

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

TREND MICRO, INC.,
Petitioner,

v.

TAASERA LICENSING LLC,
Patent Owner.

IPR2023-00799
Patent 9,092,616 B2

Before NEIL T. POWELL, KRISTI L. R. SAWERT, and
SCOTT RAEVSKY, *Administrative Patent Judges*.

POWELL, *Administrative Patent Judge*.

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

I. INTRODUCTION

A. BACKGROUND

Trend Micro Inc. (“Petitioner”) filed a Petition (Paper 2 (“Pet.”)) requesting an *inter partes* review of claims 1–7 of U.S. Patent No. 9,092,616 B2 (Ex. 1001, “the ’616 patent”). Taasera Licensing LLC (“Patent Owner”) filed a Preliminary Response. Paper 6 (“Prelim. Resp.”).

An *inter partes* review may not be instituted “unless . . . the information presented in the petition . . . shows that there is a reasonable likelihood that the petitioner would prevail with respect to at least 1 of the claims challenged in the petition.” 35 U.S.C. § 314(a). Having considered the arguments and the associated evidence, we determine that Petitioner has not shown a reasonable likelihood that it will prevail in establishing the unpatentability of any of the challenged claims. Accordingly, we do not institute *inter partes* review.

B. RELATED PROCEEDINGS

The parties note that the ’616 patent has been asserted in *In re Taasera Licensing, LLC, Patent Litigation*, MDL No. 3042, consolidated as 2:22-md-03042 (E.D. Tex.). The parties also note that the ’616 patent has been challenged in IPR2023-00706. Pet. 64; Paper 3, 2. Petitioner also identifies IPR2023-00705 and IPR2023-00800 as involving a related patent (U.S. Patent No. 8,990,948). Pet. 64. Patent Owner also identifies the following related district court proceedings: *Taasera Licensing LLC v. Trend Micro Inc.*, No. 2:21-cv-00441-JRG (E.D. Tex.); *Taasera Licensing LLC v. Check Point Software Techs. Ltd.*, No. 2:22-cv-00063-JRG (E.D. Tex.); *Trend Micro Inc. v. Taasera Licensing LLC*, No. 2:22-cv-00303-JRG (E.D. Tex.); *Palo Alto Networks, Inc. v. Taasera Licensing LLC*,

No. 2:22-cv-00314-JRG (E.D. Tex.); *Taasera Licensing LLC v. Fortinet Inc.*, No. 2:22-cv-00415-JRG (E.D. Tex.); *Taasera Licensing LLC v. Musarubra US LLC, d/b/a Trellox*, No. 2:22-cv-00427-JRG (E.D. Tex.); and *Taasera Licensing LLC v. CrowdStrike, Inc., et al.*, No. 2:22-cv-00468-JRG (E.D. Tex.). Paper 3, 2–3.

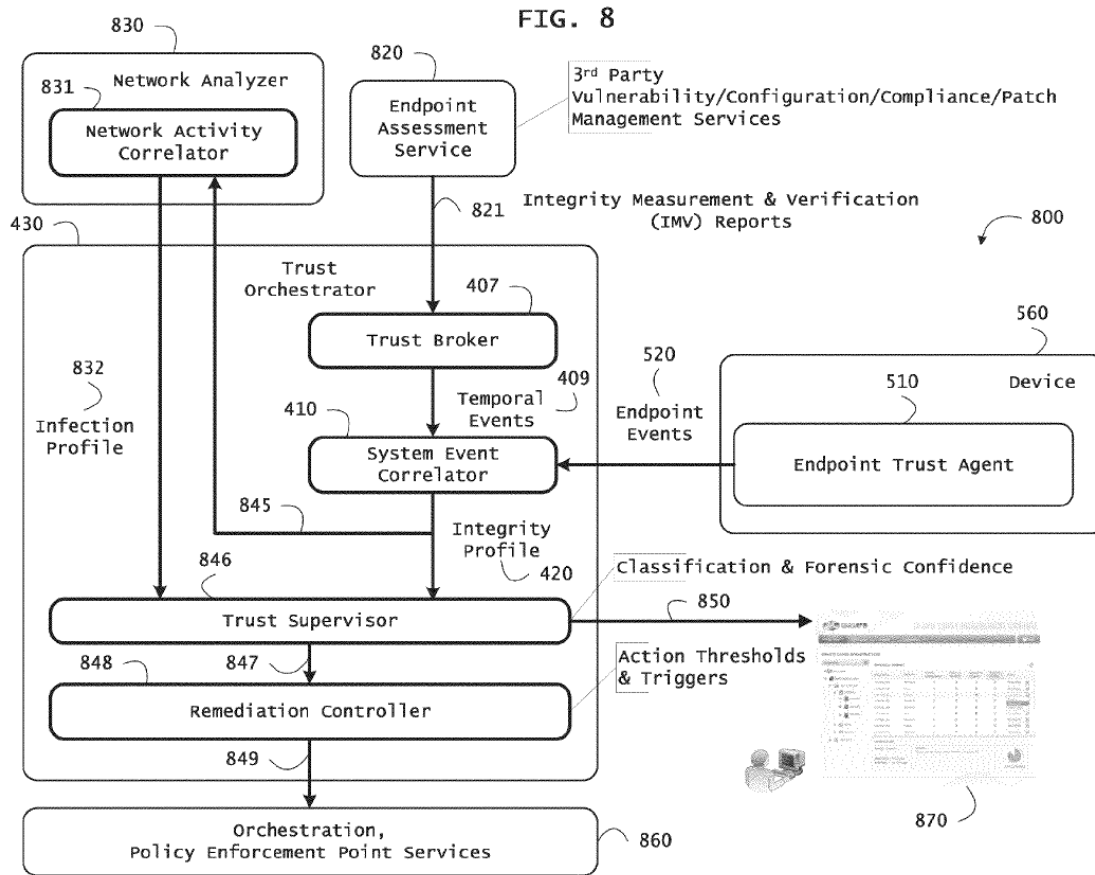
C. REAL PARTIES-IN-INTEREST

Petitioner identifies the following real-parties-in interest: “Trend Micro, Inc., Trend Micro America, Inc. and Trend Micro Incorporated.” Pet. 63. Patent Owner states that “[t]he patent owner and real party-in-interest is Taasera Licensing LLC.” Paper 3, 2.

