

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

CISCO SYSTEMS, INC.,
Petitioner,

v.

FINJAN, INC.,
Patent Owner.

Case IPR2018-00391
Patent 7,647,633 B2

Before THOMAS L. GIANNETTI, MIRIAM L. QUINN, and
PATRICK M. BOUCHER, *Administrative Patent Judges*.

QUINN, *Administrative Patent Judge*.

FINAL WRITTEN DECISION
35 U.S.C. § 318

I. INTRODUCTION

We instituted *inter partes* review pursuant to 35 U.S.C. § 314 to review claims 1–4, 8, and 11–14 of U.S. Patent No. 7,647,633 B2 (“the ’633 patent”), owned by Finjan, Inc. We have jurisdiction under 35 U.S.C. § 6. This Final Written Decision is entered pursuant to 35 U.S.C. § 318(a) and 37 C.F.R. § 42.73. For the reasons discussed below, Petitioner has shown by a preponderance of the evidence that claims 1–4, 8, and 11–13 of the ’633 patent are unpatentable. Petitioner, however, has not shown by a preponderance of the evidence that claim 14 of the ’633 patent is unpatentable.

II. BACKGROUND

A. RELATED MATTERS

The parties identify several district court cases (N.D. Cal.) involving the ’633 patent. Pet. 5; *see also* Paper 4. The ’633 patent also has been the subject of various proceedings at the USPTO, including *Palo Alto Networks, Inc. v. Finjan, Inc.*, Case IPR2015-01974 (PTAB), in which the Board issued a Final Written Decision concerning claims 14 and 19 of the ’633 patent. Paper 4; Ex. 2002. The ’633 patent also has been the subject of an *ex parte* reexamination (Control No. 90/013,016). Ex. 2001.

B. THE ’633 PATENT (EX. 1001)

The ’633 patent relates to a system and a method for protecting network-connectable devices from undesirable downloadable operations. Ex. 1001, 1:30–33. The patent describes that “[d]ownloadable information comprising program code can include distributable components (e.g.

JAVA™ applets and JavaScript scripts, ActiveX™ controls, Visual Basic, add-ins and/or others).” *Id.* at 1:60–63. Protecting against only some distributable components does not protect against application programs, Trojan horses, or zip or meta files, which are other types of “Downloadable information.” *Id.* at 1:63–2:2. The ’633 patent “enables more reliable protection.” *Id.* at 2:27–28. According to the Summary of the Invention,

In one aspect, embodiments of the invention provide for determining, within one or more network “servers” (e.g. firewalls, resources, gateways, email relays or other devices/processes that are capable of receiving-and-transferring a Downloadable) whether received information includes executable code (and is a “Downloadable”). Embodiments also provide for delivering static, configurable and/or extensible remotely operable protection policies to a Downloadable-destination, more typically as a sandboxed package including the mobile protection code, downloadable policies and one or more received Downloadables. Further client-based or remote protection code/policies can also be utilized in a distributed manner. Embodiments also provide for causing the mobile protection code to be executed within a Downloadable-destination in a manner that enables various Downloadable operations to be detected, intercepted or further responded to via protection operations. Additional server/information-destination device security or other protection is also enabled, among still further aspects.

Id. at 2:39–57.

C. ILLUSTRATIVE CLAIMS

Challenged claims 1, 8, 13, and 14 of the ’633 patent are independent. Illustrative claims 1 and 14 are reproduced below.

1. A computer processor-based method, comprising:
receiving, by a computer, downloadable-information;
determining, by the computer, whether the

downloadable-information includes executable code; and
based upon the determination, transmitting from the
computer mobile protection code to at least one information-
destination of the downloadable-information, if the
downloadable-information is determined to include executable
code.

14. A computer program product, comprising a
computer usable medium having a computer readable program
code therein, the computer readable program code adapted to be
executed for computer security, the method comprising:
providing a system, wherein the system comprises
distinct software modules, and wherein the distinct software
modules comprise an information re-communicator and a
mobile code executor;
receiving, at the information re-communicator,
downloadable-information including executable code; and
causing mobile protection code to be executed by the
mobile code executor at a downloadable-information
destination such that one or more operations of the executable
code at the destination, if attempted, will be processed by the
mobile protection code.

Id. at 20:54–62, 21:58–22:5. We refer to the three steps of claim 1
as the “receiving step,” the “determining step,” and the “transmitting
step,” respectively.

D. PROCEDURAL HISTORY

Petitioner, Cisco Systems, Inc., filed a Petition for *inter partes* review
challenging claims 1–4, 8, and 11–14 of the ’633 patent. Paper 1 (“Pet.”).
Patent Owner, Finjan, Inc., filed a Preliminary Response. Paper 6 (“Prelim.
Resp.”). On June 5, 2018, we determined that Petitioner had shown a
reasonable likelihood of prevailing on its unpatentability challenge as at
least one claim, and we instituted trial. Paper 7 (“Dec. on Inst.”).

During trial, Patent Owner filed a Patent Owner Response (Paper 12 (“PO Resp.”)) and Petitioner filed a Reply (Paper 16 (“Reply”)). Patent Owner requested authorization to file a Sur-reply, which we granted. Paper 18; Paper 22 (“Sur-reply”). Both parties filed Motions to Exclude and corresponding responsive papers. Papers 19, 23, 24, 27–29. We heard oral argument on March 6, 2019, a transcript of which is filed in the record. Paper 32 (“Tr.”).

E. EVIDENCE OF RECORD

Petitioner relies upon the following references as evidence of prior art:

- 1) *Hanson*: PCT Published Application WO 98/31124, published on July 16, 1998 (Exhibit 1004);
- 2) *Hyppönen*: U.S. Patent No. 6,577,920 B1, issued on June 10, 2003 (Exhibit 1005); and
- 3) *Touboul*: PCT Published Application WO 98/21683 (Exhibit 1007).

In addition, Petitioner supports its contentions with the Declaration of Paul Clark, Ph.D. Ex. 1003 (“Clark Declaration”). With its Patent Owner Response, Patent Owner provides a Declaration of Nenad Medvidovic, Ph. D. Ex. 2008 (“Medvidovic Declaration”).

F. GROUNDS OF UNPATENTABILITY

The following grounds of unpatentability are at issue (Pet. 32):

Claim(s)	Basis	References
1–4, 8, 11, 13, 14	§ 103	Hanson and Hyppönen
12	§ 103	Hanson, Hyppönen, and Touboul

III. ANALYSIS

A. CLAIM CONSTRUCTION

The Board interprets claim terms of an unexpired patent using the “broadest reasonable construction in light of the specification of the patent.” 37 C.F.R. § 42.100(b) (2018);¹ *see Cuozzo Speed Techs., LLC v. Lee*, 136 S. Ct. 2131, 2144–46 (2016). We presume a claim term carries its “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. *In re Translogic Tech., Inc.*, 504 F.3d 1249, 1257 (Fed. Cir. 2007) (citation omitted).

With regard to terms drafted in means-plus-function language, “[a]pplication of § 112, ¶ 6 requires identification of the structure in the specification which performs the recited function.” *Micro Chemical, Inc., v. Great Plains Chemical Co., Inc.*, 194 F.3d 1250, 1257 (Fed. Cir. 1999); *see also* 37 C.F.R. § 42.104(b)(3). Further, the statute does not permit “incorporation of structure from the written description beyond that necessary to perform the claimed function.” *Id.* at 1258.

1. Means-Plus-Function Terms

Claim 13 recites limitations written in means-plus-function format. In our Decision on Institution, we reviewed the parties’ proposed constructions for these terms. Dec. on Inst. 5–9; Pet. 20–22; Prelim. Resp. 11–13. We

¹ A recent amendment to this rule does not apply here because the Petition was filed before November 13, 2018. *See* Changes to the Claim Construction Standard for Interpreting Claims in Trial Proceedings Before the Patent Trial and Appeal Board, 83 Fed. Reg. 51,340 (Oct. 11, 2018) (amending 37 C.F.R. § 42.100(b) effective November 13, 2018).

preliminarily determined for each term the structure corresponding to the recited function as follows:

Term	Board's Preliminary Claim Construction (Structure)
means for receiving downloadable-information	Re-communicating device, such as a server or firewall
means for determining whether the downloadable-information includes executable code	Protection engine (Fig 3) in a re-communicating device, such as a server or firewall; or Detection engine (Fig. 4) within a protection engine in a re-communicating device, such as a server or firewall
means for causing mobile code to be communicated to at least one information-destination of the downloadable-information, if the downloadable-information is determined to include executable code	Protection engine (Fig 3) in a re-communicating device, such as a server or firewall; or Transfer engine (Fig. 4) within the protection engine in a re-communicating device, such as a server or firewall

Dec. on Inst. 5–9. Neither party raises any disputes concerning the proper scope of these terms. Accordingly, we adopt our preliminary claim construction as indicated above and for the reasons stated in our Decision on Institution. *Id.*

2. Prior Board Claim Constructions

In our Decision on Institution, we noted that the Board has previously had occasion to analyze two claim terms recited in the '633 patent. First, we noted that a panel of the Board issued a Decision on Appeal in connection with the reexamination (Control No. 90/013,016) of the '633 patent.

Ex. 2001. Regarding the “determining whether” phrase, the Decision on

Appeal states: “In order to disclose determining whether the downloadable-information includes executable code, [the prior art] must disclose distinguishing between two alternative possibilities: executable code is included in the downloadable-information, and executable code is not included in the downloadable-information.” *Id.* at 5.² Petitioner does not address this prior Board Decision in its Petition. Patent Owner does not further address the construction of this term in its Response. Petitioner, in its Reply and during oral argument, asserts that “the claim term should be given its plain and ordinary meaning,” but otherwise does not propose any particular construction for the term. Reply 6; Tr. 17:13–20:22 (arguing also that the Board does not need to construe the term because the prior art shows the limitation under any reasonable construction of the term). Because resolution of the scope of the phrase “determining whether” is not necessary to our determination, we do not discuss this term further. *See Nidec Motor Corp. v. Zhongshan Broad Ocean Motor Co.*, 868 F.3d 1013, 1017 (Fed. Cir. 2017); *Vivid Techs., Inc. v. Am. Sci. & Eng’g, Inc.*, 200 F.3d 795, 803 (Fed. Cir. 1999) (“[O]nly those terms need be construed that are in controversy, and only to the extent necessary to resolve the controversy.”)

