and "would improve security because only specific MAC addresses would be permitted to communicate packets over the network." *Id.* ¶ 57. Further, Dr. Crovella testifies that by including the "MAC address [per Yip] and seed IP address properties [per Kwan] in the text-based ACLs, Gai's network device 316 would be able to process packets from a host that has a known MAC address but has been assigned a new IP address (e.g., using DHCP), as was often done at the time." *Id.* ¶ 70 (citing Ex. 1007, 1:66–2:1).

Dr. Crovella explains that "[s]uch an implementation would obviate the need for Gai's administrator to manually generate updated text-based

ACLs when the network changes.” Ex. 1003 ¶ 70 (citing Ex. 1006, 2:57–60, 4:6–13, 7:16–20, 18:3–5; Ex. 1007, 7:67–8:1).

Based on the current record, Petitioner establishes sufficiently for purposes of institution that an ordinarily skilled artisan would have had reasons, i.e., as articulated by Dr. Crovella, to combine the teachings of Gai, Yip, and Kwan in the way Petitioner proposes. *See* Pet. 14–21, 27, 29–31, 38; Ex. 1003 ¶¶ 51–58, 62–72, 91–92, 98, 101, 121. Based on the current record, Petitioner also establishes sufficiently that an ordinarily skilled artisan would have had a reasonable expectation of success in combining the teachings of the references. *See* Pet. 16–17, 22, 38; Ex. 1003 ¶¶ 59–60, 73–74, 121.

(f) Preliminary Conclusion About Alleged Obviousness

As discussed above, Petitioner’s analysis addresses how the combined disclosures in Gai, Yip, and Kwan teach claim 1’s subject matter. *See supra* §§ III.D.4(a)–(d). Additionally, Petitioner provides reasons with rational underpinnings as to why an ordinarily skilled artisan would have been motivated to combine the teachings of Gai, Yip, and Kwan in the way Petitioner proposes and would have had a reasonable expectation of success. *See supra* § III.D.4(e). Hence, Petitioner demonstrates a reasonable likelihood of proving that claim 1 is unpatentable under § 103(a) because the claim would have been obvious over Gai, Yip, and Kwan.

5. DEPENDENT CLAIM 3

Claim 3 depends directly from claim 1 and specifies that “each physical address comprises a MAC address.” Ex. 1001, 10:42–43.

Petitioner contends that “the combination of Gai, Yip, and Kwan discloses that the text-based ACLs include MAC addresses of devices

authorized to access the network,” thus rendering claim 3 obvious. Pet. 42 (citing Ex. 1003 ¶ 132).

Based on the current record and for the reasons stated by Petitioner and supported by Dr. Crovella’s testimony, Petitioner establishes sufficiently for purposes of institution that the combined disclosures in Gai, Yip, and Kwan teach claim 3’s subject matter. *See* Pet. 42; Ex. 1003 ¶ 132. Hence, Petitioner demonstrates a reasonable likelihood of proving that claim 3 is unpatentable under § 103(a) because the claim would have been obvious over Gai, Yip, and Kwan.

E. Alleged Obviousness over Gai, Yip, Kwan, and Georgiou: Claims 15, 17, and 19–21

Petitioner contends that claims 15, 17, and 19–21 are unpatentable under § 103(a) because the claims would have been obvious over Gai, Yip, Kwan, and Georgiou. *See* Pet. 43–85. Above, we provided overviews of Gai, Yip, and Kwan. *See supra* §§ III.D.1–III.D.3. Below, we provide an overview of Georgiou. Then, we consider obviousness issues. For the reasons explained below, Petitioner establishes sufficiently for purposes of institution that the combined disclosures in Gai, Yip, Kwan, and Georgiou teach the subject matter in claims 15, 17, and 19–21.

1. OVERVIEW OF GEORGIOU (EXHIBIT 1008)

Georgiou is a U.S. patent titled “Dynamic Reallocation of Data Stored in Buffers Based on Packet Size,” filed on July 9, 2003, and issued on February 21, 2006. Ex. 1008, codes (12), (22), (45), (54). Georgiou states that the invention “generally relates to the implementation of computer networks, and more particularly, to a method and apparatus for efficient

utilization of memory and reduced memory waste in storing data blocks of network traffic.” *Id.* at 1:8–11; *see id.* at code (57).