D. THE ’616 PATENT

The ’616 patent discusses “[s]ystems and methods for threat identification and remediation for computing platforms.” Ex. 1001, code (57). According to the ’616 patent, “virtualized, cloud computing on-demand, and elastic models require dynamic security orchestration at both the device and network level that is lacking in traditional security approaches and technologies.” *Id.* at 2:1–5. The ’616 patent explains that its “presently disclosed exemplary technical solutions may be embodied as a method, apparatus and/or system for integrity confidence measurement.”

Id. at 5:59–61. The '616 patent discusses one embodiment in connection with Figure 8, which is reproduced below.



“[Figure] 8 is a block diagram of an exemplary architecture 800 for trust orchestration including a trust orchestrator 430, a network analyzer 830, and a device 560.” *Id.* at 17:49–51.

Device 560 may be, for example, “a desktop, server, laptop, a mobile computing device such as a smart phone or tablet, or other types of mobile computing platforms.” *Id.* at 13:20–23. “[T]rust orchestration architecture 800 may include an endpoint trust agent 510 on . . . device 560.” *Id.* at 17:57–58.

Trust orchestrator 430 includes “the trust broker 407, system event correlator 410, the trust supervisor 846, and remediation controller 848.”

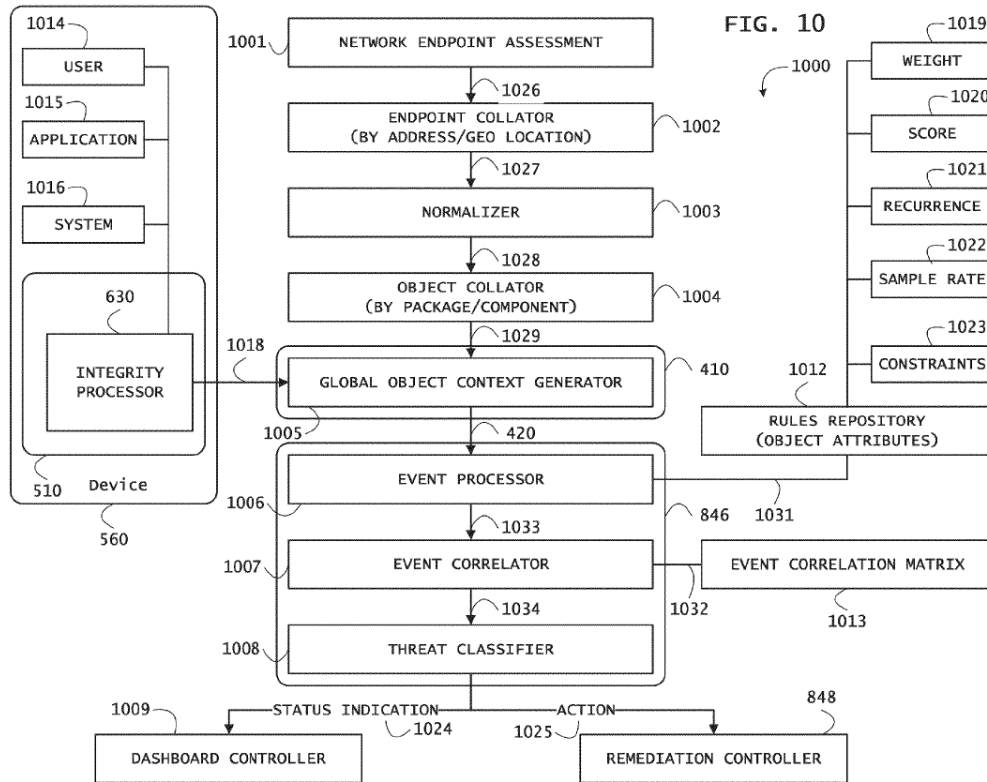
Id. at 17:62–18:3. “In embodiments, the trust orchestrator 430 is configured to receive sensory threat intelligence from network analyzers 830, endpoint assessment services 820 and endpoint trust agents 510 on devices 560.”

Id. at 18:3–6. Events are processed “to initiate remediation actions to deal with identified threats on devices 560.” *Id.* at 18:6–10. For example, “[i]n certain exemplary embodiments, the system event correlator 410 receives endpoint events 520 from the endpoint trust agent 510, temporal events 409 from the trust broker 407, correlates and sends an integrity profile for a network endpoint to the trust supervisor 846 and the network activity correlator 831.” *Id.* at 18:21–26. “The temporal events can be assertions about operational parameters expressed in the Open Vulnerability and Assessment Language (OVAL),” or “the system details can be conveyed using the SCAP protocol.” *Id.* at 11:67–12:6.

With this and other information, trust supervisor 846¹ “analyzes and classifies threats along with a forensic confidence score, sends real time actions 847 to the remediation controller 848, and real time status indications 850 to a dashboard controller 870.” *Id.* at 18:28–35. Remediation controller 848 sends directives 849 for remediation to orchestration and/or policy enforcement point services 860. *Id.* at 18:35–38.

The ’616 patent also discusses evaluating risk in connection with Figure 10, which “is described with continued reference to the embodiment[] illustrated in [Figure] . . . 8.” *Id.* at 22:10–13. Figure 10 is reproduced below.

¹ The cited sentence refers to “trust supervisor 848,” but this appears to be a typographical error. *See, e.g.*, Ex. 1001, 17:63–64 (“a trust supervisor 846, and a remediation controller 848”).