Second, our Decision on Institution noted that we issued a Final Written Decision in IPR2015-01974 (Paper 49, March 15, 2017) in which we construed “executable code” in the phrase recited in claims 14 and 19 of the ’633 patent, in the context of the surrounding claim language and the Specification, to mean that the “the executable code whose operations are processed by the mobile protection code at the destination is the same as the executable code received, i.e., *it undergoes no modification.*” Ex. 2002, 18

² See Ex. 1001, 9:52–57, Fig. 3.

(emphasis added). Petitioner mentions the “no modification” requirement while contending that the prior art meets the claim language, but does not argue the meaning of the claim language further. Pet. 57 n.5. Neither party urges that we revisit the construction of the “executable code” phrase of claim 14 (“causing mobile protection code to be executed . . . at the downloadable-information destination such that one or more operations of the executable code at the destination, if attempted, will be processed by mobile protection code”). Accordingly, we adopt our prior determination here.

3. “*information re-communicator*” and “*information monitor*”

Patent Owner proposes construction for these terms as follows (PO Resp. 11, 14):

Term	Patent Owner’s Proposal
Information re-communicator	A computing device that receives downloadable-information <i>from an external network</i> and then sends it on to its destination
Information monitor	A component of an information re-communicator that monitors downloadable-information from <i>an external network</i>

The particular dispute concerning these terms focuses on whether the downloadable-information must be received from *an external network*.

Asserting the terms so require, Patent Owner argues that Hanson does not disclose the limitation because Petitioner relies on “downloadable-information” that is received from an internal company network. PO Resp. 25–26 (relying on Medvidovic Decl. ¶ 71). Petitioner responds that the recited re-communicator is a server. Reply 4 (citing Ex. 1001, 2:60–63).

Petitioner also argues that the Specification describes an embodiment in which a “user-device” could be configured as a firewall or server and therefore would receive information from an internal network. *Id.* (citing Ex. 1001, 7:43–62).

We agree with Petitioner that the “information re-communicator” is not limited to receiving downloadable-information from an external network. First, the claims are silent as to the source of the “downloadable-information.” Therefore, from the plain meaning of the claim language, there is no requirement that the “re-communicator” receive downloadable-information from an external network. Second, the Specification describes consistently a server or a firewall as a “re-communicator.” These descriptions are general, and do not restrict the server or firewall to processing the downloadable-information solely to that coming from an “external network.” For instance, the Abstract states that a “protection engine embodiment provides, within a server, firewall or other suitable ‘re-communicator,’ for monitoring information received by the communicator.” Ex. 1001, Abst. As another example, Figure 9 depicts, at step 901, “Monitor re-communicator (e.g., server) operation.” *Id.* at Fig. 9. And significantly, in the Summary of the Invention, the Specification identifies a “server” as synonymous with “re-communicator” without limitation: “one or more network servers, firewalls or other network connectable information re-communicating devices (as are referred to herein summarily [as] one or more ‘servers’ or ‘re-communicators’).” *Id.* at 2:58–62; *see also id.* at 5:34–36 (“Embodiments provide, within one or more ‘servers’ (e.g., firewalls, resources, gateways, email relays or other information re-communicating devices)”), 7:2–5 (“system 104a . . . can include a

protection-initiating host ‘server’ or ‘re-communicator’ (e.g., ISP server 140a), 7:50–52 (“(i.e., as a ‘re-communicator’ or ‘server’”), 18:33–35 (“a protection engine monitors the receipt, by a server or other re-communicator of information, and receives such information intended for a protected information-destination (i.e., a potential Downloadable)”).

The broad description of the “re-communicating” devices as, for example, servers, leads us to the conclusion that the “information re-communicators” of claims 2, 9, 10, and 14 are generally devices or processes that “re-communicate” the received information, much like a server, firewall, gateway, or other device that transfers the received information. *See, e.g., id.* at 7:8–11 (“IPS server 140a includes one or more email, Internet or other servers 141a, or other devices or processes capable of transferring or otherwise ‘re-communicating’ downloadable information to user devices 145.”). Thus, we conclude that the term “information re-communicator” is not limited to receiving “downloadable-information” from an external network. Because the “information monitor” term (claim 8), likewise, is recited as “receiving downloadable-information by a computer” without requiring any specific network source, this term also is not limited to an external network.

4. “downloadable-information”

Patent Owner contends that the term “downloadable-information” means “information which is downloaded from a source computer which may or may not include executable code.” PO Resp. 10. Patent Owner relies on the Medvidovic Declaration in support of its construction. *Id.* (citing Medvidovic Decl. ¶ 45). Patent Owner argues that Dr. Clark

provided support for its contention during cross-examination by giving the following testimony:

Q. So how do you interpret the term “downloadable information”?

A. As it says in paragraph 32 and as we discussed earlier, it explains that ***downloadable information includes data that can be downloaded and that may or may not include executable code.***

Q. What does the term “download” mean?

A. Generally to retrieve something from a server.

Id. (quoting Ex. 1027, 44:20–45:6) (emphasis by Patent Owner). The dispute centers on whether the downloadable-information is limited only to data that is capable of being downloaded. *See* PO Resp. 23 (arguing that Hanson’s “request for server resources” are not “data that can be ***downloaded*** and that may or may not include executable code”).

Petitioner, in response, focuses on whether the claim term requires a source computer. Reply 3. But, more importantly, Petitioner argues that we need not construe the term because the asserted prior art teaches “downloadable-information” under any reasonable interpretation. *Id.* at 4. We agree with Petitioner that we need not construe the term because, as discussed below, Petitioner has shown that prior art teaches the limitation, even under Patent Owner’s interpretation.

B. PRINCIPLES OF LAW AND LEVEL OF ORDINARY SKILL IN THE ART

A claim is unpatentable under 35 U.S.C. § 103(a) if the differences between the claimed subject matter 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. *KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 406 (2007). The question of obviousness is resolved on the basis of underlying factual determinations including (1) the scope and content of the prior art; (2) any differences between the claimed subject matter and the prior art; (3) the level of skill in the art; and, (4) where in evidence, so-called secondary considerations, including commercial success, long-felt but unsolved needs, failure of others, and unexpected results. *Graham v. John Deere Co.*, 383 U.S. 1, 17–18 (1966) (“the *Graham* factors”).

As to the level of ordinary skill in the art, Petitioner provides testimony from Dr. Clark that a person of ordinary skill in the art would have had a working knowledge of downloading information from the World Wide Web, the vulnerability of downloadable-information to contain malicious operations, and methods for preventing the execution of malicious operations. Clark Decl. ¶ 21. Dr. Clark opines that a person of ordinary skill in the art would have gained the requisite knowledge and experience through education, such as a Bachelor’s degree in computer science, computer programming, electrical engineering, and four years of experience in programming, or by obtaining a Master’s degree in electrical engineering, but having only one to two years of programming experience. *Id.* Dr. Clark also opines that a person of ordinary skill in the art may have had no formal education, but may have gained the requisite level of knowledge through eight years of computer programming experience. *Id.* Patent Owner’s expert, Dr. Medvidovic, acknowledges the opinion of Dr. Clark and states his opinions “would be the same if rendered from the perspective of a person of ordinary skill in the art set out by Dr. Clark.” Medvidovic Decl. ¶ 38.

Accordingly, we adopt the level of ordinary skill in the art articulated by Dr. Clark.

C. OVERVIEW OF HANSON AND HYPPÖNEN

For all independent claims, Petitioner relies on Hanson or Hanson and Hyppönen as teaching all the recited limitations. An overview of these references follows.

1. *Hanson*

Hanson is an international application published under the Patent Cooperation Treaty as WO 98/31124 (published July 16, 1998). Ex. 1004, (11), (43). Hanson is directed to client/server computer communication over an internetwork system and to improved access of firewall protected servers. *Id.* at 1, 3–5. Hanson notes the problem of growing Internet use with respect to security, particularly for “internal company computers being compromised by an external entity.” *Id.* at 2:5–9. Hanson describes that security of internal computers is provided by a firewall, which is a filter that blocks communication in both directions: “input to the internal network of the company and output to the Internet.” *Id.* at 2:9–14. Hanson also describes further security provided by a proxy server, which behaves as a relay between an internal network and the Internet for communication requests initiated inside the company’s network. *Id.* at 2:15–17. Proxy servers trust all internal computers and assume that communication from an internal computer (to the proxy server) will not compromise security of other internal computers. *Id.* at 2:18–20. Hanson, however, recognizes that communications received from computers on the Internet at large are a

security threat, especially for a company that selectively wants to share internal company resources on the Internet. *Id.* at 2:20–24.

Hanson aims to provide secure two-way data communication over the Internet without having to open the firewall or installing a new network open to the Internet. *Id.* at 2:32–3:6. Figure 2, reproduced below, illustrates the components of Hanson's system.

