UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

CISCO SYSTEMS, INC.,
Petitioner,

v.

K.MIZRA LLC,
Patent Owner.

Case IPR2021-00593
U.S. Patent 8,234,705

PATENT OWNER'S RESPONSE
PURSUANT TO 37 C.F.R. § 42.120

TABLE OF CONTENTS

TABLE OF AUTHORITIES

CASES

EXHIBIT LIST

Exhibit No.	Exhibit Description
2009	Declaration of Dr. Nenad Medvidovic
2010	Deposition Transcript of Dr. Narasimha Reddy (12-1-2021)
2011	A Practical Guide to TPM 2.0

I. INTRODUCTION

Patent Owner's U.S. Patent 8,234,705 (the '705 Patent") claims a technique to control access to secure computer networks by detecting the health and compliance status of a computer in a trusted manner and allowing remediation of the computer to ensure compliance with network security policies. *See, e.g.*, Ex. 1001 at Abstract. Petitioner Cisco Systems, Inc. sells network access control products and solutions that infringe the '705 Patent. Patent Owner filed a lawsuit in the Western District of Texas seeking to remedy this infringement. Claims 1-3, 5-7, 9-13, and 15-19 are currently asserted in that case.

On September 24, 2021, the Board instituted review of challenged Claims 1-3, 5-13, and 15-19 based on the single ground raised in the Petition. Paper 9 at 2. The Petition challenges the claims based on the proposed combination of Gleichauf (Ex. 1005), Ovadia (Ex. 1006), and Lewis (Ex. 1007).

The Petition, however, fails to demonstrate a motivation to combine the asserted references, and even if the references could be combined, the Petition still fails to demonstrate that the combination discloses certain claim elements.

One of the key innovative features of the '705 Patent that had never been practiced before was the use of a trusted computing base and a trusted platform module to report the health of a computer as part of a security compliance check in order to gain access to a secure network. To reconstruct this key aspect of the

invention, the Petition asserts the combination of Gleichauf's posture agent software applications with Ovadia's trusted platform module (TPM), but fails to establish a motivation to combine the references in this proposed manner.

First, Gleichauf and Ovadia are fundamentally at odds and directed towards diametrically opposite goals. The purpose of Gleichauf is to ensure that the current security state of a computer has been updated with the most recent patches to the anti-virus software. The purpose of Ovadia is to ascertain integrity measurements relating to a device's configurations to ensure that the integrity metrics have remain ***unchanged***—the term of art "integrity" is misleading here and has nothing to do with the security state of the device. In fact, the "philosophy of integrity measurement, storage and reporting is that a platform may be permitted to enter any state possible including undesirable or insecure states," just as long as the integrity metrics have not been altered or changed. This directly contravenes the spirit of Gleichauf, and accordingly, a POSITA would not have been motivated to combine it with Ovadia.

Second, the proposed combination of Gleichauf and Ovadia lacks any enabling disclosure, and thus fails to meet the Federal's Circuit's test for obviousness in which prior art must be enabled to render an invention obvious. A POSITA would understand that the manner in which the Petition combines the two references is fundamentally impossible, much less a matter of obvious design choice.

That is, Gleichauf's user-space software applications simply cannot be executed in Ovadia's kernel space TPM execution engine as Petitioner suggests for a number of reasons as set forth below. Notably, the TPM is intended to be securely locked down with a segregated memory space and dedicated for a limited set of low-level system commands. A POSITA would understand that altering the TPM specification to accommodate the type of combination that Petitioner proposes would have been self-defeating—i.e., having the kernel space TPM execution engine run arbitrary user space application software in an end run around the restricted set of TPM commands and interfaces. In other words, a POSITA would not have thought to combine Gleichauf with Ovadia in the way that Petitioner suggests, much less have a reasonable expectation of success.

For all of the foregoing reasons, and as described in more detail below, the Petition fails to meet its burden to prove unpatentability.

II. OVERVIEW OF THE '705 PATENT AND THE STATE OF THE ART

The '705 Patent is titled "Contagion Isolation and Inoculation" and was issued by the United States Patent Office to inventors James A. Roskind and Aaron R. Emigh on July 31, 2012. The earliest application related to the '705 Patent was filed on September 27, 2004. Ex. 1001 at (60).

The '705 Patent is directed to methods and systems for securing a computer network by detecting viruses and/or vulnerabilities on a computer attempting to

access a secure network, quarantining the computer by restricting access to the network upon detection of an infestation or vulnerability, and permitting limited access to the protected network for remedying the insecure condition on the computer. Ex. 1001 at Abstract, 11:15-12:13.

