

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

PALO ALTO NETWORKS, INC.,
Petitioner,

v.

CROGA INNOVATIONS LTD.,
Patent Owner.

IPR2024-01421
Patent 11,223,601 B2

Before TERRENCE W. McMILLIN, KEVIN C. TROCK, and RUSSELL E.
CASS, *Administrative Patent Judges*.

McMILLIN, *Administrative Patent Judge*.

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

I. INTRODUCTION

A. *Background and Summary*

Palo Alto Networks, Inc. (“Petitioner”)¹ filed a Petition for *inter partes* review of claims 1–16 of U.S. Patent No. 11,223,601 B2 (Ex. 1001, “the ’601 patent”). Paper 2 (“Pet.”), 1. Croga Innovations Ltd. (“Patent Owner”)² filed a Preliminary Response. Paper 6 (“Prelim. Resp.”).

Patent Owner’s Preliminary Response does not address the merits of Petitioner’s challenges to patentability. *See generally* Prelim. Resp. The only issue raised in the Preliminary Response is discretionary denial in light of a parallel district court case. *Id.*

We have authority to determine whether to institute an *inter partes* review. 35 U.S.C. § 314 (2018); 37 C.F.R. § 42.4(a) (2020) (“The Board institutes the trial on behalf of the Director.”). The standard for institution is set forth in 35 U.S.C. § 314(a), which provides that *inter partes* review may not be instituted unless “there is a reasonable likelihood that the petitioner would prevail with respect to at least 1 of the claims challenged in the petition.” After considering the Petition and the Preliminary Response, as well as the evidence of record, we institute an *inter partes* review as to claims 1–16 of the ’601 patent.

B. *Related Proceedings*

The parties identify the following district court matters related to the ’601 patent: *Croga Innovations Ltd. v. Cisco Systems, Inc.*, No. 2-24-cv-

¹ Petitioner identifies itself, Palo Alto Networks, Inc., as the real party-in-interest to this proceeding. Pet. 3.

² Patent Owner identifies itself, Croga Innovations Ltd., as the real party-in-interest to this proceeding. Paper 4, 2.

00065 (E.D. Tex.); and *Croga Innovations Ltd. v. Palo Alto Networks, Inc.*, No. 2-24-cv-00208 (E.D. Tex.). Pet. 3; Paper 4, 2; Paper 6, 12.

In addition, the parties also identify another petition for *inter partes* review of the '601 patent, *Cisco Systems, Inc., v. Croga Innovations Ltd.*, IPR2024-01283. Pet. 3, Paper 4, 2. A decision granting institution of *inter partes* review of claims 1–16 was entered in IPR2024-01283 on February 13, 2025. IPR2024-01283, Paper 8. Petitioner asserts that it was not involved in the petition filed in IPR2024-01283 and states “that IPR petition did not identify or assert any of the prior art that is asserted here.” Pet. 3.

C. The '601 Patent (Ex. 1001)

The '601 patent is titled “Network Isolation for Collaboration Software.” Ex. 1001, code (54). The '601 patent discloses methods and systems “for isolation of multi-user interactive software (e.g., collaboration software) on a host computer system,” which may include a workspace and an isolated computing environment. *Id.* at 1:58–65. Examples of collaboration software include “screen sharing applications, collaborative meeting applications, remote desktop applications, gaming applications, shared document application, file sharing applications, and/or the like,” such as “WebEx, GoToMeeting, etc.” *Id.* at 4:45–47, 4:56–60.

According to the '601 patent, “[o]ne or more applications may run in the workspace via a first memory space (e.g., a trusted memory space),” and “[t]he isolated computing environment may be isolated from the workspace by an internal isolation firewall.” Ex. 1001, 1:62–67. The '601 patent further describes that “[o]ne or more isolated applications may run in the isolated computing environment via a second memory space,” wherein “[t]he isolated applications may have access to one or more Internet

destinations (e.g., untrusted Internet destinations)” and “communicate with the Internet destinations via an authentication device.” *Id.* at 2:8–13. The ’601 patent also describes a host-based firewall, which “may be configured to isolate the host computer system from communicating with one or more devices on the network.” *Id.* at 2:44–47.

The ’601 patent describes that the isolated applications may include a multi-user interactive software application, which “may be an application that allows one or more remote machines to control (e.g., access, view, interact with, download from, upload to, and/or modify) an application running on the host computer system.” Ex. 1001, 2:18–23. For example, “the multi-user interactive software application may allow the remote machines to view a display of the host computer system (e.g., for a presentation or screen-sharing).” *Id.* at 2:23–26.

According to the ’601 patent, “[i]solation of collaboration software may protect the host computer system from collaboration software-based interactions that may bypass other network security controls.” *Id.* at 4:12–

15. The ’601 patent describes systems and methods for providing:

a layer of communication isolation between collaboration software applications and untrusted network destinations in order to prevent malware from effectively exfiltrating data from the host computer system and/or establishing command and control channels with an attacker resources in order to attempt to gain control of the host computer system.

Id. at 4:36–42. Specifically, the ’601 patent describes “a sandbox protected application or process and/or an internal isolation firewall running” on the host computer system. *Id.* at 4:61–65. For example, “collaboration software may wholly or partially be contained in a sandboxed computing environment (e.g., an isolated computing environment).” *Id.* at 4:65–5:1.

Figure 2, reproduced below, illustrates a “host computer system configured to participate in a software based meeting with one or more other host computer systems,” wherein the host computer system uses “a sandbox based network isolation system.” Ex. 1001, 3:61–65.

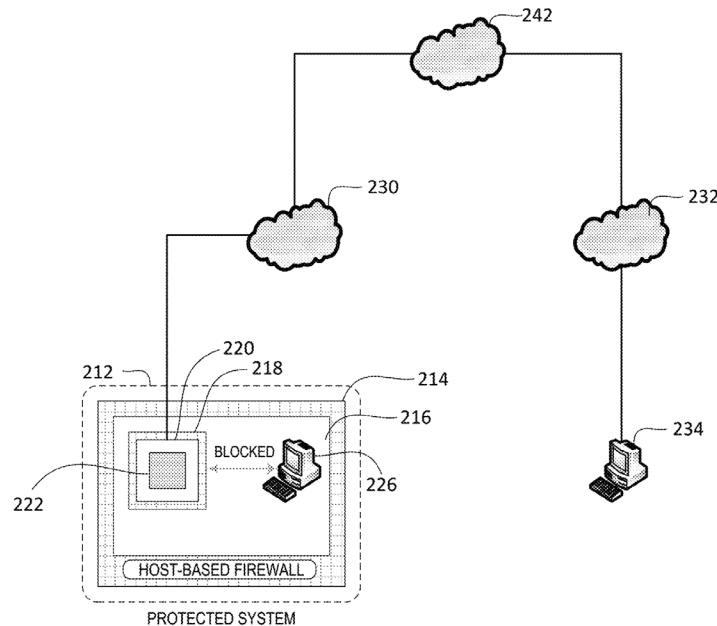


FIG. 2

Figure 2 depicts the host computer system 212 including “a host-based firewall 214, an operating system 226, a sandboxed computing environment (e.g., an isolated computing environment) 220, an internal isolation firewall 218, and a workspace 216.” *Id.* at 9:66–10:3. The host computer system 212 is configured to “isolate communication with untrusted network destinations using one or more of the host-based firewall 214, a border firewall (not shown), a web proxy (not shown), an internal isolation firewall 218, or a segregation of untrusted memory space.” *Id.* at 10:27–31. “The

sandboxed computing environment 220 may be enforced via the sandbox container process,” which “may be a security mechanism used to separate resources associated with the sandboxed computing environment 220 from other applications and/or processes that may be running on the workspace 216 of the host computer system 212.” *Id.* at 10:34–40.

The ’601 patent describes that “host computer system 234 may be configured to interact with the host computer system 212 via the collaboration software application (e.g., multi-user interactive software application) 222,” wherein “[t]he host computer system 212 may allow the host computer system 234 to access the sandboxed computing environment 220.” Ex. 1001, 12:4–7, 12:16–18. According to the ’601 patent,

[o]perating the collaboration software application 222 within the sandboxed computing environment 220 may restrict command and control between the host computing system 212 and the untrusted device to the sandboxed computing environment 220 and/or the untrusted memory space of the host computing device 212.

Id. at 11:65–12:3. The ’601 patent explains that

[i]f malware is transferred to the host computer system 212 due to interactions with the host computer system 234, the malware may be isolated to the sandboxed computing environment 220, which may make it difficult for the malware to infect the workspace 216 of the host computer system 212 and/or other devices on the network 230.

Id. at 12:25–30.

D. Challenged Claims

Petitioner challenges claims 1–16 (all) of the ’601 patent. Pet. 1. Of the challenged claims, claims 1 and 9 are independent claims. Ex. 1001, 22:42–63; 23:33–24:8. Claim 1 recites:

1. A system comprising:

a memory; and

a processor configured to:

implement a workspace that uses a host operating system, wherein the workspace enables operation of a first set of one or more applications or processes via a first memory space;

implement an isolated computing environment that uses the host operating system, wherein the isolated computing environment enables operation of a second set of one or more applications or processes via a second memory space, and wherein the second set of one or more applications or processes comprises a multi-user interactive software application;

isolate the isolated computing environment from the workspace using an internal isolation firewall;

authenticate the isolated computing environment with an authentication device; and

send data to an untrusted destination from the multi-user interactive software application via a proxy device when the isolated computing environment has been authenticated.

Ex. 1001, 22:42–63.

E. The Asserted Grounds

Petitioner challenges claims 1–16 of the '601 patent based on the grounds set forth in the table below.

Claims Challenged	35 U.S.C. §	Reference(s)/Basis ³
1–4, 7, 9–12, 15	103 ⁴	Walsh ⁵ , AAPA, Innes ⁶
5, 6, 13, 14	103	Walsh, AAPA, Innes, Netfilter ⁷
8, 16	103	Walsh, AAPA, Innes, Franco ⁸
1–4, 7, 9–12, 15	103	Austin, ⁹ AAPA, Bloch ¹⁰
5, 6, 13, 14	103	Austin, AAPA, Bloch, Walker ¹¹
8, 16	103	Austin, AAPA, Bloch, Franco

Pet. 2, 12, 15, 41, 44, 48, 76, 81.¹²

³ Petitioner contends that the cited art qualifies as prior art under applicable law. *See* Pet. 12–15. Patent Owner does not dispute the prior art status of the cited art. *See* Prelim. Resp. We preliminarily determine that the cited art qualifies as prior art.