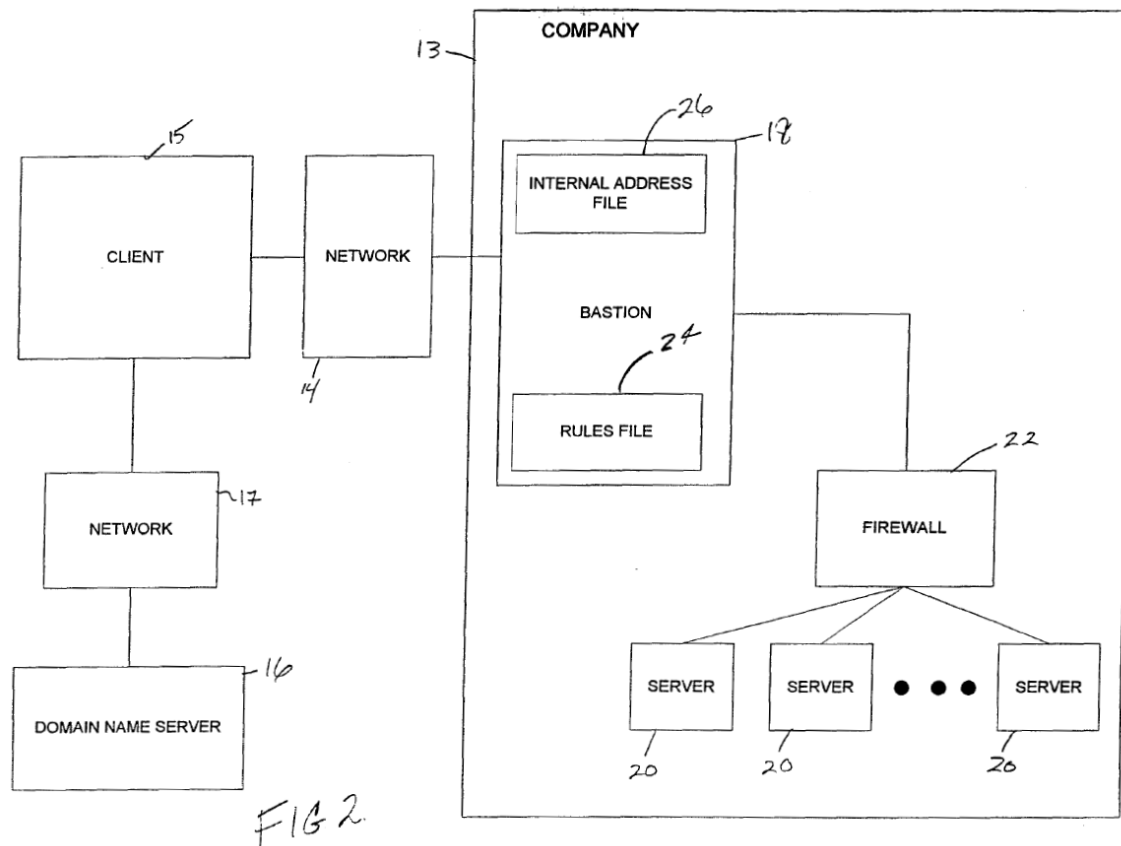


Figure 2 depicts a network system for secured communication between a requester or client 15 and servers 20, protected by firewall 22. *Id.* at 4:16–18. Figure 2 also illustrates bastion server 18, which includes internal address file 26 and rules file 24 stored in memory. Hanson describes that when client 15 sends a data packet identifying any server 20, behind firewall 22, as the destination, the data packet is delivered first to

bastion server 18. *Id.* at 5:2–16. The address of server 20 is resolved at bastion server 18 because internal address file 26 contains the server names of servers 20. *Id.* at 6:9–10. If an internal address matching the desired server is found in internal address file 26, the received packet “is checked against rules contained within the rules file.” *Id.* at 6, 16–17. Rules file 24 “provides a predetermined set of rules for maintaining secured communication of data packets passing in both directions through bastion server 18.” *Id.* at 5:8–10. “The rule checks include certain security[] programs that operate upon received data packets and, particularly data packets that are or include programs.” *Id.* at 6:22–23. In particular, Hanson describes the rules with respect to Table 3, reproduced below.

Destination IP: 204.95.95.0-94
FILE: viruses.dat
FILE: ps_error.dat
Flags: 0110001101
URL: fujil.hde.com/target.html
 JAVA Checks:
signature.class
security.class
test.class
 ActiveX Checks:
security.ocx

Table 3 - rules file.

Table 3 illustrates an example set of rules for checking data packets passing through bastion server 18. *Id.* at 9:2–3. Hanson explains that the bastion server checks data packets against known viruses and data errors by checking against the files “viruses.dat” and “ps_error.dat.” *Id.* at 10–14.

The rules file also includes “JAVA Checks” and “Active X Checks.” *Id.* at Table 3, 10:8–8, 11:3–6. Hanson describes the “JAVA Checks” code line of Table 3 as indicating the JAVA class files that execute “if the bastion receives a JAVA applet as or in a data packet.” *Id.* at 10:8–9. In particular, Hanson describes that each of the class files indicated performs “specific checks of received JAVA applets.” *Id.* at 10:9–10. For the “security.class” program, Hanson states that this program “performs security operations similar to that performed by a complete, secure JAVA virtual machine.” *Id.* at 13–14. The program performs “protective illegal operation overrides by attaching itself to the applet being sent in the data packet.” *Id.* at 10:14–16. “When the applet intended for the recipient is run at the destination client or server, ‘security.class’ is run simultaneously.” *Id.* at 10:16–17. With regard to the “Active X Checks” code line of Table 3, Hanson explains that “security.ocx,” similar to the description of JAVA Checks, “attaches to an Active X program destined for a client or server.” *Id.* at 11:6–7. “This program is run at the destination client or server and behaves similar to a common virus checking program.” *Id.* at 11:7–8. It monitors the execution of the Active X program that is running on the destination client or server so that if the Active X program attempts a destructive operation, the “security.ocx” program stops execution and warns the client or server user that the Active X program attempted an illegal operation. *Id.* at 11:8–12.

2. Hyppönen

Hyppönen, titled “Computer Virus Screening,” is directed to screening of computer data for viruses, and more particularly, macro viruses. Ex. 1005, [54], 1:6–8. Hyppönen describes the evolution of computer viruses from the 1980s and mid-1990s as consisting of a piece of executable

code that attached itself to a bona fide computer program. *Id.* at 1:11–16. To detect computer programs infected by viruses, a computer software “in general relied as a first step upon identifying those data files which contain executable code, e.g., .exe, .com, .bat.” *Id.* at 1:26–29. The second step, once identified, was to search the files for signatures of known viruses. *Id.* at 1:29–33. With the growing use of Microsoft Office, new viruses emerged, particularly those that infect macro files. *Id.* at 1:41–56. Macro files are used, for example, in templates, to provide customized tool bars, and are embedded in files with the extension “.dot” or in Microsoft Office files, or “.doc” files. *Id.* at 1:46–53.

Hyppönen’s system “effectively block[s] the transfer and/or processing of files which contain a previously unidentified (either to the local user or to the software produced) macro virus.” *Id.* at 2:40–44. Hyppönen provides a file system driver that detects a file system event, which involves writing, reading or copying of a file.” *Id.* at 4:37–39, 5:23–27. The file system driver enables file system events to proceed normally, or prevents file system events and issues appropriate alert messages to the file system. *Id.* at 4:44–49. After detecting a file system event, Hyppönen determines if the file involved includes a macro by examining the file name extension (e.g., to identify “.dot” or “.doc” files) “and/or scanning the file for embedded macros.” *Id.* at 5:25–32. Hyppönen determines a signature for the found macro(s) and checks its database(s) to ascertain whether the macro has a known virus or is an unknown macro that warrants further scrutiny. *Id.* at 5:29–62. If the signature obtained corresponds to a known virus, the file system event is suspended. *Id.* at

5:33–39. If the signature obtained is of a known, benevolent macro, the file system is allowed to proceed. *Id.* at 5:43–49. If no virus is found, and the macro is unknown and unverifiable, Hyppönen suspends the event. *Id.* at 5:53–61.

Hyppönen explains that its file system driver may be arranged to screen files for viruses other than macro viruses. *Id.* at 6:6–7. Hyppönen further describes the file system event as receiving all file access traffic, not just that of a hard disk. *Id.* at 6:9–11. For example, the file system drive may also process floppy disk data transfers, network data transfer, and CDROM data transfers. *Id.* at 6:11–16.

D. ANALYSIS OF INDEPENDENT CLAIMS 1, 8, AND 13

The parties extensively focus their arguments on the determining step of claim 1. Claim 1 recites “determining, by the computer, whether the downloadable-information includes executable code.” Claim 8 recites a content inspection engine as performing the determining step recited in claim 1. And claim 13 recites a means for determining, where the function is the determining step recited in claim 1. Although not the first-recited limitation in the claims, we begin our analysis with the determining step, given the prominent role of this limitation in this proceeding. The Petition relies on Hanson alone or in combination with Hyppönen as disclosing the determining step. Pet. 48–54. We find that Petitioner did not prove that Hanson alone teaches the determining step. We find, however, that Petitioner has proven by a preponderance of the evidence that Hyppönen teaches the determining step and that it would have been obvious to combine

Hyppönen's teachings with Hanson. We discuss our reasons for these findings under the separate headings below.

1. Whether Hanson Alone Teaches the Determining Step

With regard to Hanson, alone, Petitioner states that two disclosures regarding the bastion server are relevant to the determining step. *Id.* at 48–49. The first disclosure pertains to Hanson's rule check. *Id.* According to Petitioner, the rule checks operate “on data packets that are or include programs.” *Id.* We are not persuaded that this teaches that Hanson determines whether the downloadable-information includes executable code. Hanson describes that the rule checks are performed on data packets that may include programs, but does not describe detecting programs in “data packets.” Moreover, the fact that a reply packet may pass the checks against send rules does not teach the determining step either. All data packets in Hanson pass through the bastion server and are checked against the rules file, regardless of whether they include programs.