Georgiou explains that in various networks packets of varying sizes are sent and received and that “various techniques are used to manage the incoming stream of varying sizes of packets.” Ex. 1008, 1:13–16, 1:20–21. Georgiou also explains that “[t]ypical arrangements include employing some form of logic, either hardware centric or software centric, to receive and temporarily store the incoming packets in a memory area, e.g., a buffer pool or equivalent, until subsequent processing can access and retrieve the packets, as appropriate, for action on the packet contents.” *Id.* at 1:22–27.

Georgiou endeavors to provide “a flexible buffer management method and system for efficient utilization of memory and reducing memory waste in storing data blocks (e.g., packets) of network traffic.” Ex. 1008, 3:14–17. In particular, “after reception of a packet from the network interface, the packet is typically temporarily stored in an inbound buffer in the network interface from where it is then transferred to memory to await subsequent processing.” *Id.* at 3:17–22.

Buffer memory includes “two or more portions (i.e., buffer pools) according to packet sizes with buffers allocated in each portion suitable for storing packets according to the packet sizes associated with the portion.” Ex. 1008, 3:22–25. Hence, one portion of buffer memory is used for smaller packet sizes, and another portion of buffer memory is used for larger packet sizes. *Id.* at code (57). “This buffer sizing leads to optimization of memory utilization which is directly related to overhead costs such as memory costs, speed performance, packet processing reliability, and the like.” *Id.* at 3:57–61.

The “allocation of available memory area is performed dynamically.” Ex. 1008, 5:54–55. Thus, if “the number of available buffers in one set falls under a particular threshold, free memory area from the other set can be reassigned to the first set.” *Id.* at 5:60–63. When reassigning free memory area, “big memory buffers are partitioned into a number of small buffers,” or “multiple small buffers are merged into a larger buffer.” *Id.* at 5:63–66. “This dynamic reassignment of buffers makes [the] system adaptable to sudden network traffic pattern change.” *Id.* at 5:66–67.

2. INDEPENDENT CLAIM 15

(a) Preamble

Claim 15 recites in its preamble “[a] network switching circuit.” Ex. 1001, 11:57.

Petitioner contends that Gai teaches claim 15’s preamble because Gai discloses network device 316 with circuitry for performing “switching functions” in computer network 300. Pet. 48 (citing Ex. 1003 ¶ 147; Ex. 1006, 5:3–59, 6:27–65).

As noted above, Patent Owner did not file a Preliminary Response. *See supra* § I. Nonetheless, the burden remains on Petitioner to demonstrate a reasonable likelihood of success. 35 U.S.C. § 314(a).

Generally, a preamble does not limit a claim. *Allen Eng’g*, 299 F.3d at 1346. We need not decide whether claim 15’s preamble limits the claim because, based on the current record and for the reasons stated by Petitioner and supported by Dr. Crovella’s testimony, Petitioner establishes sufficiently for purposes of institution that Gai teaches claim 15’s preamble. *See* Pet. 48–49; Ex. 1003 ¶¶ 76–82.

(b) The “Forwarding Circuit” Limitation

Claim 15 recites:

a forwarding circuit

operable to detect specific received packets and to provide the specific packets on a processor port, and

further operable to receive packets on one of a plurality of ports including the processor port and to forward each received packet to a port corresponding to a destination address contained in the packet subject to access restrictions contained in a dynamic access control list.

Ex. 1001, 11:58–12:3 (the “forwarding circuit” limitation).

Petitioner contends that the combined disclosures in Gai, Yip, and Kwan teach the “forwarding circuit” limitation for several related reasons. *See* Pet. 49–65. First, Petitioner asserts that Gai teaches “a forwarding circuit” because Gai’s network device 316 includes forwarding entity 404. *Id.* at 49–50 (citing Ex. 1003 ¶¶ 150–151; Ex. 1006, 3:37–40, 6:18–30, 8:26–27, 17:58–63).

Second, Petitioner asserts that “Gai in combination with Kwan” teaches “operable to detect specific received packets” because:

- (1) Gai’s forwarding entity 404 is configured to “forward or switch network messages” corresponding to “received packets” communicated over computer network 300;
- (2) Gai’s forwarding entity 404 utilizes “a TCAM ACL to identify specific received packets that need additional processing (e.g., packets that should be encrypted, logged, etc.)”;
- (3) Kwan’s network device 200 includes “an access control list, which are data fields, included in a content addressable memory 240,” i.e., ACL-CAM 240;

- (4) Kwan's ACL-CAM 240 contains "source IP address and MAC address pairs" used to (a) "identify specific received packets that have a new IP address that has not been learned" and (b) "dynamically learn the new IP addresses based on the MAC addresses."