⁴ The Leahy-Smith America Invents Act, Pub. L. No. 112-29, 125 Stat. 284 (2011) (“AIA”), amended 35 U.S.C. § 103, and was effective on March 16, 2013. The application for the ’601 patent is U.S. Application No. 16/141,535, filed on September 25, 2018, which claims priority to a provisional application filed on September 28, 2017. Ex. 1001, codes (21), (22), (60), 1:7–9. Because the application for the ’601 patent claims a priority date after the effective date of the applicable AIA amendment, the post-AIA version of § 103 applies for purposes of institution.

⁵ US 8,640,187 B2, issued Jan. 28, 2014, filed May 28, 2010 (Ex. 1006).

⁶ US 9,154,488 B2, issued Oct. 6, 2015, filed May 3, 2013 (Ex. 1012).

⁷ The netfilter.org project, January 2016, originally available at: <http://netfilter.org>, currently available at: <https://web.archive.org/web/20160101194846/http://netfilter.org/> (Ex. 1013).

⁸ US 8,161,563 B2, issued Apr. 17, 2012, filed Jul. 20, 2010 (Ex. 1009).

⁹ US 2016/0306964 A1, published Oct. 20, 2016 (Ex. 1005).

¹⁰ US 7,849,502 B1, issued Dec. 7, 2010, filed Apr. 30, 2007 (Ex. 1007).

¹¹ US 2016/0149863 A1, published May 26, 2016 (Ex. 1008).

¹² While the tables of grounds in the Petition on pages 2 and 12 do not include AAPA (Applicant Admitted Prior Art) in its combinations of references, the Petition does include AAPA in the headings for each ground. *See* Pet. i–iii (Table of Contents). Accordingly, we include AAPA here.

Petitioner supports its showing of unpatentability of the challenged claims of the '601 patent with the Declaration of Samrat Bhattacharjee, Ph.D. (Ex. 1003). Patent Owner has not filed a declaration or otherwise challenged the testimony of Dr. Bhattacharjee. Accordingly, at this stage of this proceeding, the testimony of Dr. Bhattacharjee in support of Petitioner's contentions is unrebutted.

II. DISCRETIONARY DENIAL OF INSTITUTION

Institution of an *inter partes* review is discretionary. *See* 35 U.S.C. § 314(a) (authorizing institution of an *inter partes* review under particular circumstances, but not requiring institution under any circumstances); *Cuozzo Speed Techs., LLC v. Lee*, 579 U.S. 261, 273 (2016) (“[T]he agency’s decision to deny a petition is a matter committed to the Patent Office’s discretion.”); *SAS Inst., Inc. v. Iancu*, 138 S. Ct. 1348, 1356 (2018) (“[Section] 314(a) invests the Director with discretion on the question whether to institute review” (emphasis omitted)); *Harmonic Inc. v. Avid Tech., Inc.*, 815 F.3d 1356, 1367 (Fed. Cir. 2016) (“[T]he PTO is permitted, but never compelled, to institute an IPR proceeding.”).

Patent Owner urges the Board to exercise discretion to deny institution of *inter partes* review under § 314(a) based on the factors set forth in *Apple Inc. v. Fintiv, Inc.*, IPR2020-00019, Paper 11 at 5–6 (PTAB Mar. 20, 2020) (precedential) (“*Fintiv*”). *See generally* Prelim. Resp. Petitioner contends that discretionary denial would be inappropriate. Pet. 86–88.

An advanced state of a parallel district court proceeding is a “factor that weighs in favor of denying the Petition under § 314(a).” *NHK Spring Co. v. Intri-Plex Techs., Inc.*, IPR2018-00752, Paper 8 at 20 (PTAB Sept.

12, 2018) (precedential) (“*NHK*”). Specifically, an early trial date is part of a “balanced assessment of all relevant circumstances in the case, including the merits.” Consolidated Trial Practice Guide November 2019, 58.¹³

This balanced assessment involves consideration of the following factors:

1. whether the court granted a stay or evidence exists that one may be granted if a proceeding is instituted;
2. proximity of the court’s trial date to the Board’s projected statutory deadline for a final written decision;
3. investment in the parallel proceeding by the court and the parties;
4. overlap between issues raised in the petition and in the parallel proceeding;
5. whether the petitioner and the defendant in the parallel proceeding are the same party; and
6. other circumstances that impact the Board’s exercise of discretion, including the merits.

Fintiv, Paper 11 at 5–6. Our analysis of the *Fintiv* factors is provided below.

Factor 1: whether the court granted a stay or evidence exists that one may be granted if a proceeding is instituted

A motion for a stay of the parallel district court proceeding has been filed. *See* Prelim. Resp. 4. Our review of the district court docket indicates that the motion for a stay has not been ruled on.

Because the District Court has not ruled on the pending motion to stay, this factor does not weigh for or against denying institution in this case. *See Fintiv*, IPR2020-00019, Paper 15 at 12; *Sand Revolution II, LLC v. Cont’l Intermodal Group – Trucking LLC*, IPR2019-01393, Paper 24 at 7 (PTAB June 16, 2020) (informative).

¹³ Available at <https://www.uspto.gov/TrialPracticeGuideConsolidated>.

Factor 2: proximity of the court’s trial date to the Board’s projected statutory deadline for a final written decision

Jury selection in the parallel district court case is currently scheduled for October 20, 2025. Pet. 87; Prelim. Resp. 6. A final written decision in this proceeding must be entered within one year of this institution decision unless the time is extended for good cause by the Chief Administrative Patent Judge, or adjusted by the Board in case of joinder. 37 C.F.R. § 42.100(c). Thus, our final written decision will be at least several months after trial in the parallel district court proceeding is currently scheduled. “If the court’s trial date is earlier than the projected statutory deadline, the Board generally has weighed this fact in favor of exercising authority to deny institution under *NHK*.” *Fintiv*, Paper 11 at 9.

This factor supports denying institution.

Factor 3: investment in the parallel proceeding by the court and the parties

Patent Owner argues that “the parties and the district court will have invested significant time and resources in the litigation by the time the Board reaches an institution decision in this matter,” because “[b]y the date the institution decision is due, the parties will have completed claim construction discovery and submitted opening and responsive claim construction briefs.” Prelim. Resp. 8. Petitioner argues:

By the time of institution, much work will remain in the parallel litigation. For example, discovery and substantive motion practice will still be ongoing. EX1035. Currently, the court has not made any substantive rulings (e.g., a claim construction order). Petitioner’s diligence filing the petition six months after filing of the complaint, within two months after the supplemental infringement contentions, within six weeks

after the invalidity contentions, and six months before the statutory deadline favors institution.

Pet. 87.

Our review of the docket from the district court proceeding indicates that no substantive orders have been entered. *See Fintiv*, Paper 11 at 10 (“If, at the time of the institution decision, the district court has not issued orders related to the patent at issue in the petition, this fact weighs against exercising discretion to deny institution under *NHK*.”).

And, it does appear to us that Petitioner filed the Petition somewhat expeditiously after the parties exchanged infringement and invalidity contentions in the parallel district court proceeding and did not wait to file the Petition until shortly before the statutory deadline. *See Fintiv*, Paper 11 at 10 (“If the evidence shows that the petitioner filed the petition expeditiously, such as promptly after becoming aware of the claims being asserted, this fact has weighed against exercising the authority to deny institution under *NHK*.”).

This factor weighs in favor of not exercising discretion to deny institution.

Factor 4: overlap between issues raised in the petition and in the parallel proceeding

A broad stipulation by a petitioner to not pursue in a parallel district court proceeding any ground raised or that could have been reasonably raised in its *inter partes* review petition mitigates any concerns of duplicative efforts between the district court and the Board, as well as concerns of potentially conflicting decisions. *See Sotera Wireless, Inc. v. Masimo Corp.*, IPR2020-01019, Paper 12 at 19 (PTAB Dec. 1, 2020)

(precedential as to § II.A) (“*Sotera*”). Petitioner has presented such a stipulation in this proceeding. Paper 7 (Petitioner’s *Sotera* Stipulation). Petitioner’s *Sotera* Stipulation states that “Petitioner will not raise in district court any grounds raised or that reasonably could have been raised in the Petition.” *Id.* at 1.

This factor weighs strongly in favor of not exercising discretion to deny institution. *See Sotera*, Paper 12 at 19.

Factor 5: whether the petitioner and the defendant in the parallel proceeding are the same party

The parties are the same in the parallel district court proceeding. This factor supports denying institution.

Factor 6: other circumstances that impact the Board’s exercise of discretion, including the merits

As discussed below, on this preliminary record, Petitioner has met its burden of demonstrating a reasonable likelihood that it would prevail in showing that claims of the ’601 patent are unpatentable. Although we recognize the record may change during trial, as discussed in detail below, Petitioner has made a sufficiently persuasive showing, on the record presently before us, that the prior art cited in the Petition teaches or suggests all limitations of at least one challenged claim. *See Fintiv*, Paper 11 at 14–15 (discussing the merits of the Petition as a consideration).

This factor does not weigh for or against denying institution in this case. *See Sotera*, Paper 12 at 20.

Conclusion on Discretionary Denial

Factor 4 strongly supports not discretionarily denying institution, Factor 3 supports not denying institution, Factors 2 and 5 support denying

institution, and Factors 1 and 6 are neutral. We take “a holistic view of whether efficiency and integrity of the system are best served by denying or instituting review” when considering the six *Fintiv* factors. *Fintiv*, Paper 11 at 6. Our holistic review of the *Fintiv* factors, namely that, although there is currently an early trial date in the parallel district court proceeding, the *Sotera* stipulation mitigates concerns of duplicative effort and overlap and the relatively limited investment in the parallel proceeding to date indicates that the *Fintiv* factors weigh in favor of instituting *inter partes* review. For these reasons, we do not exercise our discretion to deny institution under Section 314(a).

III. OBVIOUSNESS ANALYSIS

A. Principles of Law

Under 35 U.S.C. § 103, “[a] patent claim is unpatentable 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.” *See KSR Int’l Co. v. Teleflex Inc.*, 550 U.S. 398, 406 (2007) (similar language). The question of obviousness involves resolving 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 ordinary skill in the art; and when presented (4) objective evidence of non-obviousness (not presented here). *Graham v. John Deere Co.*, 383 U.S. 1, 17–18 (1966). Further, “there must be some articulated reasoning with some rational underpinning to support the legal conclusion of obviousness.” *See KSR*, 550 U.S. at 418.