The second disclosure of Hanson that Petitioner relies on is the JAVA Checks and Active X Checks in the rules file. *Id.* at 50–51. We are not persuaded that Hanson's disclosures of JAVA Checks and Active X Checks teach the determining step. These checks indicate that certain security files attach to a data packet *if* there are JAVA applets or Active X programs in the data packet. But Hanson says nothing about detecting applets, Active X components, or any other purported “executable code.” Petitioner's expert, Dr. Clark, opines that a person of ordinary skill in the art “would have understood that Hanson's disclosure of applying ‘JAVA Checks,’ and ‘Active X Checks’ or other security program includes a determination of whether the data packet is or includes a JAVA applet, Active X program or

other executable program.” Clark Decl. ¶ 107. This is a conclusory statement—there is no reasoning provided for why or how, if Hanson does not explicitly teach the required determination, a person of ordinary skill in the art would have understood that the determination was necessary. The claims expressly require a determining step. Without supporting reasoning for the assertion, Dr. Clark’s opinion attempts to supply a claim limitation that has not been shown to be *indisputably* within the common knowledge of those skilled in the art. *See K/S Himpp v. Hear-Wear Techs., LLC*, 751 F.3d 1362, 1365 (Fed. Cir. 2014).

In this regard, Petitioner’s argument is that because Hanson describes applying a JAVA Check, for example, it necessarily detects JAVA executable code. *See, e.g.*, Reply 19 (“Thus, Hanson’s disclosure that ‘[t]he ‘JAVA Checks:’ code lines indicates JAVA class files that execute if the bastion receives a JAVA applet as or in a data packet’ . . . is conditional upon a determination that the executable is or includes a JAVA applet, rather than an Active X control.”). We do not find this argument that we should supply an element missing from Hanson—without evidence that it was common knowledge—to be persuasive. We find, therefore, that Petitioner has not met its burden of proof that Hanson alone teaches the determining step. *See In re Magnum Oil Tools Int’l, Ltd.*, 829 F.3d 1364, 1381 (Fed. Cir. 2016) (holding that because petitioner “bears the burden of proof,” the Board “must base its decision on arguments that were advanced by [petitioner]”); 35 U.S.C. § 312(a)(3); 37 C.F.R. §§ 42.22(a)(2), 42.104(b)(4)–(5).

2. Whether Hyppönen Teaches the Determining Step

In the alternative, Petitioner relies on Hyppönen as disclosing the determining step. Pet. 52–54. In particular, Petitioner argues that Hyppönen discloses examining the file extension and/or scanning the files to determine if the file contains a macro, which is a small executable program. *Id.* at 53 (citing Ex. 1005, 1:45–47, 5:23–29, Clark Decl. ¶ 110). We agree with Petitioner that Hyppönen’s teaching of examining the file to determine a file type and scanning the file to determine *whether or not* the file contains a macro or executable program sufficiently discloses the limitation “determining whether the downloadable information includes executable code.” Pet. 42–44; Ex. 1005, 1:26–31, 1:45–47, 2:35–36, 6:23–30. Hyppönen expressly states, for example, that it scans a file “to determine whether or not the file contains a macro.” Ex. 1005, 6:24–25. Hyppönen also states that the macro is a small executable program. *Id.* at 1:45–47. Hyppönen describes in particular examining the file extension, such as “.dot” or “.doc,” to determine those files that are likely to include macros, and scanning the file for embedded macros. *Id.* at 5:23–30. Therefore, Hyppönen teaches determining whether the downloadable-information includes executable code.

3. Reasons to Combine

Having found that Hyppönen teaches the determining step, we turn to analyzing the reasons to combine Hyppönen’s teachings with Hanson. According to Petitioner, it would have been obvious to a person of ordinary skill in the art “to modify the teachings of Hanson with the teachings of Hyppönen to include the conventional step of determining whether the data packet contains an executable code.” *Id.* (citing Clark. Decl. ¶ 111).

Further, Petitioner states that “[a] POSA would have been motivated to determine whether the data packet (file) contains executable code (as taught by Hyppönen) in order to efficiently focus resources on the recognized threat posed by executable files, rather than performing Hanson’s additional checks on all data packets and further expedite the delivery of those data packets not containing programs to the intended recipient without the attendant delay associated with malicious code protection.” *Id.* at 53–54 (citing Clark Decl. ¶ 111). Petitioner also asserts a reasonable expectation of success “given that the determining step was routine practice in conventional virus detection and disinfection software.” *Id.* at 54 (citing Clark Decl. ¶ 111). The portions of the Petition quoted above repeat *verbatim* the Clark Declaration, which contains no further detail or analysis regarding the statements made in the Petition.

In our Decision on Institution, we stated that Petitioner’s evidence of obviousness presented no reasonable likelihood of prevailing with regard to the combination of Hanson with Hyppönen. Dec. on Inst. 16–18. We noted that the “factual inquiry” into the reasons for “combin[ing] references must be thorough and searching, and the need for specificity pervades.” *In re Nuvasive, Inc.*, 842 F.3d 1376, 1381–82 (Fed. Cir. 2016) (internal quotations and citations omitted). A determination of obviousness cannot be reached where the record lacks “explanation as to *how* or *why* the references would be combined to produce the claimed invention.” *Trivascular, Inc. v. Samuels*, 812 F.3d 1056, 1066 (Fed. Cir. 2016); *see Nuvasive*, 842 F.3d at 1382–85; *Magnum Oil*, 829 F.3d at 1380–81. We concluded that the Petition lacked specificity and persuasive factual support for the alleged reasons to combine.

After reviewing the full record as developed during the trial, however, we reach a different conclusion. *See TriVascular*, 812 F.3d at 1068 (“[T]he Board is not bound by any findings made in its Institution Decision. At that point, the Board is considering the matter preliminarily without the benefit of a full record. The Board is free to change its view of the merits after further development of the record, and *should do so* if convinced its initial inclinations were wrong.”). Although we still find unconvincing the “efficiency” reason to combine we discussed in our Decision on Institution, Petitioner has pointed to additional evidence and argument in the record that demonstrates persuasively that a person of ordinary skill in the art would have been motivated to combine the teachings of Hyppönen with Hanson. Our analysis of all the reasons to combine that Petitioner proffered follows.

i. Efficiency

Hyppönen is directed to screening for viruses in macros when an application issues file access requests to a local hard disk drive, such as requests for opening a template in Microsoft Word. Ex. 1005, 1:6–8, 1:46–48, 1: 50–53, 4:39–53. Hanson, in contrast, performs the rule checks at the *bastion server*, where all data packets flowing between internal server 20 and client 15 are checked for compliance with various rules as discussed above. *See* Ex. 1004 11:1–13:13, Fig. 3. In particular, we note that Hanson performs a virus check on all data packets. *Id.*; *see also id.* at 9:10–14. If the data packet has not been destroyed for containing a virus and the data packet is a JAVA applet “as or in a data packet,” the bastion server additionally “attaches” the security check program to the applet intended for a recipient. *Id.* at 10:9–17. Petitioner alleges that the modified system would efficiently focus resources on the recognized threat posed by

executable files, rather than performing Hanson's additional checks on all data packets and further expediting the delivery of those data packets not containing programs to the intended recipient without the attendant delay associated with malicious code protection. *See* Pet. 53–54. Thus in the combination Petitioner asserts, Hanson's bastion server allegedly would be modified to examine further whether the file has a specific extension associated with executable files (.exe, .com .dot, etc.) or scan the file for embedded executable programs, in addition to those that are JAVA applets or Active X programs.

We stated our preliminary view in our Decision on Institution that there is no efficiency gained by this proposed modification because Hanson checks all data packets and teaches performing the JAVA Check only *if* packets contain JAVA applets and performing Active X Checks only *if* packets contain Active X programs. Dec. on Inst. 17–18. Thus, we concluded, Hanson is already efficient in performing the additional “Checks” on packets containing JAVA applets and Active X programs. *Id.*

Petitioner argues that we based our preliminary determination on an overly restrictive reading of the Petition and the cited art. Reply 20. In particular, Petitioner recasts the Petition's “efficiency” rationale as specifically implicating a “more efficient use of resources.” *Id.* at 21–22. The efficient use of resources, according to Petitioner, is Hanson's ability to check for other types of executables, not just JAVA applets or Active X components, providing thus “a higher level of protection.” *Id.* at 22. As we noted in our Decision on Institution, the allegation of efficiency is not reasonable because, at a minimum, expanding the types of files that Hanson would scan and examine at the server would not be more efficient than what

Hanson provides. Dec. on Inst. 17–18. Petitioner responds that, notwithstanding the additional categories of packets that Hanson would detect, a person of ordinary skill in the art would have understood the result is a “more efficient use of resources because of the higher level of protection received.” *Id.*

This argument of a “higher level of protection” is not the efficiency rationale that the Petition and Dr. Clark originally presented. The original reason in the Petition was to efficiently focus resources on the recognized threat posed by executable files, rather than performing Hanson’s additional checks on all data packets and further expedite the delivery of those data packets not containing programs to the intended recipient without the attendant delay associated with malicious code protection. Pet. 53–54. There is no implication in the Petition that a “higher level of protection” would have been the source of the alleged efficient focus on executable files. The reason provided in the Petition, instead, focused on somehow reducing the amount of processing Hanson’s bastion server would perform by stating that focusing on executable files avoids performing additional checks on *all data packets* and *expedites* delivery of data packets that are not executable files. *See id.* Even if the Hanson/Hyppönen system were to offer “higher level of protection” causing Hanson’s bastion server to do processing of packets that Hanson, alone, does not do, the modified system still would have processed *all data packets* received at the server, regardless of whether they are executable files or not. Recall that Hanson performs the rule check on *all data packets* passing through the bastion server, including performing a virus check. *See, e.g.,* Ex. 1004, 9:1–19 (describing the set of rules for checking data packets passing through the bastion, including checking the

data packet against known viruses and destroying the data packet if it fails to pass the check). In short, Hanson's bastion server is not doing less or doing anything more efficiently, and the late-coming "higher level of protection" argument is therefore unpersuasive.