Pet. 50–52 (citing Ex. 1003 ¶¶ 153–156; Ex. 1006, 1:37–2:41, 4:41–42, 5:3–59, 5:66–6:12, 6:27–30, 7:32–34, 9:27–31, 11:60–12:7, 13:3–45; Ex. 1007, 4:34–48, 5:25–30, 5:38–67, 6:28–67, 7:61–66, 8:17–51, Fig. 3) (emphasis omitted).

Third, Petitioner asserts that "Gai in combination with Kwan" teaches "operable . . . to provide the specific packets on a processor port" because:

- (1) Gai's network device 316 includes CPU 406;
- (2) Gai's forwarding entity 404 provides packets to a port of CPU 406, e.g., to perform encryption; and
- (3) "Kwan discloses that if a packet has a new IP address, a source IP address detector 236 of a port security processor 242 is used to learn the new IP address."

Pet. 53–55 (citing Ex. 1003 ¶¶ 159–163; Ex. 1006, 4:29–37, 7:34–43, 11:60–12:7, Fig. 4; Ex. 1007, 5:38–67, 8:17–47, Fig. 3).

Fourth, Petitioner asserts that "Gai in combination with Kwan" teaches "further operable to receive packets on one of a plurality of ports including the processor port" because:

- (1) "Gai's forwarding entity 404 receives IP messages (packets) on a plurality of interfaces 402a-e corresponding to ports";
- (2) "in the combination of Gai and Kwan the forwarding entity 404 forwards to the CPU 406 specific packets";
- (3) "[i]t would have been obvious to" an ordinarily skilled artisan "for Gai's CPU 406 to send the packets back to the forwarding entity 404 as appropriate," e.g., after

encryption to output on a port and transmit to a destination; and

- (4) “Kwan’s packets with a new IP address are disclosed as being sent to the destination after successfully learning the IP address.”

Pet. 56–60 (citing Ex. 1003 ¶¶ 166–197; Ex. 1006, 1:37–2:41, 4:41–42, 5:14–22, 6:27–30, 7:66–8:6, 8:36–55, 16:63–17:6, Fig. 4; Ex. 1007, 5:30–36, 8:43–55, Fig. 3).

Fifth, Petitioner asserts that Gai teaches “further operable . . . to forward each received packet to a port corresponding to a destination address contained in the packet” because “Gai’s forwarding entity 404 is configured to forward each incoming packet to an output port corresponding to an IP destination address extracted from the packet.” Pet. 60–62 (citing Ex. 1003 ¶¶ 114, 174–175, 177–179; Ex. 1006, 5:60–66, 6:27–30, 19:6–14, Fig. 4). According to Petitioner, an ordinarily skilled artisan “would have understood that the forwarding entity 404 selects an interface and associated port for forwarding based on the destination address contained in the message (e.g., packet), as was common in the art.” *Id.* at 61 (citing Ex. 1003 ¶ 176; Ex. 1007, 1:52–61, 2:1–51, 3:53–4:6; Ex. 1008, 1:49–54).

Sixth, Petitioner asserts that “Gai in combination with Yip and Kwan” teaches “subject to access restrictions contained in a dynamic access control list” because:

- (1) the combined disclosures in the references teach the claimed “dynamic access control list” for the reasons discussed for claim 1’s preamble and second “developing” step;
- (2) Gai’s forwarding entity 404 uses a TCAM ACL “to determine if access should be restricted (e.g., ‘deny’) or if forwarding is permitted (‘permit’); and

- (3) Kwan discloses “an access control list, which are data fields, included in a content addressable memory 240, which is referred to as an ACL-CAM . . . [to] control access and switching through the ports.”

Pet. 63–65 (alterations by Petitioner) (emphases omitted) (citing Ex. 1003 ¶¶ 180–187; Ex. 1006, 4:26–30, 7:24–39, 8:7–27, 9:31–36, 17:28–48, Figs. 5A–5E, 8, 10A–10B, 11; Ex. 1007, 4:34–48, 4:56–63, 7:61–66, 8:14–46, 9:4–16, code (57), Figs. 3–4).