B. Level of Ordinary Skill in the Art

Petitioner contends that

A POSA [(Person of Skill in the Art)] in the technical field of the '601 patent at the relevant time would have had a B.S. degree in Computer Science, Computer Engineering, or an equivalent field, as well as at least 2–3 years of academic or industry experience in the design, analysis, and implementation of operating systems, as well as the design, analysis, and monitoring of computer networks, including issues of network security and network administration, or comparable industry experience.

Pet. 11 (citing Ex. 1003 ¶ 30). At least at this stage of this proceeding, Patent Owner does not address the level of ordinary skill in the art. *See generally* Prelim. Resp.

Determining the level of ordinary skill in the art involves various factors, including the “type of problems encountered in the art; prior art solutions to those problems; rapidity with which innovations are made; sophistication of the technology; and educational level of active workers in the field.” *In re GPAC Inc.*, 57 F.3d 1573, 1579 (Fed. Cir. 1995) (citation omitted). The prior art of record also reflects the level of ordinary skill in the art. *See Okajima v. Bourdeau*, 261 F.3d 1350, 1355 (Fed. Cir. 2001). For purposes of this institution decision, we adopt the assessment offered by Petitioner.¹⁴ It is not disputed by the Patent Owner and is consistent with the '601 patent and the asserted prior art.

¹⁴ No matter how designated in this decision, any determination (except our determination to institute trial) is preliminary and non-binding. We wish to have the full record as developed during trial before rendering any binding determination, finding, or conclusion.

C. Claim Construction

In *inter partes* reviews, the Board construes claims using the same claim construction standard employed in a civil action under 35 U.S.C. § 282(b). 37 C.F.R. § 42.100(b) (2023). The “words of a claim ‘are generally given their ordinary and customary meaning,’” as would have been understood by a person of ordinary skill in the art at the time of the invention. *Phillips v. AWH Corp.*, 415 F.3d 1303, 1312 (Fed. Cir. 2005) (en banc) (quoting *Vitronics Corp. v. Conceptronic, Inc.*, 90 F.3d 1576, 1582 (Fed. Cir. 1996)). “In determining the meaning of the disputed claim limitation, we look 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 1312–1317).

Petitioner asserts that “[n]o claim term needs an explicit construction.” Pet. 11. At least at this stage of this proceeding, Patent Owner does not address claim construction. *See generally* Prelim. Resp.

There are no disputes between the parties relating to claim construction. Accordingly, at this stage, no need exists to expressly construe any claim terms to resolve any dispute between the parties.¹⁵ *See Nidec Motor Corp. v. Zhongshan Broad Ocean Motor Co.*, 868 F.3d 1013, 1017

¹⁵ If either party contends that explicit claim construction is necessary in order to make a final determination whether or not any challenged claim is unpatentable based on the arguments and evidence presented, it should clearly explain why during trial and provide a clear and unambiguous construction with supporting evidence including specifically identifying the challenges, claims, and limitations to which the construction is necessary.

(Fed. Cir. 2017) (“[W]e need only construe terms ‘that are in controversy, and only to the extent necessary to resolve the controversy.’” (quoting *Vivid Techs., Inc. v. Am. Sci. & Eng’g, Inc.*, 200 F.3d 795, 803 (Fed. Cir. 1999))).

D. Asserted Obviousness of Claims 1–4, 7, 9–12, and 15 Based on Walsh, AAPA, and Innes

Petitioner challenges claims 1–4, 7, 9–12, and 15 as being obvious over Walsh, AAPA, and Innes. *See* Pet. 12, 15–41. We start with a description of the disclosures of Walsh and Innes, and then consider the arguments and evidence presented by Petitioner. Patent Owner does not dispute any aspect of the showing in the Petition as to unpatentability of any challenged claim. *See generally* Prelim. Resp.

1. Walsh (Ex. 1006)

Walsh is titled “Systems and Methods for Providing a[] Fully Functional Isolated Execution Environment for Accessing Content.” Ex. 1006, code (54). Walsh relates to “[a] sandbox tool [that] can cooperate with components of a secure operating system to create an isolated execution environment for accessing content without exposing other processes and resources of the computing system to the untrusted content.” *Id.* at code (57).

As background, Walsh notes that “[d]ue to the rise of wide-area public networks, users have unlimited access to content, e.g. data, files, applications, programs, etc., from a variety of sources,” and that “[o]ften, the users are unaware of the origin of the content available in the public networks.” *Id.* at 1:13–17. According to Walsh, “to prevent attacks, the users utilize filtering programs, anti-virus programs, etc. in order to identify and block known dangerous content.” *Id.* at 1:26–28. Walsh states that these programs suffer from weaknesses such as lack of ability to stop

previously unknown and emerging threats and the fact that filtering and anti-virus programs are themselves subject to attack. *Id.* at 1:28–35.

Walsh also states that “[c]urrently, operating systems allow a user to place access controls on a process, such as an application program, running on the operating system,” such that “[w]hen an action by the application program exceeds it[s] level of access, the operating system blocks the action, and the application program immediate shuts down.” *Id.* at 1:42–47. Walsh explains that “[t]his prevents possible damage to the user’s computing system, but it also prevents the user from using the application program,” because “[t]o utilize the application program, the user is required to increase the level of access granted.” *Id.* at 1:47–50. That is, according to Walsh, “the user has no method to run the application program and also protect the user’s computing system from potential danger.” *Id.* at 1:50–53.

Walsh teaches a “sandbox tool” that can create an execution environment that is isolated from other processes executing on a computing system. *Id.* at 2:36–39. According to Walsh, “[t]he sandbox tool can cooperate with components of a secure operating system (OS), such as security enhanced LINUX (SELinux), to create an isolated execution environment for accessing content without exposing other processes and resources of the computing system to the content.” *Id.* at 2:39–44.

Figure 2 of Walsh, reproduced below, illustrates an exemplary software environment for utilizing an isolated execution environment. *Id.* at 1:65–66.

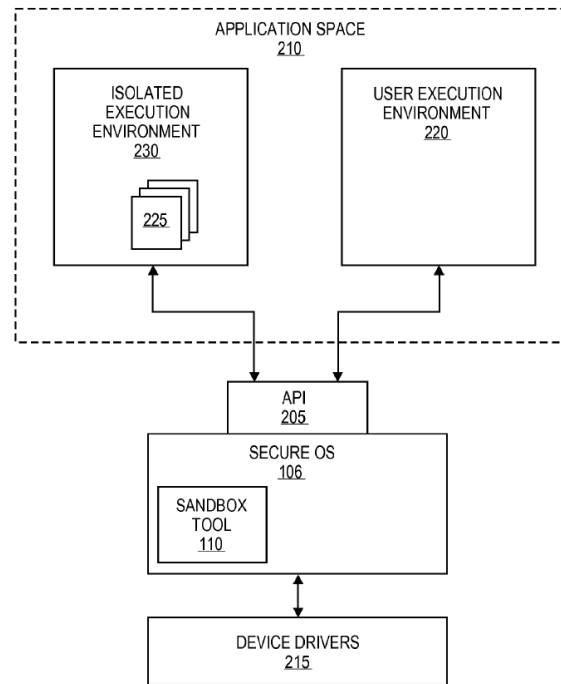


FIG. 2

Figure 2 depicts a software environment including a secure OS 106 and an application space 210. *Id.* at 5:10–21. According to Walsh, “secure OS 106 can be configured to create a user execution environment 220 in the application space 210.” *Id.* at 5:28–30. Walsh explains that “[t]he user execution environment 220 allows users to interact with [a] user computing system . . . to access content such as run application and programs, view files, etc.” *Id.* at 5:30–32.

Walsh further teaches that in order to provide access to content 225 without endangering the user computing system, sandbox tool 110 can be configured to cooperate with components of secure OS 106, to create an isolated execution environment 230 for accessing content 225 (trusted or untrusted) without exposing other processes such as the user execution

environment 220 and resources of the user computing system to the content 225. *Id.* at 5:50–57.

2. *Innes (Ex. 1012)*

Innes is titled “Secured Access to Resources Using a Proxy.” Ex. 1012, code (54). Innes relates to “providing approaches for a client device to securely access resources using a proxy device.” *Id.* at 1:6–8. According to Innes, in a virtual private network (“VPN”), “a proxy device, simply acting as a transparent pipe, may provide the necessary connection between the client device and the private network.” *Id.* at 1:14–16. Innes notes various problems with traditional VPNs, such as the amount of trust put on the client device because the client device can directly access resources on the private networks. *Id.* at 1:16–19. Innes further notes that in a traditional VPN, “the client device must be configured for the communication protocol of the resource and/or private network,” and “the proxy device cannot control or monitor the activities of the client device.” *Id.* at 1:19–22.

Figure 7 of Innes, reproduced below, illustrates “a system having a client, a proxy, resource(s), and/or authentication service(s).” *Id.* at 3:14–15.

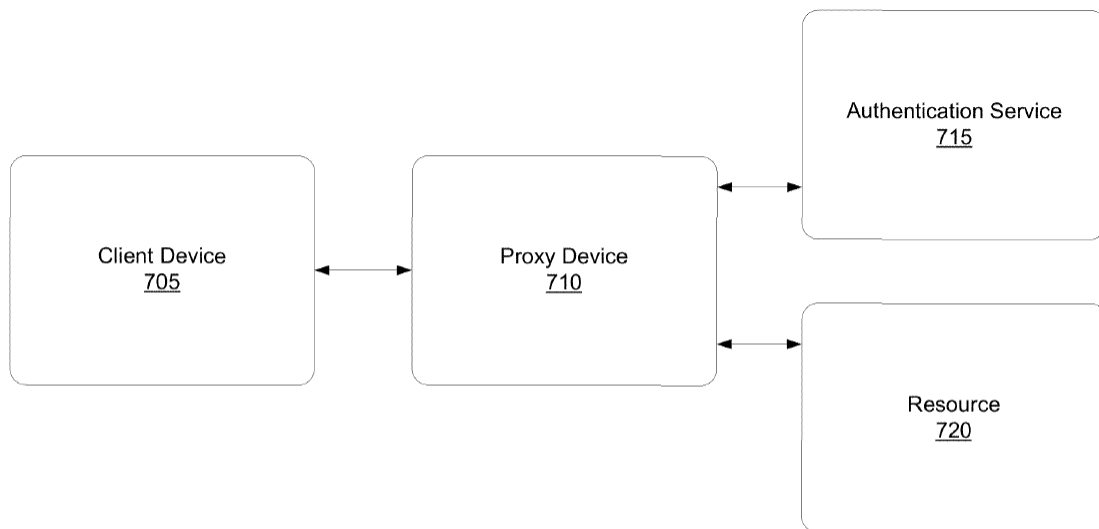


FIG. 7

Figure 7 depicts a system that includes a client device 705, a proxy device 710, one or more authentication services 715 and one or more resources 720. *Id.* at 22:2–5. Innes teaches that “[t]he client device 705 may communicate with one or more resources 720 and/or authentication services 715 using . . . proxy device 710.” *Id.* Innes explains that “client device 705 might not be configured to communicate directly with the resources 720 and/or authentication services 715,” because for example, “the client device 705 and resources 720 may use different authentication and/or communication protocols.” *Id.* at 22:5–9. According to Innes, “[t]he proxy device 710 may translate between these different protocols.” *Id.* at 22:9–10.