Prior art computer network security systems faced considerable challenges with protecting and maintaining up-to-date security software on mobile devices such as employees' personal laptops, which posed significant security risks that could allow attackers or viruses stealth access into a business's network, bypassing IT security measures. *Id*., 1:34-38. While a network security appliance or hardware can adeptly keep out unwanted external intrusions into the network, the most exploitable vulnerabilities of a computer network are found on the end-user computers that roam throughout various other insecure public and private network domains and then access the presumably secure network day in and day out. *Id*.

The '705 Patent closes this loophole by verifying that any device attempting to access a company's network meets its standards for network security and will not introduce dangerous computer programs or viruses into the network. A particularly novel aspect of the invention lies in this verification — an attestation of cleanliness from a trusted computing base associated with a trusted platform module. *See, e.g.*, Ex. 1001 at Claim 1, 13:64-14:12; Ex. 1002, 185 (Jan. 10, 2010 Office Action Response at p. 8). The inventors of the '705 Patent recognized the potential risk of

false attestations as a further significant security hole. Accordingly, they believed the implementation of a trusted computing base associated with a tamper-resistant trusted platform module to be a significant point of innovation. *See, e.g.*, Ex. 1002, 185 (Jan. 10, 2010 Office Action Response at p. 8).

Furthermore, when "a request is received from a host, e.g., via a network interface, to connect to a protected network, it is determined whether the host is required to be quarantined. If the host is required to be quarantined, the host is provided . . . limited access to the protected network . . . only as required to remedy a condition that caused the quarantine to be imposed, such as to download a software patch, update, or definition; install, remove, and/or configure software and/or settings as required by a policy; and/or to have a scan or other diagnostic and/or remedial operation performed." Ex. 1001 at 3:8-20.

III. CLAIM CONSTRUCTION

Petitioner has proposed a construction for the terms "trusted computing base" and "trusted platform module" recited in challenged claims 1, 12, and 19. Paper 2, 11-16. Patent Owner does not dispute Petitioner's proposed construction for the claim term "trusted computing base."

For the claim term, "trusted platform module," Patent Owner disagrees with the construction proposed by Petitioner. The proper construction of "trusted platform module" is "a secure cryptoprocessor that can store cryptographic keys, which

includes but is not limited to a cryptoprocessor that implements the Trusted Platform Module specification from the Trusted Computing Group." This construction accounts for general implementations of the trusted platform module as disclosed in the intrinsic record, which includes cryptoprocessors contemplated for the purpose of this invention—i.e., storing cryptographic keys and implementing cryptographic functionality in a way that utilizes cryptography to prevent falsified reports as taught by the invention. Ex. 1002, 184-185.

Petitioner offers its construction for this term as "a secure cryptoprocessor that implements the Trusted Platform Module specification from the Trusted Computing Group." Unlike Patent Owner's construction, Petitioner's construction improperly attempts to import a particular embodiment—the specific TCG specification of column 14 in the '705 Patent's description—into the claims. Nevertheless, Petitioner fails to show that the claims at issue are unpatentable under either claim construction, and the analysis set forth below applies equally under either construction.

IV. GROUND 1: CLAIMS 1-3, 5-13 AND 15-19 ARE PATENTABLE OVER THE COMBINATION OF GLEICHAUF, OVADIA, AND LEWIS

A. Legal Standard for Obviousness

A patent is not obvious unless "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." *KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 406 (2007). Hindsight reconstruction of a patent claim using multiple prior art references in piecemeal fashion is impermissible to show obviousness. *See Grain Processing Corp. v. Am.-Maize Prods. Co.*, 840 F.2d 902, 907 (Fed. Cir. 1988) ("Care must be taken to avoid hindsight reconstruction by using 'the patent in suit as a guide through the maze of prior art references, combining the right references in the right way so as to achieve the result of the claims in suit.'") (quoting *Orthopedic Equip. Co. v. United States*, 702 F.2d 1005, 1012 (Fed. Cir. 1983)).

To prove obviousness, it is "important to identify a reason that would have prompted a person of ordinary skill in the relevant field to combine the elements in the way the claimed new invention does." *KSR*, 550 U.S. at 418; *see also Nortek Security & Control LLC v. The Chamberlain Grp., Inc.,* IPR2018-01793 (PTAB Mar. 11, 2019) ("General statements, such as a POSITA 'would have attempted to maximize the product's safety and convenience of installation' . . . do not provide an adequate reason for combining aspects of the prior art to produce the claimed [invention].") (internal citations omitted). A petitioner must not only demonstrate there is a reason to make each combination, it must also demonstrate a reasonable expectation that such a combination would be successful. Otherwise, a POSA would not arrive at the claimed combination. *In re*

Stepan Co., 868 F.3d 1342, 1346 (Fed. Cir. 2017). In other words, the petitioner must explain ***how*** and ***why*** the references would be combined to arrive at the claimed invention. *See Metalcraft of Mayville, Inc. v. The Toro Co.*, 848 F.3d 1358, 1367 (Fed. Cir. 2017).