Based on the current record and for the reasons stated by Petitioner and supported by Dr. Crovella’s testimony, Petitioner establishes sufficiently for purposes of institution that the combined disclosures in Gai, Yip, and Kwan teach claim 15’s “forwarding circuit” limitation. *See* Pet. 49–65; Ex. 1003 ¶¶ 150–187.

(c) The “Memory Circuit” Limitation

Claim 15 recites:

a memory circuit coupled to the forwarding circuit, the memory circuit
operable to store packets and
operable to store an enhanced access control list
and a dynamic access control list.

Ex. 1001, 12:4–7 (the “memory circuit” limitation).

Petitioner contends that the combined disclosures in Gai, Yip, Kwan, and Georgiou teach the “memory circuit” limitation for several related reasons. *See* Pet. 65–69. First, Petitioner asserts Gai teaches “a memory circuit coupled to the forwarding circuit” because Gai’s network device 316 includes forwarding entity 404, NVRAM 408, dynamic memory 409, and TCAM 410 with forwarding entity 404 (i) directly coupled to TCAM 410 and (ii) indirectly coupled to NVRAM 408 and dynamic memory 409 via

CPU 406. *Id.* at 65–66 (citing Ex. 1003 ¶¶ 188–189; Ex. 1006, 5:65–6:12, Fig. 4); *see id.* at 67.

Second, Petitioner asserts that “Gai in combination with Georgiou” teaches “operable to store packets” because:

- (1) Gai’s network device 316 includes dynamic memory 409 as part of the claimed “memory circuit”; and
- (2) Georgiou discloses that “[t]ypical arrangements include . . . receive and temporarily store the incoming packets in a memory . . . until subsequent processing can access and retrieve the packets, as appropriate, for action on the packet contents.”

Pet. 67–68 (alterations by Petitioner) (emphasis omitted) (quoting Ex. 1008, 1:22–27) (citing Ex. 1003 ¶¶ 192–194; Ex. 1008, 1:13–21, 3:17–22, 5:51–67).

According to Petitioner, “it would have been obvious to” an ordinarily skilled artisan in view of Georgiou “to store in Gai’s dynamic memory 409 (analogous to the dynamic memory of Georgiou) packets forwarded to CPU 406 for processing, until CPU 406 has sufficient processing bandwidth.” Pet. 67–68.

Third, Petitioner asserts that “Gai in combination with Yip and Kwan” teaches “operable to store an enhanced access control list and a dynamic access control list” because:

- (1) the combined disclosures in the references teach the claimed “enhanced access control list” for the reasons discussed for claim 1’s first “developing” step;
- (2) Gai’s network device 316 includes NVRAM 408 as part of the claimed “memory circuit”;
- (3) Gai discloses NVRAM 408 as operable to store “administrator generated text-based ACLs (including

MAC addresses and seed IP address properties)” that correspond to the claimed “enhanced access control list”;

- (4) the combined disclosures in the references teach the claimed “dynamic access control list” for the reasons discussed for claim 1’s preamble and second “developing” step;
- (5) Gai’s network device 316 includes TCAM 410 as part of the claimed “memory circuit”; and
- (6) Gai discloses TCAM 410 and its associated memory 411 as operable to store dynamically generated and optimized “versions of the ACLs” that correspond to the claimed “dynamic access control list.”

Pet. 68–69 (emphasis omitted) (citing Ex. 1003 ¶¶ 197–204; Ex. 1006, 2:58–65, 4:8–13, 4:26–29, 6:13–18, 7:15–8:6, 9:18–25, 20:65–67).

Based on the current record and for the reasons stated by Petitioner and supported by Dr. Crovella’s testimony, Petitioner establishes sufficiently for purposes of institution that the combined disclosures in Gai, Yip, Kwan, and Georgiou teach claim 15’s “memory circuit” limitation. *See* Pet. 65–69; Ex. 1003 ¶¶ 188–204.

(d) The “Processor” Limitation

Claim 15 recites:

a processor coupled to the forwarding circuit and to the memory circuit, the processor

operable to define the specific packets detected by the forwarding circuit and

operable to process the specific packets stored in the memory circuit using the enhanced access control list to generate the dynamic access control list and store the dynamic access control list in the memory circuit, and

further operable to provide the specific packets to the processor port of the forwarding circuit after processing the packets.

Ex. 1001, 12:8–17 (the “processor” limitation).