According to Innes:

The client device 705 may send a request for resources 720, such as documents, emails, services, files, and the like, to the proxy device 710. The proxy device 710 may forward the request to the resource 720, and in response, authentication between the proxy device 710 and resource 720 may be initiated. At one or more points during the authentication, the

resource 720 may request a signature, such as from a client 20 certificate. The proxy device 710 might not directly have access to the client certificate, so the proxy device 710 may involve the client device 705 in the authentication process, as if the client device 705 controls access to the client certificate. For example, the proxy device 710 may request that the client device 705 sign or decrypt an authentication message using the client certificate (or a private key included therein), or return a list of available security certificates or a selection by the user of a particular security certificate.

Id. at 22:13–28.

3. *Analysis of Independent Claims 1 and 9*

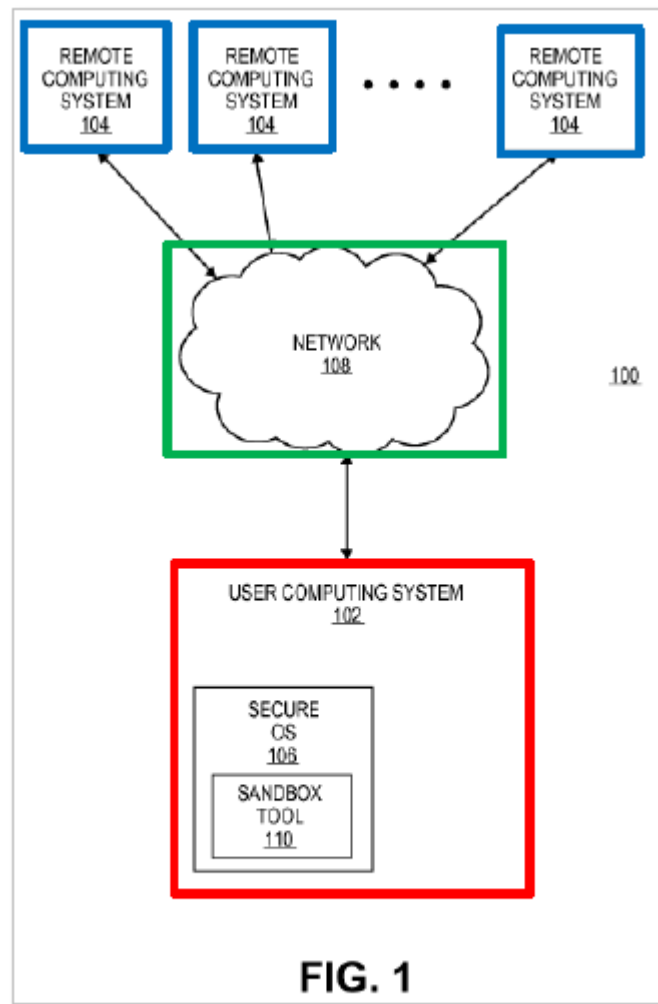
For the reasons that follow, we are persuaded that Petitioner establishes sufficiently for institution that the subject matter of independent claims 1 and 9 would have been obvious in view of the combination of Walsh, AAPA, and Innes. Our analysis of the showing in the Petition for claims 1 and 9 is provided below.¹⁶

a. *Motivation to Combine the Teachings of Walsh, AAPA, and Innes*

Petitioner contends that “[a] POSA would have been motivated to combine Walsh with Innes based on Walsh’s disclosure.” Pet. 15 (heading

¹⁶ This analysis is limited to considering the arguments and evidence presented in the Petition. As noted previously, Patent Owner argues only for discretionary denial. *See generally* Prelim. Resp. On the merits of obviousness, Patent Owner provides only the conclusory statement that, “[t]he Petition does not present a reasonable likelihood of invalidity under any ground or challenged claim.” *Id.* at 11–12. This conclusory statement is unsupported by any argument or evidence that challenges or contradicts any aspect of Petitioner’s showing that the challenged claims would have been obvious and are, thus, unpatentable. Stated more succinctly, at this stage, Petitioner’s obviousness contentions and evidence are undisputed by Patent Owner.

VI.A.1. (emphasis omitted)). In support of this contention, the Petition provides an annotated version of Figure 1 of Walsh, reproduced below,



EX1006, FIG. 1 (annotated).

Id. at 18. Figure 1 of Walsh depicts “an exemplary network 100 of computing systems.” Ex. 1006, 3:24–25. With regard to annotated Figure 1 of Walsh, the Petition states:

A POSA would have been motivated to combine Walsh with Innes based on the teachings in Walsh itself. EX1003, ¶¶136-148. Referring to annotated Figure 1 ([above]), Walsh discloses that a “user computing system 102” (in red) is connected to other “remote computing systems 104” (in blue)

via “network 108” (in green). EX1006, 3:24-54. Walsh says that the “user computing system 102” can access content via “network 108” using “a client-server model.” EX1006, 4:66-5:3. To the extent that Walsh does not provide details about the client-server model, a POSA would have been motivated to combine Walsh with a reference, like Innes, that has those details. EX1003, ¶141.

Pet. 17–18. With regard to Innes, the Petition states:

Like Walsh, Innes discloses that remote computers can access content via a network “using a web browser.” EX1012, 4:27-47, FIG. 1. The remote computers “may wish to access enterprise resources that require authentication.” EX1012, 23:43-45. To provide such authentication, [Innes] discloses a “proxy device 710” . . . that can provide “authentication service 715” to a “client device 705”.

Id. at 18.

Petitioner also argues that “[a] POSA would have been further motivated to implement Walsh’s user computing system on a network including Innes’s proxy device and authentication services to improve computer and network security.” *Id.* at 19. In support, the Petition provides:

To provide additional protection, a POSA would have been motivated to place the computing device on a network with Innes’s proxy device. EX1003, ¶147. As noted above, Walsh’s computing devices access the trusted/untrusted content over a network using a web browser. EX1006, 3:64-4:4. The content, Innes teaches, can be “securely” accessed “using a proxy device.” EX1012, 1:6-8. For example, Innes’s proxy device is configured to establish an authenticated session with the requested resource and provide session information to the computing device. EX1012, 22:15-18. The computing device uses the session information to verify the authentication session between the proxy device and resource. EX1012, 22:37-40. Combining Innes’s authentication techniques with Walsh’s computing device would have provided an additional layer of

security for accessing the trusted/untrusted content over the network. EX1003, ¶147.

Id. at 20.

Petitioner further argues that the combination would have been obvious to try:

In addition, the combination of Walsh and Innes would have been obvious to try. EX1003, ¶177. In the client-server context disclosed by Walsh, EX1006, 4:66-5:3, machines can access the Internet using known paradigms, for example, using a network-address-translation device or using a proxy. EX1003, ¶177. Thus, it would have been obvious for a POSA to try to connect Walsh's computing system to the Internet via a proxy device as disclosed by Innes. *Id.*

Id.

With regard to reasonable expectation of success, the Petition states:

A POSA would have had a reasonable expectation of success combining Walsh with Innes because the combination would amount to nothing more than placing Walsh's computing device on a network with Innes's proxy device. EX1003, ¶148. Walsh's computing device is a standard computing system. EX1006, 3:42-46. It communicates and shares data via a network that can be wired or wireless. EX1006, 3:57-63. Innes's proxy device communicates with a client device (e.g., Walsh's computing device) over a network to provide resource access. EX1012, 22:13-46.

Id.

Petitioner articulates with supporting reasoning and evidence why a skilled artisan would have been motivated to combine the relied-on teachings of Walsh and Innes. We determine that Petitioner establishes motivation to combine the relevant teachings of Walsh and Innes.

Petitioner contends that “[a] POSA would have been motivated to combine Walsh-Innes with AAPA [(Applicant Admitted Prior Art)].” Pet. 21 (heading VI.A.2. (emphasis omitted)). In support of this contention, the Petition provides:

The ’601 patent admits that the claimed “*multi-user interactive software application*” can be implemented via commercially available products such as WebEx. EX1001, 4:43-47. WebEx was a well-known product and was used as a standalone application, an email client plugin, or within a web browser. EX1003, ¶¶149-150.

A POSA would have been motivated, with a reasonable expectation of success, to use Innes’s proxy appliance with Walsh’s client computer in order to “securely” access WebEx traffic via Walsh’s computer system. EX1003, ¶150; EX1012, 1:6-8. Walsh explains that a user computer system may access content over the Internet via a “web browser” application. EX1006, 3:67-4:4. A POSA would have understood that WebEx was a well-known application for accessing traffic over the Internet, and a POSA would have understood that WebEx would be (i) installed as a standalone program on Walsh’s computer system or (ii) used within a web browser or email client on Walsh’s computer system. EX1003, ¶150. The combined Walsh-AAPA system would then be used on a network including Innes’s proxy device to provide Walsh’s client device with secure access to network resources (e.g., an HTTP server). EX1012, 28:34-48. A POSA would not have experienced any unreasonable technical hurdles using Walsh’s computing device to access WebEx on a network including Innes’s proxy device. EX1003, ¶150.

Id. at 21–22. Petitioner articulates with supporting reasoning and evidence why a skilled artisan would have been motivated to combine the relied-on teachings of Walsh and Innes with AAPA. We determine that Petitioner establishes motivation to combine the relevant teachings of Walsh, Innes and AAPA.

b. Independent Claims 1 and 9

Claim 1 is an independent system claim and claim 9 is an independent method claim. Ex. 1001, 22:42–63, 23:33–24:8. However, the language of claims 1 and 9 is, in large part, identical. *Compare id.* at 22:42–63 with *id.* at 23:33–24:8. For example, system claim 1 recites:

a processor configured to:

implement a workspace that uses a host operating system, wherein the workspace enables operation of a first set of one or more applications or processes via a first memory space.