Further, we find that, to the extent Petitioner relies on the "higher level of protection" argument as a reason to combine, this is an argument not presented previously in the Petition. Under these circumstances, we do not to consider the argument that a "higher level of protection" is indicative of the efficient use of resources, presented newly in the Reply. *See* 37 CFR 42.23(b). Petitioner had an opportunity to present this argument in its petition, but chose not to. *See SAS Inst., Inc. v. Iancu*, 138 S. Ct. 1348, 1355 (2018) (noting "the petitioner is master of its complaint"); *id.* at 1358 (noting the statute "makes the petition the centerpiece of the proceeding both before and after institution"). As the Federal Circuit has stated,

It is of the utmost importance that petitioners in the IPR proceedings adhere to the requirement that the initial petition identify "with particularity" the "evidence that supports the grounds for the challenge to each claim." 35 U.S.C. § 312(a)(3). "All arguments for the relief requested in a motion must be made in the motion. A reply may only respond to arguments raised in the corresponding opposition or patent owner response." 37 C.F.R. § 42.23(b). Once the Board identifies new issues presented for the first time in reply, neither this court nor the Board must parse the reply brief to determine which, if any, parts of that brief are responsive and which are improper. As the Board noted, "it will not attempt to sort proper from improper portions of the reply." Office Patent Trial Practice Guide, 77 Fed. Reg. 48,756, 48,767 (Aug. 14, 2012).

Acceleration Bay, LLC v. Activision Blizzard Inc., 908 F.3d 765, 775 (Fed. Cir. 2018) (quoting *Intelligent Bio-Sys., Inc. v. Illumina Cambridge Ltd.*, 821 F.3d 1359, 1369 (Fed. Cir. 2016)).

Accordingly, we are not persuaded that the “efficiency” rationale presented in the Petition has merit, and Petitioner’s “higher level of protection” argument in the Reply is untimely.

ii. Type of Executable Determination

As to additional reasons to combine, Petitioner directs us to the Petition at pages 42–44. Reply 21. Petitioner contends that the motivation to combine Hanson with Hyppönen’s teachings is to determine exactly what type of executable is in the downloadable, using the techniques of file detection and scanning. *Id.* Further, Petitioner asserts that Hyppönen’s technique is well-known (as evidenced by the explanation of the state of the art) and that the combination “would result in ‘a more robust and flexible security solution for a network or downstream user.’” *Id.* (citing Pet. 43–44). We are persuaded by Petitioner’s arguments and find that a person of ordinary skill in the art would have been motivated to combine the teachings of Hanson and Hyppönen to determine what type of executable is in the downloadable.

First, Petitioner contends, and we agree, that Hyppönen teaches examining a file-name extension and scanning the file for executable programs. Pet. 43 (citing Ex. 1005, 5:27–29). Hyppönen, by detecting the executable programs in a file, is capable of performing further analysis to protect the destination from malicious code. *Id.* (citing Ex. 1005, 5:28–65). For instance, Hyppönen’s application of this known technique is to detect a macro program in a Word file (by examining the file name extension and/or

scanning the content of the Word file), and then performing a virus check on the detected file. *See* Ex. 1005, 5:23–52. Thus, Petitioner demonstrates that Hyppönen’s technique achieves the missing piece in Hanson, determining whether the “data packet” is or contains either a JAVA applet or an Active X program in order to attach the appropriate security program. As Petitioner posits, and we agree, not all of Hanson’s data packets are or include programs, and, therefore, a person of ordinary skill in the art would have found it beneficial first to determine that the received data packet is or includes executable code, and then attach the appropriate security program to only those detected data packets. Pet. 43 (citing Nielson Decl. ¶ 93).

Second, Petitioner contends, and we agree, that a person of ordinary skill in the art would have found it beneficial to tailor the protection code to the specific file that contains executable code. *Id.* (citing Nielson Decl. ¶ 93). In other words, Hanson states that it “attaches” a “security.class” program to JAVA applets and a “security.ocx” program to Active X packets. Ex. 1004, 10:8–28, 11:3–12. We agree with Petitioner that in order to identify which security program to attach, Hanson would benefit from Hyppönen’s technique of identifying whether the data packet is or includes a JAVA applet or an Active X program. Pet. 43. Petitioner also argues, and we agree, that Hanson teaches broadly protecting against programs, not just JAVA applets or Active X programs. *Id.* (citing Ex. 1004, page 13, claim 10). In particular, Hanson teaches extending the types of data that can be protected with the “attachment of security programs and the examination of data packets at the bastion server.” Ex. 1004, 11:13–16. The combination of teachings, according to Petitioner, results in Hanson’s bastion server utilizing the specific technique taught by Hyppönen of file type extension

and/or content scanning, to determine what type of executable program is present in the received data packets. Pet. 44 (citing Nielson Decl. ¶ 93). Petitioner finally contends, and we agree, that a person of ordinary skill in the art would have had a reasonable expectation of success. *Id.* (citing Nielson Decl. ¶ 94).

Patent Owner argues that in our Decision on Institution, we stated that Petitioner had failed to explain sufficiently its contention that it would have been obvious to modify “*a server to implement techniques known for detecting macros invoked by an application at the client computer.*” Sur-reply 25 (quoting Dec. on Inst. at 17). According to Patent Owner, Petitioner’s arguments in the Reply, discussed above, do not respond to the Board’s concern explained in the Decision on Institution, and further, rely on a mischaracterization of Hyppönen. *Id.* In particular, Petitioner argued in Reply that Hyppönen’s technique would operate in a server, such as Hanson’s server, because Hyppönen teaches implementing its software processes at a server that receives network data transfers. Reply 23.

We find that Petitioner’s characterization of Hyppönen in this regard is not supported by the reference. Hyppönen teaches its detection technique specifically performed at the client computer, not a server as Petitioner argues. The passage in Hyppönen that Petitioner relies on mentions “network data transfers,” but refers to one example of a request that may be intercepted at the client computer. Ex. 1005, 6:9–16. Petitioner’s unreasonable reading of Hyppönen is not fatal to its position that Hyppönen’s technique would be combinable with Hanson’s bastion server, however. Petitioner is not relying on Hyppönen for the teaching of how to intercept a file system access or a request for a file—Hanson’s bastion server

already performs this by intercepting the data packets flowing to and from the bastion server. The particular teaching of Hyppönen that Petitioner relies upon is the examination of a file to determine what type of executable code (such as a macro) may be contained therein, if at all. *See* Ex. 1005, 2:35–36 (“scanning said file to determine whether or not the file contains a macro”); Reply 7 (arguing that Hyppönen discloses the step of scanning the file to determine whether or not the file contains a macro). And Hyppönen teaches the detection of file types in a broader scheme, as a well-known first step to virus protection. *See* Reply 22 (citing Ex. 1005, 1:5–6).

In sum, we find that Petitioner has persuasively demonstrated by a preponderance of the evidence that a person of ordinary skill in the art would have had reason to combine the teachings of Hyppönen with Hanson for the determining step.

4. Whether Hanson Teaches the Receiving and Transmitting Steps

Claim 1 recites further limitations, such as “receiving, by a computer, downloadable-information” and “transmitting from the computer mobile protection code to at least one information-destination of the downloadable-information if the downloadable information is determined to include executable code.” Claims 8 and 13 include similarly recited limitations. Upon review of the record, we are persuaded by Petitioner’s arguments and evidence that Hanson discloses these further limitations of claim 1 as shown in the Petition at pages 45–48 (receiving step), and 54–58 (transmitting step). For claim 13, the corresponding analysis we rely upon appears in the Petition at pages 70–71 (means for receiving, claim 13), and 72–73 (means for causing, claim 13). For instance, we find that Hanson’s bastion server (“means for receiving”) receives data packets, which are or include

programs. Ex. 1004, 8:22–23, 10:20–26, Fig. 7; Clark Decl. ¶¶ 96–98. And Hanson’s bastion server is configured with program instructions (“means for causing”) to transmit to a destination computer a “security.class” program if the received data packets include a JAVA applet, and “security.ocx” program if the received data packets include an Active X program, thereby disclosing the transmitting step. Ex. 1004, 12:8–18, 13:2–12; Clark Decl. ¶¶ 115–117. Because Patent Owner argues that Hanson does not teach these limitations based on claim construction issues, we discuss our further findings below.

5. Patent Owner’s Arguments Regarding Downloadable-Information

Patent Owner argues that Hanson’s bastion server does not receive “downloadable-information” because a “request for server resources are not Downloadable,” i.e., requests are not data that can be downloaded. PO Resp. 23. Petitioner responds that Hanson’s bastion server receives data packets from the server that may include executable programs. Reply 9–10. We agree with Petitioner.

Hanson describes that a server sends data packets as a reply to the request received from the client. Ex. 1004, 8:20–23. These data packets must also pass the check against a set of rules for outgoing data packets. *Id.* at 8:24–25. We thus find that Hanson describes the outgoing reply data packets in the same vein as describing the incoming data packet, and, therefore, the data packets Petitioner has identified are downloadable-information as they are sent in response to a request for a client, such as by transmitting a JAVA applet or Active X program requested by the client. As Petitioner argues, and we agree, JAVA applets and Active X programs are “downloadable-information,” as those are examples of downloadable

information identified in the '633 patent. Reply 9–10 (citing Ex. 1004, 12–13; Ex. 1001, 2:28–33).

Patent Owner characterizes the reply data packets as a request for server resources. PO Resp. 23. But Hanson does not limit its description of the outgoing reply data packets in that manner, and, instead, places emphasis on satisfying “the need for secure two-way data communication over the Internet.” Ex. 1004, 3:6; *see also* Ex. 1004, 6:26–27 (stating that Figure 7, which describes the reply packet flow from the server to the client illustrates “an example of two-way packet communication between a client and a firewall protected server over the Internet.”). Therefore, we are persuaded that the Petition demonstrates Hanson’s bastion server receives “downloadable-information” by a computer.

We determine, therefore, that Petitioner has shown by a preponderance of the evidence that Hanson and Hyppönen teaches all the limitations of independent claims 1 and 13.