“[Figure] 10 is a block diagram of a system for determining a calculus of risk.” *Id.* at 6:62–64.

Specifically, Figure 10 shows system 1000. *Id.* at 22:15–16. Like Figure 8’s architecture 800, system 1000 includes endpoint trust agent 510, device 560, and trust supervisor 846. *Id.* at 22:15–24, 22:33–36, Fig. 10.

“[T]rust supervisor 846 uses an event processor 1006, an event correlator 1007, a threat classifier 1008, an event correlation matrix 1013 and a rules repository 1012.” *Id.* at 22:37–39.

Similar to architecture 800 using temporal events 409, system 1000 uses temporal events 1029. *Id.* at 22:43–59. In generating temporal events 1029, system 1000 may use network endpoint assessments 1001 and multiple stages of processing with endpoint collator 1002, normalizer 1003, and object collator 1004. *Id.* “In an embodiment, the network endpoint assessment 1001 comprises performing a security scan of network endpoints

for vulnerability, configuration, compliance, and patch management and use the results of these scans to generate a security assessment report 1026.”

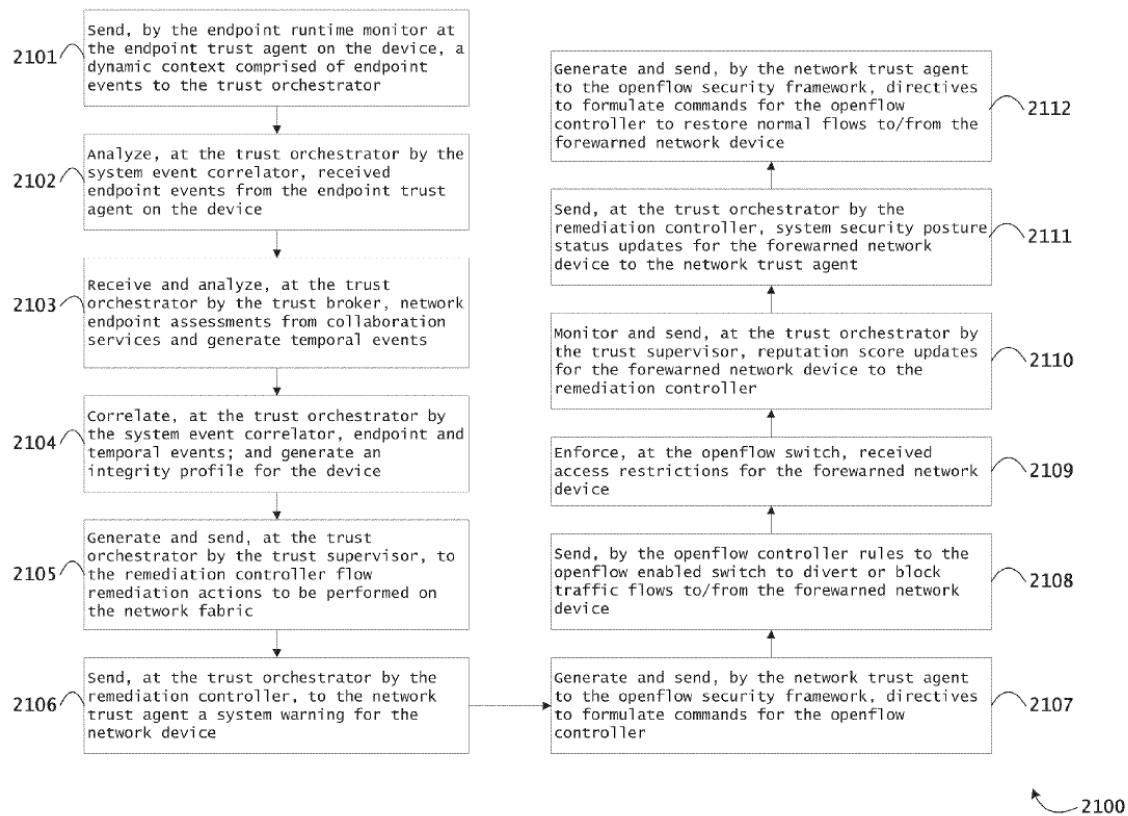
Id. at 22:43–47.

Additionally, similar to architecture 800 using endpoint events 520, system 1000 may use endpoint events 1018: “global object context generator 1005 receives endpoint events 1018 that may include the local runtime object execution context from the integrity processor 630 on a device [560], and temporal events 1029 that may include endpoint security context from the object collator 1004.” *Id.* at 22:60–65.² Using this information, the various components used by trust supervisor 846 may perform certain additional actions and ultimately “categorize threats, send status indications 1024 to the dashboard controller 1009 for real time visibility of endpoint integrity, and send action requests 1025 to the remediation controller 848.” *Id.* at 22:66–23:13.

The ’616 patent discloses one exemplary method in connection with Figure 21. *Id.* at 7:13–17. Figure 21 is reproduced below.

² The quoted sentence refers to “device 1017,” but that appears to be a typographical error, given that Figure 10 labels the “device” with reference numeral 560.

FIG. 21



“[Figure] 21 is a flowchart illustrating steps by which an attestation service for runtime operational integrity of systems executing on a computing platform using a network trust agent, endpoint trust agent and trust orchestration server can be provided.” *Id.* at 32:45–49.

The steps shown in Figure 21 are part of method 2100, which “can provide network flow level remediation for an infected device 560.” *Id.* at 32:49–52. As Figure 21 shows, step 2101 involves “[s]end[ing], by the endpoint runtime monitor at the endpoint trust agent on the device, a dynamic context comprised of endpoint events to the trust orchestrator.” *Id.* at Fig. 21, 32:56–59. Step 2102 involves “[a]nalyz[ing], at the trust orchestrator by the system event correlator, received endpoint events from the endpoint trust agent on the device.” *Id.* at Fig. 21, 32:60–62.