Id. at 22:44–48 (emphasis added). And, method claim 9 recites:

implementing a workspace that uses a host operating system, wherein the workspace enables operation of a first set of one or more applications or processes via a first memory space.

Id. at 23:34–37 (emphasis added). In the remaining limitations of claims 1 and 9, the only differences are that the limitations in claim 1 begin with the active form of a verb (implement, isolate, authenticate, send) and the limitations in claim 9 begin with the present participle or “ing” form of the same verb (implementing, isolating, authenticating, sending). *See id.* at 22:49–63, 23:38–24:8. In the Petition, the discussion of claims 1 and 9 is combined. *See* Pet. 22–36. Our limitation-by-limitation analysis of the showing in the Petition for claims 1 and 9 is provided below.

*A system (claim 1)/method (claim 9) comprising:*¹⁷

Petitioner relies on Walsh as disclosing the preambles of claims 1 and 9. Pet. 22–23. The Petition provides:

¹⁷ Petitioner does not discuss whether the preambles of claims 1 and 9 are limiting. *See* Pet. 22 (“To the extent the preambles are limiting, Walsh discloses them.”). We determine that the preambles of claims 1 and 9 are

Walsh discloses “a user computing system 102” (a “*system*”) included within a network 100. EX1006, 3:35-38. Walsh’s “user computing system 102” implements a “*method*” because it creates an isolated execution environment and accesses “content (trusted or untrusted) without exposing other processes and resources of the user computing system 102 to the content.” EX1003, ¶151; EX1006, 4:30-35.

Id. at 22. We determine that Walsh sufficiently discloses a system and a method to support institution.

a memory (claim 1);

Petitioner relies on Walsh as disclosing this limitation. Pet. 23. The Petition provides, “Walsh’s ‘user computing system 102’ includes ‘hardware resources’ such as ‘processors’ and ‘**memory**.’ EX1003, ¶¶152-153; EX1006, 3:38-43.” *Id.* (footnote omitted). The cited passage in Walsh states:

The computing systems, such as the user computing system 102 and remote computing systems 104, can include hardware resources, such as processors, **memory**, network hardware, storage devices, and the like, and software resources, such as operating systems (OS), application programs, and the like.

Ex. 1006, 3:38–43 (emphasis added). The cited passage in Walsh supports Petitioner’s contention that Walsh discloses this limitation.

a processor configured to (claim 1):

Petitioner relies on Walsh as disclosing this limitation. Pet. 23–24. The Petition provides, “Walsh’s ‘user computing system 102’ includes

not limiting because they do not recite essential structure or step and are not necessary to give life, meaning, and vitality to the claims. *See Shoes by Firebug LLC v. Stride Rite Children’s Group, LLC*, 962 F.3d 1362, 1367 (Fed. Cir. 2020).

‘hardware resources’ such as ‘**processors**’ and ‘memory.’ EX1003, ¶¶154-155; EX1006, 3:38-40.” *Id.* The cited passage in Walsh states:

The computing systems, such as the user computing system 102 and remote computing systems 104, can include hardware resources, such as **processors**, memory, network hardware, storage devices, and the like, and software resources, such as operating systems (OS), application programs, and the like.

Ex. 1006, 3:38–43 (emphasis added). The cited passage in Walsh supports Petitioner’s contention that Walsh discloses this limitation.

implement/implementing a workspace that uses a host operating system, wherein the workspace enables operation of a first set of one or more applications or processes via a first memory space (claims 1 and 9);

Petitioner relies on Walsh as disclosing this limitation. Pet. 24–25.

The Petition provides:

Walsh’s “user execution environment 220” (“a workspace”) is created by a “secure OS 106” (“a host operating system”) of the “user computing system 102”: “secure OS 106 can be configured to create a user execution environment 220 in the application space 210.” EX1006, 5:28-32; *see also* EX1006, FIG. 1. “The user execution environment 220” (“workspace”) “enables operation of a first set of one or more applications or processes” because Walsh discloses that the “user execution environment 220 allows users to interact with the user computing system 102 to access content such as run application and programs, view files, etc.” EX1006, 5:30-32. The “user execution environment 220” (“workspace”) is supported by its own “user namespace such as home directory and temporary directory” (“a first memory space”). EX1006, 5:33-40. The “sandbox tool 110” is configured to “allocate resources (storage space, memory, etc.) of the user computing system 102, which are necessary to create the isolated execution environment.” *Id.*, 4:35-38; EX1003, ¶¶156-157.

Id. at 24. The cited passages of Walsh state:

In particular, the sandbox tool 110 can be configured to allocate resources (storage space, memory, etc) of the user computing system 102, which are necessary to create the isolated execution environment, . . .

During operation, the secure OS 106 can be configured to create a user execution environment 220 in the application space 210. The user execution environment 220 allows users to interact with the user computing system 102 to access content such as run application and programs, view files, etc.

Ex. 1006, 4:35–38, 5:28–32. The cited passages in Walsh support Petitioner’s contention that Walsh discloses this limitation.

implement/implementing an isolated computing environment that uses the host operating system, (claims 1 and 9)

Petitioner relies on Walsh as disclosing this limitation. Pet. 25–26.

The Petition provides:

Walsh discloses an “isolated execution environment 230” (“*isolated computing environment*”) that is created by the “secure OS 106” (“*host operating system*”): “[S]andbox tool 110 can be configured to cooperate with components of a secure OS 106, to create an **isolated execution environment 230**” EX1006, 5:50-57; EX1003, ¶¶158-159.

Id. The cited passage in Walsh states:

In embodiments, in order to provide access to content 225 without endangering the user computing system 102, the sandbox tool 110 can be configured to cooperate with components of a secure OS 106, to create an isolated execution environment 230 for accessing content 225 (trusted or untrusted) without exposing other processes such as the user execution environment 220 and resources of the user computing system 102 to the content 225.

Ex. 1006, 5:50–57. The cited passages in Walsh support Petitioner’s contention that Walsh discloses this limitation.

wherein the isolated computing environment enables operation of a second set of one or more applications or processes via a second memory space, and (claims 1 and 9)

Petitioner relies on Walsh as disclosing this limitation. Pet. 26–28.

The Petition provides:

Walsh’s “isolated execution environment 230” (“*isolated computing environment*”) executes applications and programs (“*a second set of one or more applications or processes*”) that require isolation. EX1006, 4:47-52 (“[S]andbox tool 110 can be configured to create the isolated execution environment to allow the **applications, programs** . . . to be . . . **executed**”), 10:18-21 (“[C]ontent can be any content (files, **applications, programs**) which is required by the isolated execution environment 230”), 10:21-23 (“[C]ontent 225 can be **application programs**, files, data, etc. . . .”).

Walsh discloses “storage space” and “memory” (“*a second memory space*”) that is allocated to the “isolated execution environment 230” by a “sandbox tool 110.” EX1006, 4:35-42, 5:57-61. The “isolated execution environment 230 can only access the resources” (e.g., the “memory”) allocated to it. EX1006, 5:61-65. In particular, the “sandbox tool 110” creates “a namespace for the isolated execution environment 230 (e.g., home directory and tmp directory) which is separate from the namespace of the user execution environment 220.” EX1006, 6:4-9; EX1003, ¶¶160-163.

Id. at 27. The cited passages in Walsh state:

In particular, the sandbox tool 110 can be configured to allocate resources (storage space, memory, etc) of the user computing system 102, which are necessary to create the isolated execution environment, and apply security policies of the secure OS 106 to the isolated execution environment such that content running in the isolated execution environment can only access the resources allocated to the isolated execution environment. . . . The sandbox tool 110 can be configured to create the isolated execution environment to allow the applications, programs, files, and/or data to be accessed, executed, or viewed without

exposing the user computing system 102 to any possible malicious, dangerous, or damaging actions of the content. . . . In particular, the sandbox tool 110 can be configured to allocate resources (storage space, memory, etc) of the user computing system 102, which are necessary to create the isolated execution environment 230. The sandbox tool 110 can be configured to apply security polices of the secure OS 106 to the isolated execution environment 230 such that the content 225 running in the isolated execution environment 230 can only access the resources allocated to the isolated execution environment 230. . . .

In embodiments, to create the isolated execution environment 230, the sandbox tool 110 can be configured to cooperate with the secure OS 106 to create a namespace for the isolated execution environment 230 (e.g. home directory and tmp directory) which is separate from the namespace of the user execution environment 220. . . .

The necessary content can be any content (files, applications, programs) which is required by the isolated execution environment 230 to function similarly as the user execution environment 220. The content 225 can be application programs, files, data, etc. that the user specified in the request.

Ex. 1006, 4:35–42, 4:47–52, 5:57–65, 6:4–9, 10:18–23. The cited passages in Walsh support Petitioner’s contention that Walsh discloses this limitation.

wherein the second set of one or more applications or processes comprises a multi-user interactive software application; (claims 1 and 9)

Petitioner contends that “Walsh alone or in combination with AAPA discloses this element.” Pet. 28. The Petition provides:

Walsh discloses that the content executing in the isolated execution environment 230 (“*the second set of one or more applications or processes*”) may include “any number and type of data, applications programs such as word processing applications, **web browser applications**, file sharing applications, **electronic mail (e-mail) applications**, multimedia applications, **chat applications**, etc.” EX1006, 3:67-4:4. A

chat application, as disclosed by Walsh, is a “*multi-user interactive software application*.” EX1003, ¶165.

In addition, the ’601 patent admits that the claimed “*multi-user interactive software application*” existed in commercially available products such as WebEx. EX1001, 4:45-49. And a POSA would have understood that WebEx could be used (i) within a web browser, (ii) as a standalone application, or (iii) as an email client plugin. EX1003, ¶166.

As explained above, a POSA would have been motivated, with a reasonable expectation of success, to execute WebEx (“*multi-user interactive software application*”) as one of Walsh’s untrusted applications in the isolated execution environment 230 so that WebEx could not “access files and processes of any other namespace.” EX1006, 6:23-33. This combination, moreover, would have amounted to nothing more than combining familiar elements (e.g., Walsh’s content isolation and the commercially available WebEx application) according to their known methods of use. *KSR Int’l Co. v. Teleflex Inc.*, 550 U.S. 398, 416 (2007).

In the Walsh-AAPA combined system, Walsh’s untrusted application (“*the second set of one or more applications or processes*”) would be WebEx (“*a multiuser interactive software application*”). EX1003, ¶¶167-168.