6. Additional Limitations of Claim 8

In addition to the limitations of claim 8 discussed above, claim 8 recites “an information monitor for receiving downloadable-information by a computer.” Patent Owner argues that Hanson does not teach the information monitor because the bastion server receives information from an internal network, rather than an external network. PO Resp. 26. As discussed in Section III.A.3, *supra*, we are not persuaded by Patent Owner’s arguments that the recited re-communicator must receive the downloadable-information from an external network. Instead, we find that Petitioner has shown sufficiently that Hanson’s bastion server is an information monitor as recited. Pet. 64–65. Further, for the reasons stated above with respect to

claims 1 and 13, we are persuaded that Petitioner has shown by a preponderance of the evidence that all the further recited limitations of claim 8 are taught by Hanson and Hyppönen.

E. ANALYSIS OF DEPENDENT CLAIMS 2–4, AND 11

Claim 2 depends from claim 1, and further recites “wherein the receiving includes monitoring received information of an information re-communicator.” Petitioner argues that Hanson’s bastion server “monitors the received data packets by checking them against a set of rules including checks for virus, programs, addresses, ports, and files.” Pet. 58 (citing Clark Decl. ¶ 120). Furthermore, Petitioner argues that Table 3 describes the rules checking process that constitutes the recited “monitoring.” *Id.* at 58 (citing Ex. 1004, 11:2–3). The “information re-communicator,” according to Petitioner, is taught by the bastion server, which “re-transmits to a client the received data packets that have successfully passed the rules.” *Id.* at 59 (citing Ex. 1004, 10:20–27, Fig. 7).

Patent Owner argues that Hanson’s bastion server does not receive information from an external network. PO Resp. 26. As discussed in Section III.A.3, *supra*, we are not persuaded by Patent Owner’s arguments that the recited re-communicator must received the downloadable-information from an external network. Instead, we agree with Petitioner that Hanson’s bastion server is the information re-communicator. We are also persuaded that by applying the rules check of Table 3 to the incoming data packet, Hanson discloses monitoring the information received at Hanson’s bastion server. Therefore, we conclude that Petitioner has demonstrated by a

preponderance of the evidence that the combination of Hanson and Hyppönen teaches the further recited limitation of claim 2.

Claim 3 depends from claim 1, and further recites “wherein the information re-communicator is a network server.” Petitioner shows that Hanson’s bastion server is a network server because it is part of a network and is within a network. Pet. 60 (citing Ex. 1004, 6:16–20, 33–35, Fig. 2. We agree with Petitioner’s argument and evidence showing that Hanson teaches the limitation recited in claim 3. Except for the “re-communicator” argument discussed above with respect to claim 2, Patent Owner offers no argument particular to this claim. We conclude that Petitioner has demonstrated by a preponderance of the evidence that the combination of Hanson and Hyppönen teaches the limitation further recited in claim 3.

Claim 4 depends from claim 1, and further recites “wherein the determining comprises analyzing the downloadable-information for an included type indicator indicating an executable file type.” Claim 11 depends from claim 8, and similarly recites “wherein the content inspection engine comprises a file type detector for determining whether the downloadable-information includes a file type indicator indicating an executable file type.” Petitioner relies on the teachings of Hyppönen as discussed above with respect to claim 1, where Hyppönen identifies that a file includes executable code by examining the file type, such as “.exe, .com, .bat” as well as “.dot” for detecting macros. Pet. 62–63 (citing Ex. 1005, 5:27–29; Clark Decl. ¶ 130); Pet. 70 (relying, for claim 11, on the arguments and evidence presented with respect to claim 4). We agree with Petitioner that Hyppönen, in view of our analysis of the combination of Hanson and Hyppönen for claim 1, teaches analyzing the data packets received at

Hanson's bastion server for an included file type, such as identifying a ".dot" file as indicating that the information in the data packet is a file that contains or is likely to contain a macro. We conclude that Petitioner has demonstrated by a preponderance of the evidence that the combination of Hanson and Hyppönen teaches the limitation further recited in claims 4 and 11.

F. ANALYSIS OF DEPENDENT CLAIM 12

Petitioner contends that claim 12, which depends from claim 8, would have been obvious over the combination of Hanson, Hyppönen, and Touboul. Pet. 82–84. Claim 12 recites "wherein the content inspection engine comprises a parser for parsing the downloadable-information and a content analyzer communicatively coupled to the parser for determining whether one or more downloadable-information elements of the downloadable-information correspond with executable code elements." Petitioner relies on Touboul's disclosure of a code scanner that uses "conventional parsing techniques to decompose the code . . . of the Downloadable." *Id.* at 83 (citing Ex. 1007, 11:14–16). Touboul also discloses performing a full-content inspection. *Id.* (citing Ex. 1007, 12:12–14). The results of the code scanner, according to Touboul, are communicated with a logical engine to determine whether to allow or block the Downloadable. Ex. 1007, 4:1–3. Petitioner argues that these teachings in Touboul show conventional methods for determining the content of data packets. Pet. 84. And a person of ordinary skill in the art would have been motivated to use such a conventional technique as it would allow Hanson's bastion server to "efficiently determine what type of executable code is

present in the data packet in order to apply a tailored security measure.” *Id.* (citing Clark. Decl. ¶ 180). We agree with Petitioner that it would have been advantageous to apply Touboul’s technique to the modified Hanson bastion server in order to determine what type of executable code is present in the data packet. As discussed above with respect to the combination of Hanson with Hyppönen’s teachings, such a determination would allow Hanson’s bastion server to attach, for example, the “security.class” program to a JAVA applet, after detecting that the data packets are or include a JAVA applet using the conventional detection techniques of scanning the data packet’s content. Petitioner also argues, and we agree, that a person of ordinary skill in the art would have been confident that there was a reasonable expectation of success given the vast and long use of conventional parsing techniques and subsequent analysis of the parsed code in computer security methods and systems. Pet. 84 (citing Clark Decl. ¶ 181).

We conclude that Petitioner has shown by a preponderance of the evidence that the combination of Hanson, Hyppönen, and Touboul teaches the further recited limitations of claim 12.

G. ANALYSIS OF INDEPENDENT CLAIM 14

In our Decision on Institution we stated that Petitioner had demonstrated, under the reasonable likelihood threshold, that claim 14 would have been obvious over Hanson. Dec. on Inst. 14. In particular, we noted, with respect to the “executable code” limitation, that Dr. Clark explains attachment or tagging as not modifying the data packets, which include the alleged executable code, because the security program is a separate object. Ex. 1003 ¶ 118 (cited in Pet. 57 n.5). We noted that despite

the lack of explanation of how the tagging or attachment would be performed according to the state of the art and knowledge of a person of ordinary skill in the art, the sworn testimony by Dr. Clark was sufficient, at that time, to demonstrate that the “executable code” is not modified. Dec. on Inst. 14. We also stated that we were satisfied, for purposes of institution, that Dr. Clark’s testimony was consistent with the ordinary meaning of “tagging,” which at least suggests the security code and data packets are kept separate. *Id.* However, we specifically stated that our institution decision was preliminary and noted that the factual determination on this issue would be made under the preponderance of the evidence at the conclusion of all briefing and on a full record. *Id.*

We now have a full record and the parties’ briefing on this issue. Based on the full record before us, we determine that Petitioner has not shown by a preponderance of the evidence that the “executable code” is not modified.

Claim 14 recites “causing mobile protection code to be executed by the mobile code executor at a downloadable-information destination such that one or more operations of the executable code at the destination, if attempted, will be processed by the mobile protection code.” Ex. 1001, 22:1–5. As stated in Section III.A.2, we have construed previously that claim phrase with regard to “executable code” as follows: the executable code whose operations are processed by the mobile protection code at the destination is the same as the executable code received, i.e., it undergoes no modification. Ex. 2002, 18.

Patent Owner argues that Dr. Clark’s testimony of attaching or tagging a security program to a JAVA applet is devoid of any explanation of

what it means for a security program to be “attached.” PO Resp. 27. That testimony of Dr. Clark, according to Patent Owner is conclusory and lacks factual support, and, therefore Patent Owner urges that we give it little to no weight. *Id.* at 26–27. But during cross-examination, Dr. Clark explained that tagging means “adding a label of some kind or attaching” and that the security program “either precedes or succeeds the data packet in some way.” Ex. 1027, 66:2–74:7. Patent Owner contends that even though Dr. Clark attempts to provide new testimony on the issue, the explanations provided are still deficient because Hanson does not explain how to perform attachment—more importantly, Hanson is silent as to whether the applet is modified in any way by the disclosed “tagging” of a security program. PO Resp. 28. In short, Patent Owner contends that Hanson’s use of the words “tagging” or “attaching” are, by themselves, insufficient to show how Hanson transmits the applet with the security program. And Dr. Clark’s testimony, according to Patent Owner is too vague in the details of how “tagging” would allegedly be performed without modifying the applet, and, therefore, the testimony is insufficient to fill the evidentiary gap.

Petitioner, in response, argues that “Hanson is clear that the executable code in the received downloadable-information is the *same* executable code that is sent to the destination and run at the client (e.g., that ‘the’ downloadable is run ‘simultaneously’ with the security program), and so by definition it is not modified (or it would not be the same).” Reply 11. Petitioner’s argument relies on Hanson’s statement: “When the applet intended for the recipient is run at the destination client or server, ‘security.class’ is run simultaneously.” Ex. 1004, 10:16–17. Because Hanson describes that “the applet” is running, rather than a “modified

applet,” Hanson, according to Petitioner, must be recommunicating the received applet (read here “unmodified applet”) to the client. Reply 11–12 (also explaining that the same disclosure applies to Active X programs and “security.ocx”). Further, Petitioner posits that Hanson protects the computer running the JAVA applet or Active X program by overriding illegal operations or stopping the downloadable program, which are the same protective mechanisms the ’633 patent employs. *Id.* at 12. Thus, there is no need to modify the executable code, consistent with the ’633 patent. Finally, Petitioner points out that Hanson includes additional disclosure relevant to this issue. *Id.* at 13. In particular, Petitioner relies on claim 10 of Hanson, which states “tagging a security program onto the received data packet, if a specific program exists within the received data packet; and performing the security check program simultaneously with the specific program of the received data packet at said second server.” Ex. 1005, 15. Again, Hanson’s semantics of “performing the security program simultaneously with the specific program,” according to Petitioner, proves that a specific program received at the bastion server is the same as the specific program running at the destination computer.