Subsequently, step 2103 involves “[r]eceive[ing] and analyz[ing], at the trust orchestrator by the trust broker, network endpoint assessments from collaboration services,” along with “generat[ing] temporal events.”

Id. at Fig. 21; 32:60–67. “In step 2104, the system event correlator 410 may correlate endpoints 520 and temporal [events 409] and generate an integrity profile 420 for the device 560.” *Id.* at 33:1–3.³ Steps 2105–2108 include sending “remediation actions,” “a system warning,” “directives,” and “rules.” *Id.* at Fig. 21, 33:4–20. Step 2109 involves “[e]nforc[ing] . . . access restrictions.” *Id.* at Fig. 21, 33:21–25. Steps 2110–2112 involve sending “reputation score updates,” “system security posture status updates,” and “directives.” *Id.* at 33:26–39.

E. ILLUSTRATIVE CLAIM

Of the challenged claims, claim 1 is independent. Each of claims 2–7 depends, directly or indirectly, from claim 1. Independent claim 1 is reproduced below with Petitioner’s added identifiers bolded and in brackets.

1. **[1[a]]** A method of providing an attestation service for providing runtime operational integrity of a system using a computing platform comprising a network trust agent, an endpoint trust agent, and a trust orchestration server, the method comprising:

[1[b]] sending, by the endpoint trust agent on a monitored device, a dynamic context including endpoint events and actions of the monitored device and applications executing on the monitored device at runtime;

[1[c]] receiving, at the trust orchestration server, the dynamic context including the endpoint events of the monitored device and the applications executing on the monitored device at runtime;

³ The sentence’s recitation of “temporal 409 events” appears to be a typographical error.

- [1[d]] analyzing, by the trust orchestration server, the received endpoint events;
- [1[e]] receiving, by the trust orchestration server, third party network endpoint assessments;
- [1[f]] generating, by the trust orchestration server, temporal events based at least in part on analyzing the third party network endpoint assessments;
- [1[g]] correlating, by the trust orchestration server, the received endpoint events and the generated temporal events; and
- [1[h]] generating, by the trust orchestration server, an integrity profile for the system.

Id. at 39:53–40:9.

F. ASSERTED GROUNDS OF UNPATENTABILITY

Petitioner challenges the patentability of claims 1–7 of the '616 patent on the following grounds (Pet. 3):

Claim(s) Challenged	35 U.S.C. §	References/Basis
1–6	103(a) ⁴	Mukherjee ⁵ , Miliefsky ⁶
7	103(a)	Mukherjee, Miliefsky, Guo ⁷

Petitioner also submits the Declaration of Seth James Nielson, Ph.D.
Ex. 1003.

⁴ The Leahy-Smith America Invents Act (“AIA”), Pub. L. No. 112-29, 125 Stat. 284, 287–88 (2011), revised 35 U.S.C. § 103, effective March 16, 2013. The application that resulted in the '616 patent was filed before March 16, 2013, so we apply the pre-AIA version of § 103.

⁵ Mukherjee, et al., *Network Intrusion Detection*, IEEE Network, published May/June 1994 (Ex. 1005, “Mukherjee”).

⁶ Miliefsky, U.S. Patent Application Publication No. 2007/0192867 A1, published Aug. 16, 2007 (Ex. 1006, “Miliefsky”).

⁷ Guo, et al., U.S. Patent No. 8,104,089 B1, issued Jan. 24, 2012 (Ex. 1011, “Guo”).

II. ANALYSIS

A. LEVEL OF ORDINARY SKILL IN THE ART

Petitioner contends that an ordinarily skilled artisan for the '616 patent “would have had a bachelor’s degree in computer science, or a related field, or an equivalent technical degree or equivalent work experience, and an additional one to two years of education or experience in the field of computer security topics.” Pet. 10 (citing Ex. 1003 ¶ 83). Petitioner adds that “[m]ore education can supplement practical experience and vice versa.” *Id.* Patent Owner states that it “utilizes Petitioner’s proposed level of skill in the art.” Prelim. Resp. 13.

Based on this record, we adopt Petitioner’s articulation of the level of ordinary skill in the art, which appears to be consistent with the '616 patent and the asserted prior art, and we apply it in our analysis. *See Okajima v. Bourdeau*, 261 F.3d 1350, 1355 (Fed. Cir. 2001) (the prior art, itself, can reflect appropriate level of ordinary skill in art).

B. CLAIM CONSTRUCTION

We apply the same claim construction standard used in district courts, namely that articulated in *Phillips v. AWH Corp.*, 415 F.3d 1303 (Fed. Cir. 2005) (en banc). *See* 37 C.F.R. § 42.100(b) (2022). In applying that standard, claim terms generally are 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 and in the context of the entire patent disclosure. *Phillips*, 415 F.3d at 1312–13.

Both parties express that we do not need to construe any claim language explicitly. Pet. 10–11; Prelim. Resp. 11. Having considered the record, we agree with the parties that no explicit claim construction is