Id. at 28–29. The cited passage in the ’601 patent states:

The collaboration software may be implemented using software that is commercially available, for example, such as WebEx, GoToMeeting, etc.

The terms “collaboration software” and “multi-user interactive software” may be used interchangeably herein.

Ex. 1001, 4:45–49. The cited passages in Walsh state:

The content can include any number and type of data, applications programs such as word processing applications, web browser applications, file sharing applications, electronic mail (e-mail) applications, multimedia applications, chat applications, etc. . . .

In embodiments, the sandbox tool 110 can be configured to cooperate with the secure OS 106 to isolate the isolated execution environment 230 from the user execution environment 220. For example, the sandbox tool 110 can instruct the secure OS 106 to unshare any other namespaces (e.g. the user name space of the user execution environment 220) with the separate namespace of the isolated execution environment and to mount the separate namespace over any existing namespaces so that the processes of isolated execution environment 230 cannot access files and processes of any other namespace except the separate namespace.

Ex. 1006, 3:67–4:4, 6:23–33.

In the '601 patent, the Applicant admits that WebEx is prior art “*multi-user interactive software*.” The cited passages in Walsh support Petitioner’s contentions regarding this limitation. We determine that Petitioner has established that the prior art discloses this limitation.

isolate/isolating the isolated computing environment from the workspace using an internal isolation firewall; (claims 1 and 9)

Petitioner relies on Walsh as disclosing this limitation. Pet. 30. The Petition provides:

Walsh discloses a “sandbox tool 110” that implements “security policies” (“*internal isolation firewall*”) to isolate the “isolated execution environment 230” (“*isolated computing environment*”) from the “user execution environment 220” (“*workspace*”):

[S]andbox tool 110 can be configured to cooperate with components of a secure OS 106, to create an isolated execution environment 230 for accessing content 225 (trusted or untrusted) **without exposing other processes such as the user execution environment 220 and resources of the user computing system 102 to the content 225.**

EX1006, 5:50-57.

In particular, the sandbox tool 110 is “configured to apply security policies of the secure OS 106 to the isolated execution environment 230 such that **the content 225 running in the isolated execution environment 230 can only access the resources allocated to the isolated execution environment 230.**” EX1006, 5:61-65.

Id. The cited passages in Walsh support Petitioner’s contention that Walsh discloses this limitation.

authenticate/authenticating the isolated computing environment with an authentication device; and (claims 1 and 9)

Petitioner contends that “[t]he Walsh-AAPA-Innes combination discloses this element.” Pet. 30. The Petition provides:

A POSA would have been motivated to combine Walsh-AAPA with Innes to securely access content on the network disclosed by Walsh. Specifically, Walsh notes that in distributed computing environments local computing systems may be connected to remote computing systems via a network. EX1006, 3:24-54. Walsh says that the local computing systems can access content via the network using “a client-server model.” EX1006, 4:66-5:3. To the extent that Walsh does not provide implementation details about the client-server model, Innes does.

Like Walsh, Innes discloses that remote computers can access content via a network “using a web browser.” EX1012, 4:27-47, FIG. 1. The remote computers “may wish to access enterprise resources that require authentication.” EX1012, 23:43-45. To provide such authentication, Innes discloses a “proxy device 710.” And Innes discloses “example method steps for authentication” and for “providing secured access to resources using a proxy,” such as “proxy device 710.” EX1012, 27:43-44, FIGs. 9A-C.

A POSA would have been motivated, with a reasonable expectation of success, to implement the Walsh-AAPA user computing system on a network including Innes’s proxy device and authentication services in order to securely access WebEx content on Walsh’s network. EX1003, ¶176. Walsh discloses

that computing devices may implement content isolation by using a “user execution environment 220” (for trusted content) and an “isolated execution environment 230” (for untrusted content). EX1006, FIG. 2. The computing devices access the trusted/untrusted content over a network using a web browser. EX1006, 3:64-4:4. WebEx is used within a web browser. EX1003, ¶176. The content, Innes teaches, can be “securely” accessed “using a proxy device.” EX1012, 1:6-8. And Innes’s “proxy device” allows computing devices to request certain authentication services, which “may be provided by an authentication server.” EX1012, 27:43-61. Moreover, this combination would have been obvious because it would amount to nothing more than combining familiar elements according to their known methods of use. *KSR*, 550 U.S. at 416.

In addition, the combination of Walsh and Innes would have been obvious to try. EX1003, ¶177. In the client-server context disclosed by Walsh, EX1006, 4:66-5:3, machines can access the Internet using known paradigms, for example, using a network-address-translation device or using a proxy. EX1003, ¶177. Thus, it would have been obvious for a POSA to try to connect Walsh’s computing system to the Internet via a proxy device as disclosed by Innes. *Id.*

In the Walsh-AAPA-Innes system, Walsh’s “isolated execution environment 230” (“*isolated computing environment*”) would be authenticated via Innes’s authentication server (“*authentication device*”).

Id. at 31–32. The cited passages in Walsh state:

FIG. 1 illustrates an exemplary network 100 of computing systems, according to various embodiments. It should be readily apparent to those of ordinary skill in the art that the network 100 depicted in FIG. 1 represents a generalized schematic illustration and that other components may be added or existing components may be removed or modified. Moreover, the network 100 may be implemented using software components, hardware components, or combinations thereof.

As illustrated in FIG. 1, the network 100 can represent the systems of private entities, such as individuals, businesses, partnerships, companies, corporations, etc., and public entities,

such as universities, governmental agencies, etc. The network 100 can include a number of computing systems, such as a user computing system 102 and remote computing systems 104. The computing systems, such as the user computing system 102 and remote computing systems 104, can include hardware resources, such as processors, memory, network hardware, storage devices, and the like, and software resources, such as operating systems (OS), application programs, and the like.

The user computer system 102 can include a secure OS 106, such as security enhanced Linux ("SELinux"), available from Red Hat™, Inc. In this example, SELinux implements a monolithic kernel which is configured to provide an X-Window computing environment to the user computing system 102. SELinux is a version of Linux that integrates FLASK architectural components to provide general support for the enforcement of many kinds of mandatory security policies, including those based on the concepts of type enforcement, role-based access control ("RBAC"), and multi-level security ("MLS"). . . .

As such, the user computing system 102 can access the secure OS 106 and the sandbox 110 stored on the one or more remote computing system 104 via the network 108 in order to access content using a client-server model.

Ex. 1006, 3:24–54, 4:66–5:3. The cited passages in Innes state:

Aspects described herein generally relate to providing approaches for a client device to securely access resources using a proxy device. . . .

The components may include data server 103, web server 105, and client computers 107, 109. Data server 103 provides overall access, control and administration of databases and control software for performing one or more illustrative aspects describe herein. Data server 103 may be connected to web server 105 through which users interact with and obtain data as requested. Alternatively, data server 103 may act as a web server itself and be directly connected to the Internet. Data server 103 may be connected to web server 105 through the network 101 (e.g., the Internet), via direct or indirect

connection, or via some other network. Users may interact with the data server 103 using remote computers 107, 109, e.g., using a web browser to connect to the data server 103 via one or more externally exposed web sites hosted by web server 105. Client computers 107, 109 may be used in concert with data server 103 to access data stored therein, or may be used for other purposes. For example, from client device 107 a user may access web server 105 using an Internet browser, as is known in the art, or by executing a software application that communicates with web server 105 and/or data server 103 over a computer network (such as the Internet). . . .

FIGS. 9A-C are flowcharts of example method steps for authentication and/or providing secured access to resources using a proxy. FIGS. 9A-C illustrate each step being performed by one of the client device 705, proxy device 710, authentication service(s) 715, and/or resource(s) 720. However, the method steps may be performed by any of the client device 705, proxy device 710, authentication service(s) 715, resource(s) 720, and/or a combination thereof. As previously noted, the resources 720 and authentication services 715 may be provided by the same server (or group of servers). Alternatively, the resources 720 and authentication services 715 may be provided by different servers (or group of servers). For simplicity, some of the method steps are illustrated in FIGS. 9A-C as being performed by a single resource/authentication service entity (e.g., a server or group of servers). However, the authentication services may be provided by an authentication server (or groups of authentication servers) and the resources may be provided by a different server (or group of servers).

Ex. 1012, 1:6–8, 4:27–47, 27:43–61.

We determine that Petitioner has established that the prior art discloses this limitation.

send/sending data to an untrusted destination from the multi-user interactive software application via a proxy device when

*the isolated computing environment has been authenticated.
(claims 1 and 9)*

Petitioner contends that “[t]he Walsh-AAPA-Innes combination discloses this element.” Pet. 33. The Petition provides:

Walsh recognizes that untrusted tasks may be accessed over the Internet using a web browsing. EX1006, 1:12-25. To address this issue, Walsh discloses that a web browser may be isolated when instructed to view (“*send data to*”) an untrusted website (“*an untrusted destination*”):

As illustrated in FIG. 5B . . . , the sandbox tool 110 can display a **user interface 535 for the first isolated execution environment . . .** in the desktop GUI 505. The user interface 535 can include various menus and widgets for accessing application programs, such as a tool bar 550 and application icon 555 for accessing **a web browser application program.**

EX1006, 12:22-29, FIG. 5B

As shown in Figure 7 . . . , Innes discloses “a proxy device 710” (“*proxy device*”) that provides a “client device 705” access to “resources 720, such as documents, emails, services, files, and the like.” EX1012, 22:13-15. . . .

As explained above, a POSA would have been motivated, with a reasonable expectation of success, to implement the Walsh-AAPA user computing system on a network including Innes’s proxy device and authentication services in order to securely access WebEx content on Walsh’s network. EX1003, ¶182. Walsh discloses that computing devices may implement content isolation by using a “user execution environment 220” (for trusted content) and an “isolated execution environment 230” (for untrusted content). EX1006, FIG. 2. The computing devices access the trusted/untrusted content over a network using a web browser. EX1006, 3:64-4:4. WebEx is used within a web browser. EX1003, ¶182. The content, Innes teaches, can be “securely” accessed “using a proxy device.” EX1012, 1:6-8. And Innes’s “proxy device” allows computing devices to request certain authentication services, which “may be provided by an

authentication server.” EX1012, 27:43-61. Moreover, this combination would have been obvious because it would amount to nothing more than combining familiar elements according to their known methods of use. *KSR*, 550 U.S. at 416.