Petitioner contends that the combined disclosures in Gai, Yip, Kwan, and Georgiou teach the “processor” limitation for several related reasons. *See* Pet. 70–79. First, Petitioner asserts that Gai teaches “a processor coupled to the forwarding circuit and to the memory circuit” because Gai’s network device 316 includes forwarding entity 404, CPU 406, NVRAM 408, dynamic memory 409, and TCAM 410 with CPU 406 (i) directly coupled to forwarding entity 404, NVRAM 408, and dynamic memory 409 and (ii) indirectly coupled to TCAM 410 via ACL converter 424. *Id.* at 70–71 (citing Ex. 1003 ¶¶ 205–208; Ex. 1006, 6:13–27, 6:36–41, 6:52–55, Fig. 4).

Second, Petitioner asserts that “Gai in combination with Yip and Kwan” teaches “operable to define the specific packets detected by the forwarding circuit” because:

- (1) “Gai in combination with Kwan discloses that forwarding entity 404 identifies specific received packets that need additional processing (e.g., encrypted, logged, etc., and packets that have a new IP address that has not been learned)” for the reasons discussed for claim 15’s “forwarding circuit” limitation;
- (2) “Gai in combination with Yip and Kwan discloses that Gai’s CPU 406 executes ACL converter 424 software to dynamically generate an optimized TCAM ACL from one or more text-based ACLs” for the reasons discussed for claim 1’s preamble and second “developing” step; and
- (3) Gai’s CPU 406 “define[s] the specific packets” that forwarding entity 404 identifies for additional processing by (a) “executing ACL converter 424” to “dynamically

generate” the unified ACL for the TCAM and
(b) “storing the unified ACL in the TCAM.”

Pet. 71–73 (emphasis omitted) (citing Ex. 1003 ¶¶ 209–213; Ex. 1006, 6:52–55, 9:4–27, 9:42–11:4, 12:20–16:17, Figs. 4, 6–7, 9A–9B).

Third, Petitioner asserts that “Gai in combination with Yip, Kwan, and Georgiou” teaches “operable to process the specific packets stored in the memory circuit using the enhanced access control list to generate the dynamic access control list and store the dynamic access control list in the memory circuit” because:

- (1) “Gai in combination with Georgiou discloses storing packets for processing in dynamic memory 409” for the reasons discussed for claim 15’s “memory circuit” limitation;
- (2) Georgiou discloses that the “process[or] can access and retrieve the packets [from memory], as appropriate, for action on the packet contents”;
- (3) Kwan discloses that “port security processor 242 is used to learn the new IP addresses”;
- (4) Gai’s CPU 406 would in the learning process use “seed IP address properties in one or more text-based ACLs and only match a new IP address with a MAC address that is consistent”;
- (5) Gai’s CPU 406 “would then use the one or more text-based ACLs (including the newly learned IP address) to dynamically generate an optimized TCAM ACL”; and
- (6) Gai’s TCAM 410 would store the dynamically generated and optimized ACL.

Pet. 73–76 (alterations by Petitioner) (emphasis omitted) (quoting Ex. 1008, 1:26–27) (citing Ex. 1003 ¶¶ 214–221; Ex. 1006, 6:52–59, 9:22–10:48,

20:38–53, 22:15–29; Ex. 1007, 4:56–58, 5:38–67, 6:14–25, 8:17–47, Fig. 3; Ex. 1008, 3:17–22).

Fourth, Petitioner asserts that “Gai in combination with Kwan” teaches “further operable to provide the specific packets to the processor port of the forwarding circuit after processing the packets” because:

- (1) “Gai in combination with Kwan discloses that the forwarding entity 404 is operable to receive packets on one of a plurality of ports including CPU 406 port” for the reasons discussed for claim 15’s “forwarding circuit” limitation;
- (2) “it would have been obvious to” an ordinarily skilled artisan “for Gai’s CPU 406 to provide the packets after encryption and learning a new IP addresses (among other packet processing operations) to its port so that the packet may be sent to the destination”;
- (3) Kwan’s Figure 3 shows that packets are passed to “a destination port once the IP address has been learned”; and
- (4) “it would have been obvious to” an ordinarily skilled artisan “for Gai’s CPU 406 to provide packets, after successfully learning a new IP address, to the forwarding entity 404 port so that the packets may be passed to the destination device.”

Pet. 76–79 (citing Ex. 1003 ¶¶ 222–226; Ex. 1006, Fig. 4; Ex. 1007, Fig. 3).