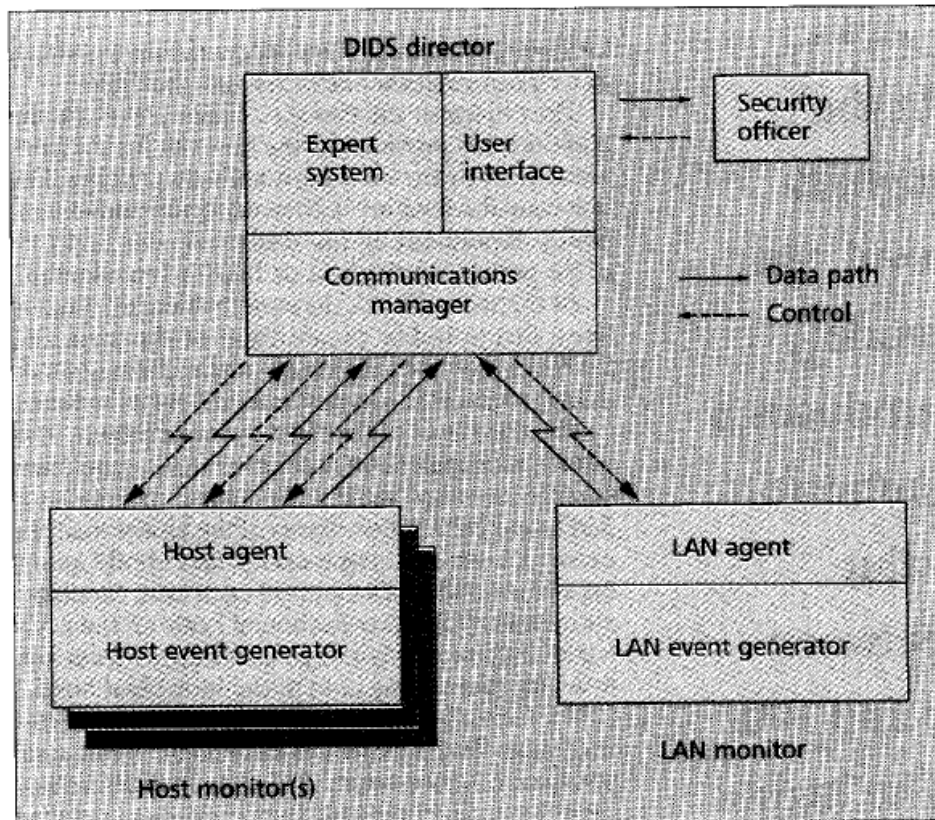
necessary for this Decision. *See Nidec Motor Corp. v. Zhongshan Broad Ocean Motor Co.*, 868 F.3d 1013, 1017 (Fed. Cir. 2017) (holding that only claim terms in controversy need to be construed, and only to the extent necessary to resolve the controversy (citing *Vivid Techs., Inc. v. Am. Sci. & Eng'g, Inc.*, 200 F.3d 795, 803 (Fed. Cir. 1999))).

C. GROUND 1 – ALLEGED OBVIOUSNESS OVER MUKHERJEE AND MILIEFSKY

1. *Overview of Mukherjee*

Mukherjee discloses that “[i]ntrusion detection is a new, retrofit approach for providing a sense of security in existing computers and data networks.” Ex. 1005, 1. Mukherjee discusses host-based and network-based intrusion detection systems (IDSs). *Id.* Mukherjee explains that “host-based systems employ the host operating system’s audit trails as the main source of input to detect intrusive activity,” while network-based IDSs “build their detection mechanism on monitored network traffic.” *Id.*

Mukherjee also describes a Distributed Intrusion Detection System (DIDS), which performs both host-based and network-based monitoring. *Id.* at 2, 11. Mukherjee discusses such a system in connection with Figure 2, which is reproduced below.



■ **Figure 2.** *Communications architecture.*

Figure 2 of Mukherjee depicts the communications architecture of DIDS. *Id.* at 11–12. As shown in Figure 2, the architecture includes “the DIDS director, a single host monitor per host, and a single LAN monitor for each broadcast LAN segment in the monitored network.” *Id.* at 11. “The host and LAN monitors are primarily responsible for the collection of evidence of unauthorized or suspicious activity,” and the DIDS director “is primarily responsible for its evaluation.” *Id.*

To evaluate the evidence, the DIDS director includes an expert system. *Id.* at 12. “The expert system uses rules derived from a hierarchical Intrusion Detection Model (IDM).” *Id.* “The IDM consists of six layers, with each layer representing the result of a transformation performed on the data.” *Id.* at 12–13. At the first level, objects include audit records supplied “by the host OS, by the LAN monitor, or by any third-party auditing

package,” and “all of the activity on the host or LAN is represented.” *Id.* at 13. “At the second level, the event . . . is both syntactically and semantically independent of the source standard format for events.” *Id.* The third layer “creates a subject,” which “introduces a single identification for a user across many hosts on the network.” *Id.* “The fourth layer of the model introduces the event in context.” *Id.* Mukherjee elaborates that “[t]here are two kinds of context: temporal and spatial.” *Id.* “The fifth layer of the model considers the *threats* to the network and the hosts connected to it. Events in context are combined to create threats.” *Id.* At the IDM’s top level, “the model produces a numerical value between one and 100 which represents the overall security state of the network.” *Id.*

2. *Overview of Miliefsky*

Miliefsky relates to “computer security, and more particularly to a micro-appliance for use in defending against common vulnerabilities and exploits.” Ex. 1006 ¶ 4. Miliefsky describes “security systems that provide proactive automated defense against hackers” “by determining the components that exist on a network system and generating a list of network assets.” *Id.* ¶ 39. Miliefsky discloses a software engine that acts as a gateway and controls certain processes of information gathering:

An Information Security (INFOSEC) software engine acts as a gateway between users, personal computers, servers, services and the computer network (internet, intranet, extranet, wide area network, wireless network or local area network). The information Security (INFOSEC) Engine controls the computer-based network scanning, standards-based vulnerability assessment through common vulnerabilities and exposures (CVEs) testing, reporting and remediation as well as interfacing with the INFOSEC ENGINE PLUGINS.

Id. ¶ 49.