In the Walsh-AAPA-Innes system, Walsh’s web browser would access WebEx (“*multi-user interactive software*”) to communicate with (“*send data to*”) an untrusted website (e.g., a WebEx server) (“*an untrusted destination*”) when Walsh’s “isolated execution environment 230” (“*isolated computing environment*”) has been “*authenticated*” by Innes’s authentication server (“*authentication device*”).

Id. at 33–36. The cited passages in Walsh state:

In today's distributed computing environments, security is of the utmost importance. Due to the rise of wide-area public networks, users have unlimited access to content, e.g. data, files, applications, programs, etc., from a variety of sources. Often, the users are unaware of the origin of the content available in the public networks. Malicious entities utilize this ease of accessibility and anonymity to attack the users. For example, the malicious entities can plant viruses, Trojans, or other malicious agents in publicly available content in order to attack the users' computing systems and steal sensitive information from the users. As such, the users must treat content from unknown sources as untrusted and possibly dangerous. . . .

As illustrated in FIG. 5B, the sandbox tool 110 can display a user interface 535 for the first isolated execution environment and a user interface 540 for the second isolated execution environment in the desktop GUI 505. The user interface 535 can include various menus and widgets for accessing application programs, such as a tool bar 550 and application icon 555 for accessing a web browser application program.

Ex. 1006, 1:12-25, 12:22–29. Innes states:

The client device 705 may send a request for resources 720, such as documents, emails, services, files, and the like, to the proxy device 710.

Ex. 1012, 22:13–15.¹⁸

We determine that Petitioner has established that the prior art discloses this limitation.

c. Summary as to Independent Claims 1 and 9

Petitioner has shown that the combination of Walsh, AAPA, and Innes teaches or suggests all the limitations of claims 1 and 9. Petitioner provides articulated reasoning with rational underpinnings to support combining the relevant teachings of Walsh, AAPA, and Innes. As noted previously, Patent Owner does not dispute any aspect of Petitioner’s showing as to unpatentability and does not provide any contrary arguments or evidence. We determine that there is a reasonable likelihood that Petitioner will prevail with respect to the unpatentability of claims 1 and 9.¹⁹

4. Dependent Claims 2–4, 7, 10–12, and 15

Petitioner also asserts that dependent claims 2–4, 7, 10–12, and 15 are obvious in view of a combination of the teachings of Walsh, AAPA, and Innes. Pet. 36–41. The Petition provides a detailed and supported showing

¹⁸ The second and third cited passages in Innes (Ex. 1012, 1:6–8, 27:43–61) were quoted above in our analysis of the immediately preceding limitation and are not quoted again here.

¹⁹ Thus, we determine that there is a reasonable likelihood that the Petitioner will prevail with respect to at least one of the claims challenged in the Petition pursuant to 35 U.S.C. § 314 and that *inter partes* review should be instituted. Accordingly, we institute as to all the challenged claims and all the challenges raised in the Petition. 37 C.F.R. §42.108(a) (“When instituting *inter partes* review, the Board will authorize the review to proceed on all of the challenged claims and on all grounds of unpatentability asserted for each claim.”).

that the additional limitations recited in dependent claims 2–4, 7, 10–12, and 15 are taught by Walsh, AAPA, and Innes. *Id.*

We determine that there is a reasonable likelihood that Petitioner will prevail with respect to the unpatentability of claims 2–4, 7, 10–12, and 15.

E. Asserted Obviousness of Dependent Claims 5, 6, 13, and 14 Based on Walsh, AAPA, Innes, and Netfilter

Petitioner challenges dependent claims 5, 6, 13, and 14 as being obvious over Walsh, AAPA, Innes, and Netfilter. *See* Pet. 12, 41–44.

1. Netfilter (Ex. 1013)

Netfilter relates to “[t]he netfilter.org project,” and discusses “software of the packet filtering framework inside the Linux 2.4.x and later kernel series.” Ex. 1013, 1. Netfilter states that “[s]oftware commonly associated with netfilter.org is iptables.” *Id.* Netfilter indicates that “netfilter/iptables” may be employed to, among other things, “build internet firewalls based on stateless and stateful packet filtering” and “deploy highly available stateless and stateful firewall clusters.” *Id.*

2. Dependent Claims 5, 6, 13, and 14

The Petition provides a detailed and supported showing that the additional limitations recited in dependent claims 5, 6, 13, and 14 are taught by Walsh, AAPA, Innes, and Netfilter. Pet. 42–44. Petitioner articulates with supporting reasoning and evidence why a skilled artisan would have been motivated to combine the relied-on teachings of Walsh, AAPA, Innes, and Netfilter. *Id.* at 41–42.

We determine that there is a reasonable likelihood that Petitioner will prevail with respect to the unpatentability of claims 5, 6, 13, and 14.

F. Asserted Obviousness of Dependent Claims 8 and 16 Based on Walsh, AAPA, Innes, and Franco

Petitioner challenges dependent claims 8 and 16 as being obvious over Walsh, AAPA, Innes, and Franco. *See* Pet. 12, 44–48.

1. Franco (Ex. 1009)

Franco is titled “Running Internet Applications with Low Rights.” Ex. 1009, code (54). Franco teaches “applications that are configured to interact with the Internet in some way are executed in a restricted process with a reduced privilege level that can prohibit the application from accessing portions of an associated computing device.” *Id.* at code (57).

According to Franco, “through various interactions that can take place between an application and the Internet, so called malware or spyware can get downloaded on [a] user’s system and can adversely impact the system’s performance and, perhaps more importantly, can impermissibly install malicious software.” *Id.* at 1:22–27. As examples, Franco states that “buffer overruns and other security holes can allow malware to maliciously make its way onto a user’s system.” *Id.* at 1:27–29.

Franco discloses “applications that are configured to interact with the Internet . . . executed in a restricted process with a reduced privilege level that can prohibit the application from accessing portions of an associated computing device.” *Id.* at 2:32–36. Franco teaches that “the restricted process can prohibit applications from read and write access (or at least write access) to portions of a system’s computer-readable media, such as the hard disk.” *Id.* at 2:36–39. According to Franco, “a special portion of the disk, termed a ‘containment zone’, is designated and used by applications in this restricted process.” *Id.* at 2:41–43. The application has full access to the ‘containment zone’ and the ‘containment zone’ is treated as untrusted

data by the rest of the system and applications on the system.” *Id.* at 2:43–46.

Franco further teaches a broker mechanism that acts to broker access to the restricted area and to ensure that the user is aware of and can approve the application’s access to the restricted areas of the computing system. *Id.* at 2:51–54. According to Franco, “[i]n at least some embodiments, explicit user interaction is needed in order to gain access to the restricted area.” *Id.* at 2:54–56.

2. Dependent Claims 8 and 16

The Petition provides a detailed and supported showing that the additional limitations recited in dependent claims 8 and 16 are taught by Walsh, AAPA, Innes, and Franco. Pet. 46–48. Petitioner articulates with supporting reasoning and evidence why a skilled artisan would have been motivated to combine the relied-on teachings of Walsh, AAPA, Innes, and Franco. *Id.* at 44–46.

We determine that there is a reasonable likelihood that Petitioner will prevail with respect to the unpatentability of claims 8 and 16.

G. Asserted Obviousness of Claims 1–4, 7, 9–12, and 15 Based on Austin, AAPA, and Bloch

Petitioner challenges claims 1–4, 7, 9–12, and 15 as being obvious over Austin, AAPA, and Bloch. *See* Pet. 12, 48–76.

1. Austin (Ex. 1005)

Austin is titled “Computer Device and Method for Isolating Untrusted Content on a Clipboard.” Ex. 1005, code (54). Austin relates to a “computer device and respective method [that] provides a primary clipboard accessible from a primary user account, while a sandbox is used to isolate and contain a secondary user account.” *Id.* at code (57). “A secondary

clipboard is provisioned and associated with the secondary user account,” and “[t]he computer device . . . intercepts requests from the secondary user account such as for cut, copy or paste type clipboard operations which are ordinarily directed toward the primary clipboard, and satisfies those clipboard operation requests instead by using the secondary clipboard.” *Id.*

Figure 4 of Austin, reproduced below, illustrates a schematic view of an example computer device. *Id.* ¶ 45.

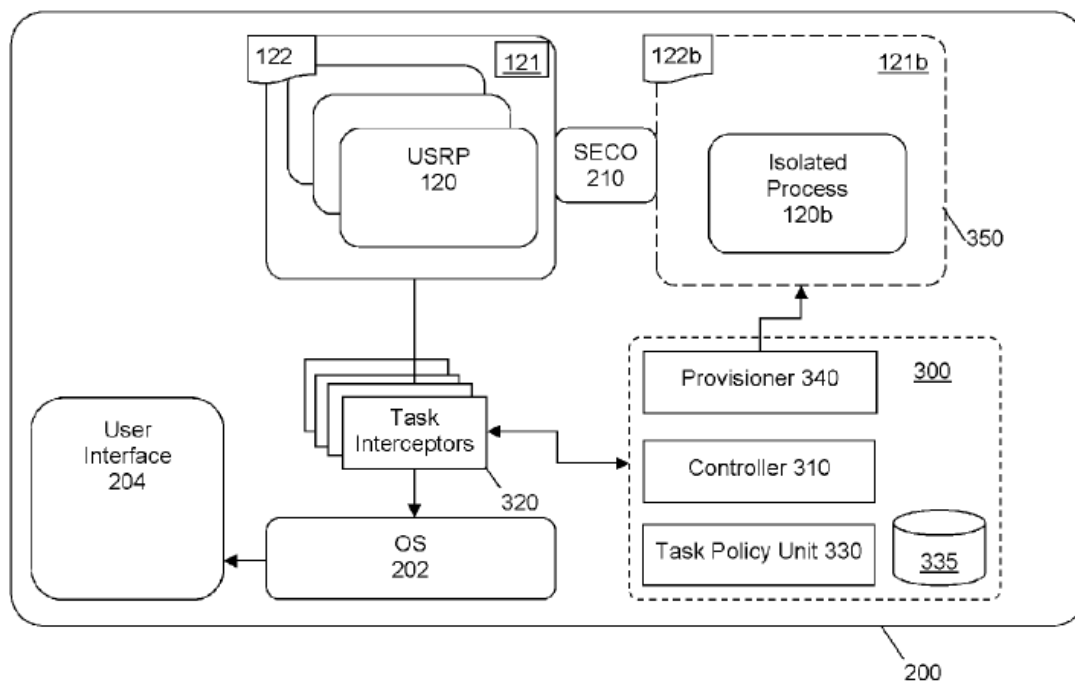


Fig. 4

Figure 4 depicts a computer device 200 including an operating system 202, a user interface 204, and a content isolation agent 300. *Id.* ¶¶ 70–71, 74. As shown in figure 4, the content isolation agent 300 includes a controller

unit 310, one or more task interceptor units 320, a task policy unit 330, and a provisioner unit 340. *Id.* ¶ 74. According to Austin,

Where the controller 310 determines that the task in question relates to untrusted content, i.e. is determined to be an untrusted task, then the controller 310 causes the untrusted task to be opened in a task isolation environment or sandbox 350. The controller 310 may cause a task isolation environment 350 to be provisioned. The isolation environment 350 may be configured based on configuration settings provided by the policy unit 330. Suitably, the policy unit 330 stores policies relevant to each identifiable task which is capable of being intercepted by the task interceptor units 320. Thus, the computer device 200 is capable of implementing different types of task isolation environments, each having different configurations, based on the policy settings.