Based on the current record and for the reasons stated by Petitioner and supported by Dr. Crovella’s testimony, Petitioner establishes sufficiently for purposes of institution that the combined disclosures in Gai, Yip, Kwan, and Georgiou teach claim 15’s “processor” limitation. *See* Pet. 70–79; Ex. 1003 ¶¶ 205–226.

(e) Alleged Reasons to Combine the Teachings of the References

Petitioner identifies reasons that would have prompted an ordinarily skilled artisan to combine the teachings of Gai, Yip, Kwan, and Georgiou in the way Petitioner proposes, e.g., to “facilitate[e] processing of packets (e.g., encrypt, log, learn new IP address, etc.).” Pet. 44–47, 53, 64, 68; *see id.* at 14–21, 27, 29–31, 38. Further, Petitioner asserts that an ordinarily skilled artisan would have had a “reasonable expectation of success” in combining the teachings of the references. *Id.* at 47; *see id.* at 16–17, 22, 38.

Dr. Crovella’s testimony supports Petitioner’s positions. *See* Ex. 1003 ¶¶ 51–75, 91–92, 98, 101, 121, 135–145, 157, 186, 195. For instance, Dr. Crovella testifies that storing IP packets “in dynamic memory while awaiting processing” according to Georgiou “would allow the packets awaiting processing to be retained until the CPU is freed up, thereby facilitating processing of packets (e.g., encrypt, log, learn new IP address, etc.).” *Id.* ¶¶ 139–140. Further, Dr. Crovella testifies that the proposed combination “would also reduce the number of packets that need to be retransmitted (since fewer packets would be dropped), and thus, make more efficient use of Gai’s network bandwidth.” *Id.* ¶ 141; *see id.* ¶ 195.

Based on the current record, Petitioner establishes sufficiently for purposes of institution that an ordinarily skilled artisan would have had reasons, i.e., as articulated by Dr. Crovella, to combine the teachings of Gai, Yip, Kwan, and Georgiou in the way Petitioner proposes. *See* Pet. 14–21, 27, 29–31, 38, 44–47, 53, 64, 68; Ex. 1003 ¶¶ 51–58, 62–72, 91–92, 98, 101, 121, 135–144, 157, 186, 195. Based on the current record, Petitioner also establishes sufficiently that an ordinarily skilled artisan would have had a

reasonable expectation of success in combining the teachings of the references. *See* Pet. 16–17, 22, 38, 47; Ex. 1003 ¶¶ 59–60, 73–74, 121, 145.

(f) Preliminary Conclusion About Alleged Obviousness

As discussed above, Petitioner’s analysis addresses how the combined disclosures in Gai, Yip, Kwan, and Georgiou teach claim 15’s subject matter. *See supra* §§ III.E.2(a)–(d). Additionally, Petitioner provides reasons with rational underpinnings as to why an ordinarily skilled artisan would have been motivated to combine the teachings of Gai, Yip, Kwan, and Georgiou in the way Petitioner proposes and would have had a reasonable expectation of success. *See supra* § III.E.2(e). Hence, Petitioner demonstrates a reasonable likelihood of proving that claim 15 is unpatentable under § 103(a) because the claim would have been obvious over Gai, Yip, Kwan, and Georgiou.

3. INDEPENDENT CLAIM 19

Claim 19 recites in its preamble “[a] computer network.” Ex. 1001, 12:27.

Petitioner contends that Gai teaches claim 19’s preamble because Gai discloses computer network 300. Pet. 79 (citing Ex. 1003 ¶ 230).

Claim 19 recites “a network switch” including a “forwarding circuit,” a “memory circuit,” and a “processor” as similarly recited in claim 15. Ex. 1001, 12:28–50; *see id.* at 11:58–12:17 (claim 15).

Petitioner asserts that Gai teaches “a network switch” because Gai discloses network device 316 with circuitry for performing “switching functions” in computer network 300. Pet. 79–80 (citing Ex. 1003 ¶ 231). Petitioner asserts that the combined disclosures in Gai, Yip, Kwan, and Georgiou teach claim 19’s “forwarding circuit,” “memory circuit,” and

“processor” limitations for the reasons that the combined disclosures teach the similar limitations in claim 15. *Id.* at 80–81; *see id.* at 49–79.

Claim 19 recites “a plurality of hosts, each host coupled to a respective port of the network switch.” Ex. 1001, 12:51–52 (the “plurality of hosts” limitation).