Miliefsky discloses that “the network is scanned and/or probed for any and all attached equipment and related assets.” *Id.* ¶ 40. “The method will dynamically detect and map changes to LAN and WAN connected equipment including searching for equipment which may be deemed as rogue and creating a network-based assets list, wherein the list contains information as to the location of the network-based assets.” *Id.* Miliefsky also describes auditing “network-based assets for common vulnerabilities and exposures (CVEs)” and sharing audit data “with various INFOSEC countermeasures designed to help protect network-based assets against attack.” *Id.* ¶ 43. Miliefsky discusses an example involving an INFOSEC countermeasure that is an IDS:

In the event the INFOSEC countermeasure is an IDS, dynamic alerting of the IT manager or an alternative alert recipient and sharing the related CVE tests data with the IDS to help the IDS reduce false positives in the IDS alerting module as well as reduce the traffic load related to intrusion detections which attack a particular IP address that is not susceptible to that particular attack methodology based upon the related CVE tests data and will take place through the Countermeasure Communications Client plug-in which has been written specifically for that MAKE, MODEL and VERSION IDS.

Id. ¶ 64.

Miliefsky also discloses a clientless network admission control system for controlling devices’ access to networks:

This engine provides a means to control the access of network devices onto networks. The engine does not require any software to be installed on any of the target devices. The engine uses a combination of the network discovery engine, vulnerability assessment engine, database correlation engine, wireless and mobile device detection engine to determine when a network device has permission to access the network. This determination is also based upon information obtained from the regulatory

compliance reviewing and reporting system and policies. This engine interacts with the countermeasure communications system to control the access of each network appliance. The engine is designed to work in a multi-branch solution and provide extensible authorization. It securely connects to firewalls, smart-switches, and VPNs to reconfigure their rules and access control lists around CVE related problems and ports, not people and productivity.

Id. ¶ 94.

Miliefsky discloses that “MITRE’S OVAL standard” “will help accelerate the release of a preferred embodiment of RSS-based Security appliances and services.” *Id.* ¶ 102. “OVAL is the Open Vulnerability Assessment Language,” which is “an international, information security community baseline standard for how to check for the presence of vulnerabilities and configuration issues on computer systems.” *Id.* ¶ 103.

3. Discussion

Petitioner asserts that, in combination, Mukherjee and Miliefsky disclose or teach each limitation of claims 1–6. Pet. 23–56. Petitioner cites Mukherjee’s DIDS system as disclosing many aspects of the challenged claims’ limitations. *See, e.g.*, Pet. 23–33. For example, Petitioner argues that Mukherjee’s DIDS director, host monitor, and LAN monitor correspond to the claimed “trust orchestration server,” “endpoint trust agent,” and “network trust agent,” respectively. *Id.* at 25–26 (citing Ex. 1005, 8, 11–16, Fig. 2; Ex. 1003 ¶¶ 131–132). With respect to limitations 1[b]–1[d], Petitioner asserts that Mukherjee teaches collecting events and actions by host monitors, as well as sending those events to the DIDS director for analysis. Pet. 33–38 (citing Ex. 1005, 2, 11–13; Ex. 1003 ¶¶ 159–174).

Petitioner argues that it would have been obvious to modify Mukherjee’s system to include Miliefsky’s computer security systems to

mitigate and proactively block intruders. Pet. 20–22. Petitioner argues a person of ordinary skill would have been motivated to combine Mukherjee and Miliefsky “to provide compatible systems that detect current threats **and** mitigate future threats.” *Id.* at 20 (citing Ex. 1003 ¶¶ 119–122). According to Petitioner, “Miliefsky describes plugging its device into IDS systems (Mukherjee’s DIDs systems).” *Id.* at 32 (citing Ex. 1003 ¶¶ 117–123, 154). Petitioner cites Miliefsky as teaching certain limitations of the challenged claims, such as the claimed “temporal events.” Pet. 43–44.

Patent Owner argues that “Petitioner fails to identify how certain elements of the Challenged Claims are obvious.” Prelim. Resp. 1. Patent Owner faults Petitioner’s contentions regarding limitations 1[e]–1[g]. *Id.* at 15–21.

Having reviewed all of the parties’ arguments and evidence, we determine Petitioner does not demonstrate a reasonable likelihood of establishing unpatentability of any of claims 1–6. For the reasons that follow, Petitioner does not support sufficiently its positions regarding at least limitations 1[f] and 1[g].

a) Claim 1

(1) limitation 1[f]: “generating, by the trust orchestration server, temporal events based at least in part on analyzing the third party network endpoint assessments.”

Petitioner argues that “Mukherjee in view of Miliefsky discloses” “generating, by the trust orchestration server, temporal events based at least in part on analyzing the third party network endpoint assessments,” as recited in limitation 1[f]. Pet. 43 (citing Ex. 1003 ¶¶ 187–188, 198);

Ex. 1001, 40:3–5. Petitioner asserts that Miliefsky’s “network-based assets list”⁸ constitutes the claimed “temporal events”:

Miliefsky discloses a “network based assets list” that is derived from a network scan to determine the assets and configuration of the network which corresponds to temporal events because the “network based assets list” can be expressed as assertions about operational parameters. EX1003, ¶¶198-199; EX1006, ¶¶[0040]-[0041],[0103]-[0104]. Miliefsky describes using the same open source framework (OVAL) for this collection of configuration data as described by the ’616 patent. EX1001, 11:59-12:16; EX1003, ¶¶200-201. Accordingly, a [person having ordinary skill in the art] would have understood that the OVAL implementation of Miliefsky discloses generating temporal events as claimed. *Id.*

Pet. 43–44. Petitioner further argues that “Miliefsky teaches ‘generating . . . temporal events based at least in part on analyzing the third party network endpoint assessments.’” *Id.* at 44 (citing Ex. 1003 ¶¶ 198–201).