Patent Owner rebuts Petitioner’s contentions with testimony of Dr. Medvidovic explaining that there are two methods to invoke Hanson’s security program. *See* Ex. 1008, generally. The first method involves modifying the applet to insert a call to the “security.class” program, so that the “security.class” program is invoked before any malicious instruction in the applet runs. *See* Medvidovic Decl. ¶¶ 74–76; Ex. 1008 at 68:9–25, 69:7–71:20. In this first method, the “security.class” program is separate from the applet, but the applet has been modified to include the call to the

“security.class” program. The second method is to include the “security.class” code as part of the applet, so the applet and security program run simultaneously. Ex. 1008 at 74:4–75:13. In this scenario, however, there is no “security.class” program that is distinct from the applet—it would be a class integrated in the applet code.

Petitioner, in response, appears to offer a third method, one in which the “security.class” program would be “associated” with the applet. *See* Reply 16 (citing Ex. 1027, 69:2–14 (testifying that “you’re adding [a tag] in some way to designate the data packet that contains the executable”). Dr. Clark testifies that “tagging” or “attaching” means “adding” to the received data packet, either preceding or succeeding the data packet, but according to Dr. Clark, Hanson is silent as to “how exactly you do that.” Ex. 1027, 70:6–71:3. Dr. Clark reasons that Hanson assumes the security program is “transported with[,] in some way[,] the original data packet.” *Id.* Dr. Clark also opines, in a Supplemental Clark Declaration, that Java Virtual Machine “classes instantiated by an applet[] that restricted the Applet’s operation without modifying the applet’s code before or even after the applet was received existed at the time of the ’633 filing.” Ex. 1024 ¶ 5 (citing Ex. 1029 for the proposition that “core or system classes (e.g. security.class) were loaded by JVM’s as needed”).

Our review of the testimony and the various arguments by the parties leads us to conclude that a person of ordinary skill in the art would have recognized that Hanson could perform “tagging” in at least three ways, as described above. No one technique is particularly identified or preferable, and the experts disagree as to how a person of ordinary skill in the art would have interpreted Hanson’s teaching. For one, Dr. Clark, espouses the view

of a general attachment, where the security program is transported somehow with the data packet, and somehow runs simultaneously with the applet at the destination. In contrast, Dr. Medvidovic provides technical details on how to implement a security program for a JAVA applet that is both transported with it and run simultaneously—but the applet must be modified to invoke the security program. Ex. 2008 ¶¶ 74–76. On balance, we conclude that the weight of the evidence establishes that the JAVA applet is modified in order for the security program to run simultaneously with the applet as discussed in Hanson. As Dr. Medvidovic testifies, the “security.class” code is either invoked from within the applet itself or embedded in the applet code as part of the classes of the applet. Dr. Clark explains only that the two are transported in some way by associating with a “tag,” which only explains how the two pieces of code would be transported and linked (*see* Ex. 1029 ¶ 6), but Dr. Clark does not explain sufficiently how the applet and the security program would run as described in Hanson. And we give little weight to Dr. Clark’s opinion that it was known to instantiate Java Virtual Machine classes without modifying the applet code before it is received at the client (*see* Ex. 1029 ¶ 5), because Hanson’s “security.class” program is in addition to the security operations running in a Java Virtual Machine (*see* Ex. 1004, 10:13–20). Dr. Clark’s opinion focuses on a general discussion of Java Virtual Machine, without tethering that explanation to how *Hanson’s* security program is attached to and would run simultaneously with the applet without modifying the applet.

In an *inter partes* review, Petitioner bears the burden of proving a proposition of unpatentability by a preponderance of the evidence. The burden of showing something by a preponderance of the evidence simply

requires the trier of fact to believe that the existence of a fact is more probable than its nonexistence before the trier of fact may find in favor of the party who carries the burden. *Concrete Pipe & Products of California, Inc. v. Construction Laborers Pension Trust for Southern California*, 508 U.S. 602, 622 (1993). Here, we find that Petitioner has not shown that it is more probable that Hanson’s JAVA applet, or any other executable code, is not modified at the bastion server. Instead, we find that the nonexistence of this fact is more probable—that Hanson’s JAVA applet is modified—and, therefore, we find that Petitioner has not carried its burden by a preponderance of the evidence.

Accordingly, we determine that Petitioner has not shown by a preponderance of the evidence that claim 14 would have been obvious over Hanson.

H. OBJECTIVE INDICIA OF NON-OBVIOUSNESS

We now turn to the analysis of Patent Owner’s evidence of indicia of non-obviousness, as it pertains to claims 1–4, 8, and 11–13. As a threshold issue, Patent Owner argues that Petitioner’s failure to address the objective indicia of non-obviousness in its Petition is “fatal to its obviousness arguments” because Petitioner had in its possession “substantial evidence related to secondary considerations of nonobviousness” due to the active litigation between the parties. PO Resp. 38. We are not persuaded by Patent Owner’s argument.

Patent Owner first introduces evidence of objective indicia of non-obviousness in its Response. Petitioner had no duty to address in the Petition arguments not yet made or evidence not yet introduced in the record by Patent Owner. We do not agree with Patent Owner that, because there

was discovery on objective indicia of nonobviousness in the parties' district court litigation, Petitioner's failure to address such indicia in the Petition is a "fatal flaw." *Cf. id.* at 37–38.

Further, Patent Owner argues that "Finjan's patented inventions have received much praise and commercial success that Petitioner fails to acknowledge." *Id.* at 39. We address each of these indicia in turn.

1. Commercial Success and Licensing

Patent Owner asserts that because its licensees "have product that are sold, which embody the inventions disclosed in the '633 patent, in the same security market as Petitioner, a presumption exists that the commercial success of Finjan's licensees products is due to the patented invention of the '633 patent." PO Resp. 44 (citing Harstein Decl. ¶¶ 5–12; Exs. 2020, 2021, 2022). This argument goes to whether there is a nexus between the alleged "commercial success" and the claimed invention of the '633 patent.

Petitioner argues that there is no "nexus" between the merits of the claimed invention and the evidence Patent Owner proffers regarding commercial success (or any other objective indicia of nonobviousness). Reply 24–25. In particular, Petitioner argues that there is no link between the license fees paid and the invention claimed in the '633 patent. *Id.* at 25.

The licenses address dozens other patents that Finjan owns in the computer security market, and there is no indication that any particular portion of the licenses is attributable to the '633 patent. Indeed, Petitioner points out Patent Owner's own argument and evidence that the licenses "were entered to avoid litigation" among the other characterized benefits alleged by Patent Owner. Reply 25 (citing PO Resp. 40–41, 44).

We agree with Petitioner that the arguments Patent Owner presents do not demonstrate a nexus between “commercial success” and the invention claimed in the ’633 patent. In particular, Patent Owner fails to produce evidence sufficient to show that its licensing program was successful because of the merits of the subject matter claimed in the ’633 patent, as opposed to, for example, other of the numerous patents in Patent Owner’s licensed portfolio, business decisions to avoid litigation, prior business relationship, or for other economic reasons.

To be accorded substantial weight, there must be a nexus between the claimed invention and the evidence of objective indicia of nonobviousness. *In re GPAC Inc.*, 57 F.3d 1573, 1580 (Fed. Cir. 1995). Nexus is a legally and factually sufficient connection between the objective evidence and the claimed invention, such that the objective evidence should be considered in determining nonobviousness. *Demaco Corp. v. F. von Langsdorff Licensing Ltd.*, 851 F.2d 1387, 1392 (Fed. Cir. 1998). Although “there is a presumption of nexus for objective considerations when the patentee shows that the asserted objective evidence is tied to a specific product and that product ‘is the invention disclosed and claimed in the patent’ (*WBIP, LLC v. Kohler Co.*, 829 F.3d 1317, 1339 (Fed. Cir. 2016)), Patent Owner carries the burden of coming forward with evidence that the “thing . . . that is commercially successful is the invention disclosed and claimed in the patent” (*Demaco*, 851 F.2d at 1392). Moreover, “[w]hen the thing that is commercially successful is not coextensive with the patented invention—for example, if the patented invention is only a component of a commercially successful machine or process—the patentee must show prima facie a legally sufficient relationship between that which is patented and that which

is sold.” *Id*; see *ClassCo, Inv. v. Apple, Inc.*, 838 F.3d 1214, 1220 (Fed. Cir. 2016) (“there is no nexus unless the evidence presented is reasonably commensurate with the scope of the claims” (internal quotation omitted)).

In this case, Patent Owner has not made such a showing. For instance, Patent Owner’s general allegations that the licensees sell products in the same market covered in Finjan’s patent is not indicative that any of the products sold indeed practice or embody the ’633 patent, or that the products are sold in any part because of the advantages conferred by the claimed invention in the ’633 patent. Further, we note, and are persuaded by, Petitioner’s argument that Patent Owner has presented the same licenses and similar arguments in another *inter partes* review covering an unrelated patent and different subject matter (IPR2015-01892), which casts doubt on the existence of a relationship between the proffered licenses and the alleged indicia of nonobviousness. In the absence of an established nexus with the claimed invention, Patent Owner’s purported evidence of commercial success and licensing is not entitled to much, if any, weight and generally have no bearing on the legal issue of obviousness. See *In re Vamco Mach. & Tool, Inc.*, 752 F.2d 1564, 1577 (Fed. Cir. 1985).

2. Industry Praise

Patent Owner argues that “the praise the ’633 patent has received in the industry further demonstrates that the ’633 Patent was not obvious.” PO Resp. 45. As evidence of this alleged “industry praise,” Patent Owner relies on the Harstein Declaration (Exhibit 2012) generally. Patent Owner, however, does not explain in any detail what industry praise it relies on and how any of the contents of the Harstein Declaration and various exhibits generally cited (Exs. 2020, 2021, 2022), specifically, disclose industry praise

that has nexus to the '633 patent. *See* 37 C.F.R. § 42.63104(b)(5) (“The Board may exclude or give no weight to the evidence where a party has failed to state its relevance or to identify specific portions of the evidence that support the challenge.”). The argument and evidence, as presented, fail to show any industry praise, and, therefore, this factor does not weigh in Patent Owner’s favor on the legal issue obviousness.