Petitioner contends that Gai teaches the “plurality of hosts” limitation because:

- (1) Gai’s network device 316 includes ports;
- (2) Gai’s computer network 300 includes end stations 306–312 coupled via LAN 302 or LAN 304 to network device 316; and
- (3) end stations 306–312 “correspond to ‘hosts’ as that term is used in the ’518 patent.”

Pet. 81–83 (emphasis omitted) (citing Ex. 1001, 1:5–21, 2:37–40; Ex. 1003 ¶¶ 246–252; Ex. 1006, 2:1–3, 5:3–13, 5:60–66, 17:4–6, Fig. 3; Ex. 1007, 3:46–50).

Based on the current record and for the reasons stated by Petitioner and supported by Dr. Crovella’s testimony, Petitioner establishes sufficiently for purposes of institution that the combined disclosures in Gai, Yip, Kwan, and Georgiou teach claim 19’s subject matter. *See* Pet. 79–83; Ex. 1003 ¶¶ 230–252. Additionally, as discussed above, Petitioner provides reasons with rational underpinnings as to why an ordinarily skilled artisan would have been motivated to combine the teachings of Gai, Yip, Kwan, and Georgiou in the way Petitioner proposes and would have had a reasonable expectation of success. *See supra* § III.E.2(e). Hence, Petitioner demonstrates a reasonable likelihood of proving that claim 19 is

unpatentable under § 103(a) because the claim would have been obvious over Gai, Yip, Kwan, and Georgiou.

4. DEPENDENT CLAIMS 17, 20, AND 21

Claim 17 depends directly from claim 15. Ex. 1001, 12:21–23.

Claims 20 and 21 depend directly from claim 19. *Id.* at 12:53–58.

For claims 17, 20, and 21, Petitioner provides an analysis supported by Dr. Crovella’s testimony about how the combined disclosures in Gai, Yip, Kwan, and Georgiou teach the claimed subject matter. *See* Pet. 79, 84; Ex. 1003 ¶¶ 227–229, 253–257.

Based on the current record and for the reasons stated by Petitioner and supported by Dr. Crovella’s testimony, Petitioner establishes sufficiently for purposes of institution that the combined disclosures in Gai, Yip, Kwan, and Georgiou teach the subject matter in claims 17, 20, and 21. *See* Pet. 79, 84; Ex. 1003 ¶¶ 227–229, 253–257. Hence, Petitioner demonstrates a reasonable likelihood of proving that claims 17, 20, and 21 are unpatentable under § 103(a) because the claims would have been obvious over Gai, Yip, Kwan, and Georgiou.

IV. CONCLUSION

Based on the arguments and evidence presented by the parties, we determine that there is a reasonable likelihood Petitioner would prevail with respect to at least one claim challenged in the Petition. Hence, we institute an *inter partes* review of all challenged claims on all challenges included in the Petition. *See SAS Inst. Inc. v. Iancu*, 138 S. Ct. 1348, 1355 (2018) (noting that the language of 35 U.S.C. § 314(b) “indicates a binary choice—either institute review or don’t”); *PGS Geophysical AS v. Iancu*, 891 F.3d 1354, 1360 (Fed. Cir. 2018) (interpreting the statute as requiring “a simple

yes-or-no institution choice respecting a petition, embracing all challenges included in the petition”); 37 C.F.R. § 42.108(a).

At this preliminary stage, we have not made a final determination about the patentability of any challenged claim, the construction of any claim term, phrase, or limitation, or any other legal or factual issue.

V. ORDER

Accordingly, it is

ORDERED that, according to 35 U.S.C. § 314(a), an *inter partes* review of claims 1, 3, 15, 17, and 19–21 in the ’518 patent is instituted on all challenges included in the Petition; and

FURTHER ORDERED that, according to 35 U.S.C. § 314(c) and 37 C.F.R. § 42.4, notice is hereby given of the institution of a trial that commences on the entry date of this decision.

IPR2024-00734
Patent 7,623,518 B2

For PETITIONER:

Theodore M. Foster
David L. McCombs
Gregory P. Huh
HAYNES AND BOONE, LLP
ipr.theo.foster@haynesboone.com
david.mccombs.ipr@haynesboone.com
gregory.huh.ipr@haynesboone.com

For PATENT OWNER:

Brett E. Cooper
BC LAW GROUP, P.C.
bcooper@bc-lawgroup.com
Lionra_Counsel@b-clg.com

Robert A. Auchter
AUCHTER PLLC
robert@auchterlaw.com