Patent Owner argues that Petitioner is incorrect for multiple reasons. Prelim. Resp. 19–21. According to Patent Owner, we should reject Petitioner’s argument that Miliefsky’s network-based assets list constitutes “temporal events,” as recited in claim 1. *Id.* at 20. Patent Owner also contends that “Petitioner’s argument that the claimed temporal events are disclosed by combining ‘configuration data collected or processed’ with ‘vulnerability assessment data’ or ‘internal assessments’ combined with ‘external assessments’ is incorrect.” *Id.* Patent Owner elaborates that “[c]ombinations of two existing data presupposes that both exist at a given time and, therefore one set of data cannot be based on the other.” *Id.*

⁸ Given Miliefsky’s disclosure of a “network-based assets list,” Petitioner’s reference to a “network based assets list” (with no hyphen) appears to be a typographical error. Compare Ex. 1006 ¶ 40 (reciting “network-based assets list”), with Pet. 43 (reciting “network based assets list”).

Accordingly, Patent Owner argues that Petitioner does not show that Miliefsky meets claim 1's requirement of "generating . . . temporal events based at least in part on analyzing the third party network endpoint assessments." *Id.* at 20–21.

Having reviewed the parties' arguments and evidence, we agree with Patent Owner that Petitioner has not addressed limitation 1[f] persuasively. Pet. 43–44; Prelim. Resp. 19–21. Even assuming Petitioner is correct that Miliefsky's network-based assets list constitutes "temporal events," as recited in claim 1, Petitioner does not persuasively address certain other aspects of the claim language.

For example, Petitioner does not support sufficiently its contention that "Miliefsky teaches 'generating . . . temporal events *based at least in part on analyzing the third party network endpoint assessments.*'" Pet. 44 (citing Ex. 1003 ¶¶ 198–201) (emphasis added). Petitioner cites Miliefsky's paragraph 94, which does not expressly disclose generating the alleged temporal events (the network-based assets list) based at least in part on analyzing third party network endpoint assessments. Ex. 1006 ¶ 94; Pet. 44. According to Petitioner, however, Miliefsky discloses "configuration data collected or processed at particular time ('temporal events') can be combined with vulnerability assessment data and include related data, which . . . is disclosed as being sourced from third parties." Pet. 44 (citing Ex. 1003 ¶ 201). Petitioner also argues that "internal assessments such as the generated network asset list are disclosed as being combined with external assessments to provide feedback to each other (e.g., to produce control lists)." *Id.* (citing Ex. 1003 ¶¶ 202–203; Ex. 1006 ¶ 94).

Even if accurate, these arguments do not show sufficiently for institution that Miliefsky teaches generating temporal events based at least in

part on analyzing third party network endpoint assessments. Instead, these arguments indicate that Miliefsky discloses *combining* the alleged temporal events (the network-based assets list) with third party network endpoint assessments. As Patent Owner observes, combining temporal events with endpoint assessments logically would not constitute generating the temporal events based on the endpoint assessments. Prelim. Resp. 20–21.

In support of its position that “Miliefsky teaches ‘generating . . . temporal events based at least in part on analyzing the third party network endpoint assessments,’” Petitioner also cites Dr. Nielson’s testimony. Pet. 44 (citing Ex. 1003 ¶¶ 201–203). The cited testimony resembles Petitioner’s arguments. For example, Dr. Nielson testifies that “the configuration data, Miliefsky explains, can be combined with vulnerability assessment data based on CVEs in order to protect against attacks.” Ex. 1003 ¶ 201. Dr. Nielson also testifies that Miliefsky stores external third-party assessments and uses internal and external assessments for network safety. *Id.* ¶¶ 202–203. Dr. Nielson’s testimony about *combining* data, *storing* data, and *using* data for network safety does not support sufficiently Petitioner’s position that Miliefsky teaches *generating* the alleged temporal events *based at least in part on analyzing third party network endpoint assessments*.

(2) *limitation 1[g]: “correlating, by the trust orchestration server, the received endpoint events and the generated temporal events”*

According to Petitioner, “[a person having ordinary skill in the art] would understand that the combination of Mukherjee and Miliefsky discloses” “correlating, by the trust orchestration server, the received endpoint events and the generated temporal events,” as recited in limitation 1[g]. Pet. 47; Ex. 1001, 40:6–7. Addressing the claimed

“received endpoint events,” Petitioner argues that Mukherjee’s DIDS director . . . receives device and application behavior data and audit records which correspond to ‘endpoint events.’” *Id.* at 45 (citing Ex. 1003 ¶¶ 198, 201, 212; Ex. 1005, 12). As noted above in Section II.C.3.a.1, Petitioner contends that Miliefsky’s network-based assets list constitutes the claimed “temporal events.” *Id.* at 43–45.

Addressing the claimed “correlating,” Petitioner asserts that it “would have been obvious to a [person having ordinary skill in the art] to include the temporal events [(i.e., Miliefsky’s network-based assets list)] discussed with limitation 1[f] into the correlation operations of Mukherjee as suggested by Mukherjee.” Pet. 46 (citing Ex. 1003 ¶ 216). Citing Mukherjee’s discussion of its IDM’s fourth layer, Petitioner argues that “Mukherjee explicitly teaches correlating temporal context based on time windows” and that “events themselves are analyzed for patterns (i.e., relationships between events) based on signatures of an attack.” *Id.* at 46 (citing Ex. 1005, 13; Ex. 1003 ¶¶ 63–65, 214–216).