Id. ¶ 86. Austin explains that the untrusted task runs in a temporary user account and accesses the profile of the temporary user for both data access and registry access. *Id.* ¶ 91. According to Austin, “[a]ny malicious activity only affects the temporary user account . . . and does not affect the primary user account.” *Id.*

Austin teaches an example computer device adapted in order to provide isolation for content on a clipboard. *Id.* ¶ 100. According to Austin, the computer device includes a primary clipboard that performs “clipboard operations such as ‘cut’, ‘copy’, ‘paste’, etc.” *Id.* ¶ 102. Austin further teaches a secondary clipboard provided to be used in the context of the secondary user account by the isolated application. *Id.* ¶ 103. Austin indicates that “this clipboard mechanism maintains security and isolation of potentially malicious code by being able to confine potentially insecure clipboard operation to remain solely within a respective security context. *Id.* ¶ 105.

2. Bloch (Ex. 1007)

Bloch is titled “Apparatus for Monitoring Network Traffic.” Ex. 1007, code (54). Bloch relates to “[a] data processing apparatus [that] can perform HTTP traffic monitoring and filtering of HTTP requests from clients and responses from servers.” *Id.* at code (57).

Figure 6B of Bloch, reproduced below, illustrates an example deployment topology of a proxy appliance. *Id.* at 1:50–51.

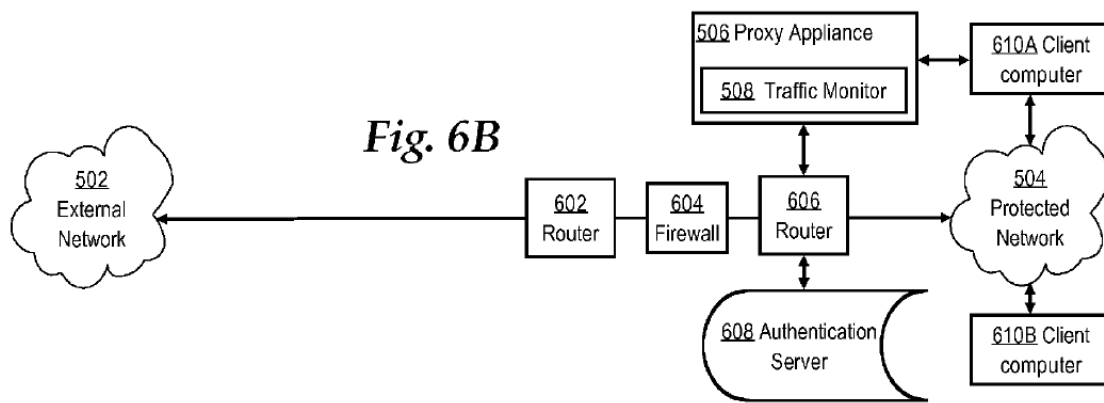


Figure 6B depicts an external network 502 coupled to a proxy appliance 506 through an edge router 602 and firewall 604 that protect a protected network 504. *Id.* at 8:30–32. According to Bloch, “proxy appliance 506 is coupled to an authentication server 608 so that the proxy appliance can provide an integrated user authentication service.” *Id.* at 8:20–23.

According to Bloch, a proxy appliance may receive a client request from a computer located within the protected network that is seeking to retrieve information from an HTTP server, FTP server, or other network resource that is located outside the protected network. *Id.* at 5:21–25. Bloch teaches that “a test is performed to determine whether a user associated with the client computer that initiated the client request has been authenticated,

and whether the user is within a group of users that are authorized to access the requested resource. *Id.* at 5:25–29.

Bloch explains that

If the user is authenticated and the user’s group is authorized, then several checks are made for parameters associated with the client request. In step 406, the client IP address is checked against an administrator blacklist and in step 408 the server IP address is checked against the administrator blacklist. If either IP address is in the blacklist, then control transfers to step 412 in which the “access denied” error page is returned.

In step 410 and step 418, checks are made against administrator whitelists for both the client IP address and server IP address. If either of the IP addresses is in the respective whitelist, then control passes to step 414, in which the client request is allowed and the proxy appliance requests a network resource from a server on behalf of the client. When the response is received, the proxy appliance provides the server response to the client at step 416.

Id. at 5:35–50. Bloch further teaches

the user-agent is checked against a malware blacklist, as some user-agents identify themselves using a known identity, such as Gator, and then the file extension of the requested resource is checked against an administrator blacklist. If the user-agent is in the malware blacklist, or if the file extension is in the administrator blacklist, then access is denied.

Id. at 5:60–66.

3. *Analysis of Independent Claims 1 and 9*

The Petition provides a detailed and supported showing that the limitations recited in independent claims 1 and 9 are taught by Austin, AAPA, and Bloch. Pet. 58–72. Petitioner articulates with supporting reasoning and evidence why a skilled artisan would have been motivated to combine the relied-on teachings of Austin, AAPA, and Bloch. *Id.* at 48–57.

We determine that there is a reasonable likelihood that Petitioner will prevail with respect to the unpatentability of claims 1 and 9.

4. Dependent Claims 2–4, 7, 10–12, and 15

Petitioner also asserts that dependent claims 2–4, 7, 10–12, and 15 are obvious in view of a combination of the teachings of Austin, AAPA, and Bloch. Pet. 72–76. The Petition provides a detailed and supported showing that the additional limitations recited in dependent claims 2–4, 7, 10–12, and 15 are taught by Austin, AAPA, and Bloch. *Id.*

We determine that there is a reasonable likelihood that Petitioner will prevail with respect to the unpatentability of claims 2–4, 7, 10–12, and 15.

H. Asserted Obviousness of Dependent Claims 5, 6, 13, and 14 Based on Austin, AAPA, Bloch, and Walker

Petitioner challenges dependent claims 5, 6, 13, and 14 as being obvious over Austin, AAPA, Bloch, and Walker. *See* Pet. 12, 76–80.

1. Walker (Ex. 1008)

Walker is titled “Method and System for Managing a Host-Based Firewall.” Ex. 1008, code (54). According to Walker, a “‘firewall’ of a device refers generally to firewalling rules to control a flow of information to and from the device.” *Id.* ¶ 66. As an example, Walker teaches a host computing device having an installed operating system and a firewall, “wherein the operating system is a Windows operating system . . . and the firewall is implemented using the Windows Filtering Platform (WFP).” *Id.* ¶ 70.

According to Walker,

The host computing device may be configured to implement firewall rules based on information extracted from the relevant packet. This information may include header information any one or more of the Network layer (layer 3)

header, Transport layer (layer 4) header, Session layer (layer 5) header, Presentation layer (layer 6) header and/or Application layer (layer 7) header.

Id. ¶ 142. For example, Walker teaches that “[o]ne method of firewalling uses specific criteria found in, and below, Layer 4 of the OSI model.” *Id.* ¶ 143. Walker explains that “[i]n one implementation, firewalling controls flow of data based on a source or destination address(es) being used, and/or the destination ports.” *Id.*

2. Dependent Claims 5, 6, 13, and 14

The Petition provides a detailed and supported showing that the additional limitations recited in dependent claims 5, 6, 13, and 14 are taught by Austin, AAPA, Bloch, and Walker. Pet. 78–80. Petitioner articulates with supporting reasoning and evidence why a skilled artisan would have been motivated to combine the relied-on teachings of Austin, AAPA, Bloch, and Walker. *Id.* at 76–78.

We determine that there is a reasonable likelihood that Petitioner will prevail with respect to the unpatentability of claims 5, 6, 13, and 14.

I. Asserted Obviousness of Dependent Claims 8 and 16 Based on Austin, AAPA, Bloch, and Franco

Petitioner challenges dependent claims 8 and 16 as being obvious over Austin, AAPA, Bloch, and Franco. *See* Pet. 12, 81–84.

1. Dependent Claims 8 and 16

The Petition provides a detailed and supported showing that the additional limitations recited in dependent claims 8 and 16 are taught by Austin, AAPA, Bloch, and Franco. Pet. 83–84. Petitioner articulates with supporting reasoning and evidence why a skilled artisan would have been

motivated to combine the relied-on teachings of Austin, AAPA, Bloch, and Franco. *Id.* at 81–82.

We determine that there is a reasonable likelihood that Petitioner will prevail with respect to the unpatentability of claims 8 and 16.

IV. CONCLUSION

For the foregoing reasons, we determine that trial should be instituted on claims 1–16 of the '601 patent.

V. ORDER

Upon consideration of the record before us, it is

ORDERED that, pursuant to 35 U.S.C. § 314(a), an *inter partes* review of claims 1–16 of U.S. Patent No. 11,223,601 B2 is instituted with respect the grounds set forth in the Petition (*see* Pet. 12); and

FURTHER ORDERED that, pursuant to 35 U.S.C. § 314(c) and 37 C.F.R. § 42.4(b), *inter partes* review of U.S. Patent No. 11,223,601 B2 shall commence on the entry date of this Order, and notice is hereby given of the institution of a trial.

IPR2024-01421
Patent 11,223,601 B2

FOR PETITIONER:

Jonathan Tuminaro
Ethan Goldschen
Dan Block
STERNE, KESSLER, GOLDSTEIN & FOX PLLC
jtuminar-ptab@sternekessler.com
egoldschen-ptab@sternekessler.com
dblock-ptab@sternekessler.com

FOR PATENT OWNER:

Brett Cooper
Seth Lindner
BC LAW GROUP, P.C.
bcooper@bc-lawgroup.com
slindner@bc-lawgroup.com

Robert Auchter
AUCHTER PLLC
robert@auchterlaw.com