3. *Long-Felt Need and Recognition of a Problem*

Patent Owner additionally contends that “the '633 Patent satisfied a long-felt need in the industry and recognized a problem that was not solved.” PO Resp. 45 (citing Medvidovic Decl. ¶¶ 79–80). According to Patent Owner, “the '633 patent distinguishes itself over the prior art because the prior art modifies executable code, while the '633 Patent does not modify executable code, and a desire existed for practical systems that were not resource intensive.” *Id.* (citing also Ex. 1001, 1:58–2:6, 2:66–3:4, 4:11–16, 10:39–45). Patent Owner fails to present arguments and evidence sufficient to show a long-felt but unresolved need.

To establish evidence of a long-felt but unresolved need, a patent owner must come forward with evidence to show that there was a persistent problem that was recognized by those of ordinary skill in the art. *See In re Gershon*, 372 F.2d 535, 539 (CCPA 1967). The problem must not have been solved previously by another, and the claimed invention must, in fact, satisfy the long-felt need. *See Newell Cos., Inc. v. Kenney Mfg. Co.*, 864 F.2d 757, 768 (Fed. Cir. 1988). Patent Owner’s argument is a bare assertion that the '633 patent satisfied a long-felt need and recognized a problem that was not solved. Further, Patent Owner’s assertion is not commensurate with claims 1–4, 8, and 11–13. For instance, Patent Owner states that the '633

patent solves the desire for practical system that were not resource intensive because the prior art modifies executable code, while the '633 patent does not. PO Resp. 45. This “no-modification” feature may be part of claim 14, but the feature is not part of claims 1–4, 8, and 11–13, which are the challenged claims involved in our analysis of indicia of nonobviousness. Therefore, even if there were a long-felt need for a system that did not perform “modification” of executable code before sending it to the downloadable-information destination, as recited in claim 14, such a showing would not be germane to the other challenged claims. The argument and evidence, as presented, fail to show any long-felt need solved by the invention claimed in the pertinent '633 patent claims, and, therefore, this evidence is entitled to little weight in the legal issue of obviousness.

I. CONCLUSIONS OF OBVIOUSNESS OF CLAIMS 1–4, 8, AND 11–14

We have considered the secondary considerations of non-obviousness and accorded them appropriate weight along with our analysis of all of the other *Graham* factors, and conclude that Petitioner has shown obviousness of claims 1–4, 8, and 11–13. Accordingly, and for the foregoing reasons, we conclude that Petitioner has shown by a preponderance of the evidence that the subject matter of claims 1–4, 8, and 11–13 of the '633 patent would have been obvious to a person of ordinary skill in the art at the time of the invention, as asserted, and that those claims are, therefore, unpatentable.

We also conclude, however, that Petitioner has not shown by a preponderance of the evidence that claim 14 is unpatentable over Hanson.

J. MOTIONS TO EXCLUDE

Both parties filed motions to exclude. Neither motion is granted for the reasons that follow.

1. Patent Owner's Motion to Exclude

Patent Owner's Motion to Exclude (Paper 19, "PO Mot.") seeks exclusion of Exhibit 1029 and the Declaration of Dr. Paul Clark (Exhibit 1003).

i. Exhibit 1029

Patent Owner argues that Exhibit 1029 (an excerpt from a publication titled "Web Security & Commerce") is a belated submission, inadmissible under Fed. R. Evid. 403, and "should be excluded because it is improper and highly prejudicial for Petitioner to introduce new evidence in its Reply when it should have been included in its petition." PO Mot. 1–2. Petitioner argues that Exhibit 1029 is relied on by Dr. Clark in his rebuttal declaration and is submitted in response to arguments made in the Patent Owner response. Paper 27.

Exhibit 1029 is not cited in Petitioner's Reply. Rather, it is used as evidence that Dr. Clark cites as support for his opinions of Java Virtual Machine instantiation of a class, which we discussed above. Moreover, not all introduction of new evidence is prejudicial. "[T]he introduction of new evidence in the course of the trial is to be expected in *inter partes* review trial proceedings and, as long as the opposing party is given notice of the evidence and an opportunity to respond to it, the introduction of such evidence is perfectly permissible under the APA." *Genzyme Therapeutic Prods. LP v. Biomarin Pharm. Inc.*, 825 F.3d 1360, 1366 (Fed. Cir. 2016). Here, Patent Owner has not been prejudiced. Patent Owner had an

opportunity to address Exhibit 1029 in its sur-reply, but it chose not to. And furthermore, as Petitioner asserts, Petitioner's reliance on Exhibit 1029 appears to stem from the arguments presented by Patent Owner. We have addressed Dr. Clark's opinion that relies on Exhibit 1029 in our substantive discussion of claim 14, and given it the appropriate weight.

This request to exclude the exhibit is denied.

ii. Exhibit 1003

Patent Owner requests that we exclude the Clark Declaration because "his opinions are conclusory, do not disclose underlying facts or data in support of his opinions, and are unreliable." PO Mot. 1. Petitioner responds that the Clark Declaration includes citations of references relied upon. Paper 27, 2. Patent Owner's request is denied. Under 37 C.F.R. § 42.65(a), expert testimony that does not disclose the underlying facts or data on which the opinion is based is entitled to little or no weight. Thus, the issue is one of according the proper weight to the evidence presented, not a matter for evidence exclusion. Accordingly, this request is denied.

2. *Petitioner's Motion to Exclude*

Petitioner seeks exclusion of the Harstein Declaration (Exhibit 2012), and Exhibits 2013–2025, 2030, 2031, and 2035, referred to in the Harstein Declaration. Paper 23 ("Pet. Mot.").

i. Harstein Declaration

Exhibit 2012 is a declaration of Mr. Phil Harstein, the current president of Finjan. Petitioner seeks to exclude this declaration because of Harstein's alleged lack of personal knowledge of the events attested to, namely, the licensing activity and license agreements. *Id.* at 3–4. Petitioner further argues that Mr. Harstein's testimony is conclusory, does not disclose

supporting facts or data, is biased and unreliable, and provides no basis to support his qualifications as an expert. *Id.* at 4–5.

Patent Owner argues that Mr. Harstein testifies as to being familiar with Finjan’s licensing practices and the licensing of Finjan’s worldwide patent portfolio. Paper 24, 1. Further, to the extent Petitioner’s objections noted deficiencies in Mr. Harstein’s testimony laying the foundation for his knowledge of Finjan’s business records, Finjan submitted a Supplemental Declaration (Exhibit 2036) to cure the deficiencies. *Id.* at 1–2.

We do not agree with Petitioner that the Harstein Declaration lacks foundation for personal knowledge and that it should be excluded as hearsay or for lack of personal knowledge under Fed. R. Evid. 602. The Supplemental Declaration submitted by Patent Owner addresses Petitioner’s objections regarding whether the witness has sufficient personal knowledge of Finjan’s operations, business records, and his personal involvement in license negotiations. Ex. 2036 ¶ 15.

As to the general allegation that the Declaration constitutes hearsay, the allegation is vague and conclusory and thus Petitioner’s argument is not persuasive. As to whether the Harstein Declaration purports to supply unqualified expert opinions, we do not agree with Petitioner. The statement that Petitioner complains of is a factual statement that Finjan “invested over \$65 million dollars in developing patented technologies related to proactive content behavior inspection. Such investment contributed to Finjan being awarded 29 U.S. issued patents and 27 issued foreign patents.” Ex. 2012 ¶ 5. To the extent Petitioner has an issue with the veracity or factual support for the statement, that was a matter to be resolved on cross-examination, not as a basis for exclusion. And in any event, we are not persuaded that the

statement is unqualified expert opinion. Rather that statement is from the viewpoint of an officer of the company with knowledge of the company's investments, assets, and operation.

Therefore, Petitioner's request is denied.

ii. Exhibits

Petitioner urges exclusion of several exhibits (Exhibits 2013–2025, 2030, 2031, and 2035) that Patent Owner relies on in support for its contentions on objective indicia of nonobviousness. Pet. Mot. 4–6. However, we note that the objections that the exhibits are hearsay, irrelevant, prejudicial, misleading, and of minimal value, have little to no foundation, with bare assertions about the general nature of the documents and their alleged lack of authenticity. Patent Owner argues, and we agree, that hearsay exceptions of business records and government filings are applicable to most of these exhibits, and that the rest are not hearsay because they are not being used for the truth of the matter asserted. In any event, we need not address the merits of each of Petitioner's allegations, as we do not rely on these Exhibits to render our Final Written Decision. Therefore, the request is denied.

IV. ORDER

Accordingly, it is

ORDERED that claims 1–4, 8, and 11–13 of the '633 patent have been shown to be unpatentable; and

FURTHER ORDERED that claim 14 of the '633 patent has not been shown to be unpatentable;

FURTHER ORDERED that Petitioner's Motion to Exclude is denied;

FURTHER ORDERED that Patent Owner's Motion to Exclude is denied;

FURTHER ORDERED that, because this is a Final Written Decision, parties to the proceeding seeking judicial review of the decision must comply with the notice and service requirements of 37 C.F.R. § 90.2.

IPR2018-00391
Patent 7,647,633 B2

PETITIONER:

Patrick D. McPherson (Lead Counsel)
Patrick Muldoon
Joseph A. Powers
DUNE MORRIS LLP
PDMcPherson@duanemorris.com
PCMuldoon@duanemorris.com
JAPowers@duanemorris.com

PATENT OWNER:

James Hannah
Jeffrey H. Price
Michael Lee
KRAMER LEVIN NAFTALS & FRANKEL LLP
jhannah@kramerlevin.com
jprice@kramerlevin.com
mhlee@karamerlevin.com