Patent Owner counters that Petitioner does not show it would have been obvious to correlate temporal events and received endpoint events. Prelim. Resp. 21–22. Patent Owner argues that “Petitioner does not disclose how Mukherjee’s alleged endpoint events and generated temporal events are correlated.” *Id.* at 21. Patent Owner asserts that “[n]one of Mukherjee’s ‘correlate[d] events’ are the temporal events (*i.e.*, network-based asset lists) described by Petitioner.” *Id.* at 22. Given this, Patent Owner argues that Petitioner does not articulate “why a [person of ordinary skill in the art] would be motivated to combine Mukherjee with Miliefsky to compare an event to descriptions of locations and protocols of network based assets,

when Petitioner’s cited portion of Mukherjee clearly describes comparing two events for their proximity in time.” *Id.*

Having reviewed the parties’ arguments and evidence, we agree with Patent Owner that Petitioner has not persuasively addressed limitation 1[g]’s requirement for “correlating . . . the received endpoint events and the generated temporal events.” Ex. 1001, 40:6–7; Pet. 45–47. Petitioner does not provide a sufficiently persuasive, objective basis for its contention that it would have been obvious to include Miliefsky’s network-based assets list “into the correlation operations of Mukherjee.” Pet. 46 (citing Ex. 1003 ¶ 216). Petitioner argues that Mukherjee suggests such a modification in the following passage:

The fourth layer of the model introduces the event in context. There are two kinds of context: temporal and spatial. As an example of temporal context, behavior which is unremarkable during standard working hours may be suspicious during off-hours [35]. The IDM, therefore, allows for the application of information about wall-clock time to the events it is considering. Wall-clock time refers **to information about the time of day, weekdays versus weekends and holidays, as well as periods when an increase in activity is expected.** In addition to the consideration of external temporal context, **the expert system uses time windows to correlate events occurring in temporal proximity.** This notion of temporal proximity implements the heuristic that a call to the UNIX who command followed closely by a login or logout is more likely to be related to an intrusion than either of those events occurring alone. **Such related patterns of behavior are also referred to as attack signatures** [23,24].

Ex. 1005, 13 (emphases added consistent with emphases in Petition);
Pet. 45–46.

Petitioner does not sufficiently support its position that this passage of Mukherjee would have suggested adding a network-based assets list into

Mukherjee's disclosed correlation operations. The cited passage discloses certain information, but not an assets list. Ex. 1005, 13. And, as Patent Owner notes, the only "correlat[ing]" mentioned in the cited passage is "correlat[ing] events occurring in temporal proximity." *Id.*; Prelim. Resp. 21–22. Petitioner does not provide a persuasive, objective basis for its view that these disclosures suggest including Miliefsky's network-based assets list into the process of correlating temporally proximate events. Pet. 46. Petitioner does not, for example, persuasively show for institution how or why Mukherjee's discussion of correlating temporally proximate events and/or its reference to attack signatures allegedly would have suggested correlating Miliefsky's network-based assets list and endpoint events. *Id.* Nor does Dr. Nielson's testimony substantiate Petitioner's argument that the cited passage of Mukherjee suggests the proposed modification. *Id.* (citing Ex. 1003 ¶ 216). Indeed, Dr. Nielson does not testify that Mukherjee suggests including a network-based assets list in correlation operations. *E.g.*, Ex. 1003 ¶ 216.

Moreover, Petitioner and Dr. Nielson do not otherwise sufficiently support Petitioner's position that it would have been obvious to correlate Miliefsky's network-based assets list and Mukherjee's events. Pet. 46; Ex. 1003 ¶¶ 209–220. Consistent with Patent Owner's arguments, Petitioner and Dr. Nielson do not articulate or substantiate a sufficiently persuasive reason that a person of ordinary skill in the art would have had for modifying Mukherjee to correlate Miliefsky's network-based assets list and Mukherjee's temporally proximate events. *E.g.*, Pet. 45–47; Ex. 1003 ¶¶ 209–220; Prelim. Resp. 21–22. Petitioner and Dr. Nielson do not, for example, explain how or why a person of ordinary skill in the art would have

thought that correlating assets and events would work or improve Mukherjee's system. Pet. 45–47; Ex. 1003 ¶¶ 209–220.

In connection with his opinion that the proposed modification would have been obvious, Dr. Nielson testifies that “Mukherjee already discloses using third-party audit packages.” Ex. 1003 ¶ 216. To the extent this testimony may support Petitioner's argument that it would have been obvious to use a third-party audit package with Mukherjee's system in some manner, it does not explain why it allegedly would have been obvious specifically to correlate Miliefsky's network-based assets list and Mukherjee's correlation events.

(3) Summary for Claim 1

For the reasons explained above, Petitioner does not demonstrate a reasonable likelihood of establishing obviousness of independent claim 1 over Mukherjee and Miliefsky.

b) Dependent Claims 2–6

Petitioner's arguments and evidence regarding dependent claims 2–6 do not cure the above-discussed deficiencies in Petitioner's challenge to independent claim 1. Pet. 48–56. Accordingly, Petitioner does not demonstrate a reasonable likelihood of establishing obviousness of any of dependent claims 2–6 over Mukherjee and Miliefsky.

D. GROUND 2 – ALLEGED OBVIOUSNESS OVER MUKHERJEE, MILIEFSKY, AND GUO

Building from Ground 1's challenge of independent claim 1 as allegedly obvious over Mukherjee and Miliefsky, in Ground 2, Petitioner cites Guo to address dependent claim 7's additional limitations related to a malware analyzer. Pet. 59. Petitioner's arguments and evidence for Ground 2 do not cure the above-discussed deficiencies in the challenge to

independent claim 1 in Ground 1. Accordingly, Petitioner does not demonstrate a reasonable likelihood of establishing claim 7 would have been obvious over Mukherjee, Miliefsky, and Guo.

III. CONCLUSION

Because we determine that the information presented in the record does not establish there is a reasonable likelihood that Petitioner would prevail with respect to at least one challenged claim of the patent '616 patent, we do not institute an *inter partes* review.

IV. ORDER

For the reasons given, it is:

ORDERED that the Petition is *denied*, and no trial is instituted.

IPR2023-00799
Patent 9,092,616 B2

FOR PETITIONER:

Robert Hansen
Scott Amankwatia
THE MARBURY LAW GROUP, PLLC
bhansen@marburylaw.com
samankwatia@marburylaw.com

FOR PATENT OWNER:

Peter Lambrianakos
Vincent Rubino
Joseph Mercadante
FABRICANT LLP
plambrianakos@fabricantllp.com
vrubino@fabricantllp.com
jmercadante@fabricantllp.com