It is a bedrock principle of patent law that "a patent is not proved obvious merely by demonstrating that each of its elements was, independently, known in the prior art." *See KSR*, 550 U.S. at 418. Thus, simply stating that a POSA could have combined certain references is insufficient. *See InTouch Techs., Inc. v. VGO Communications, Inc.*, 751 F.3d 1327, 1352 (Fed. Cir. 2014). Moreover, the Federal Circuit has held that "[t]o render a claim obvious, the prior art taken as a whole, must enable a skilled artisan to make and use the claimed invention." *Raytheon Techs. Corp. v. General Elec. Co.*, 993 F.3d 1374, 1380 (Fed. Cir. Apr. 16, 2021).

B. Independent Claims 1, 12, and 19 Are Patentable Over the Combination of Gleichauf, Ovadia, and Lewis

1. The Petition Fails to Demonstrate that the Combination Discloses "Quarantining the First Host" Including "Receiving a Service Request Sent by the First Host"

Claim 1 recites a method for protecting a network comprising at least three steps in a specific order, the second of which Petition fails to show is met by the combination of references. The ***first step*** requires "detecting an insecure condition on a first host" including "contacting a trusted computing base associated with a trusted platform module within the first host," "receiving a response," and

"determining whether the response includes a valid digitally signed attestation of cleanliness." Separately, only "when it is determined [by the first step of detecting an insecure condition] that the response does not include a valid digitally signed attestation of cleanliness," ***the second step*** then "quarantin[es] the first host." This second step of "quarantining the first host" includes "preventing the first host from sending data to one or more other hosts associated with the protected network," which includes an initial step of "receiving a service request sent by the first host" before the host is directed to a quarantine notification page. Finally, ***the third step*** recites "permitting the first host to communicate with the remediation host."

By its plain language, claim 1 therefore requires "contacting a trusted computing base associated with a trusted platform module within the first host" in the first step of "detecting an insecure condition on a first host." Ex. 2009 [Medvidovic Decl.] at ¶ 57-59. And the claim separately requires "receiving a service request sent by the first host" in the subsequent second step of "quarantining the first host"—after "it is determined that the response does not include a valid digitally signed attestation of cleanliness" from step one. Ex. 2009 [Medvidovic Decl.] at ¶ 57-59. Dr. Reddy—Petitioner's expert—agrees with this reading of claim 1. Ex. 2010 at 25, 47-49. Claims 12 and 19 contain similar limitations and must be interpreted in the same manner. The Petition, however, fails to show that Gleichauf meets this second step as required.

The Petition alleges that Gleichauf discloses the claimed "receiving a service request sent by the first host" in the second step of "quarantining the first host" but fails to show that "receiving a service request" in Gleichauf occurs after the first claim step of "detecting an insecure condition on a first host" including "contacting a trusted computing base associated with a trusted platform module within the first host." Paper 2 at 46-47. The Petition alleges Gleichauf's "access request" or "signaling request" in an "attempt to access the network resources" corresponds to the claimed "service request" in the second claim step. *Id*.

Gleichauf however discloses that the access/signaling request (item 40 in Fig. 1) is the initial step of device 22 accessing network 24 that occurs before any other step:

Ex. 1005 – Gleichauf at Fig. 1

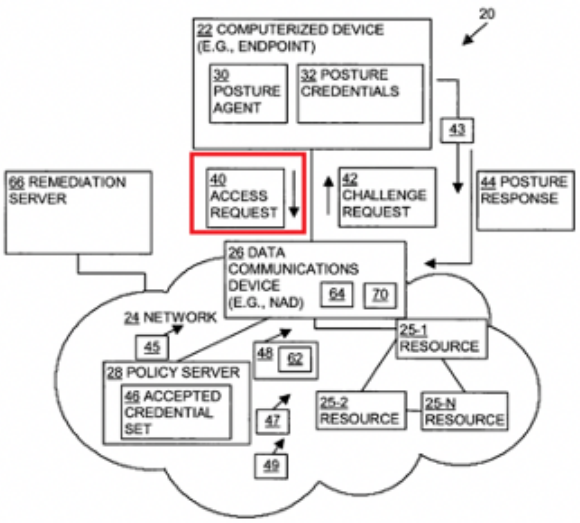


FIG. 1

Ex. 1005 at 10:49-55; Ex. 2009 [Medvidovic Decl.] at ¶ 60. This is confirmed by both Fig. 2 of Gleichauf, which shows the procedure performed by data communication device 26 on network 24, and Fig. 5, which shows the procedure performed by endpoint device 22. Ex. 1005 at Figs. 2, 5; Ex. 2009 [Medvidovic Decl.] at ¶ 61-62. In Fig. 5, the endpoint device transmits the access request to the network as the first step of accessing the network, before posture credentials are collected in items 144-152:

Ex. 1005 – Gleichauf at Fig. 5

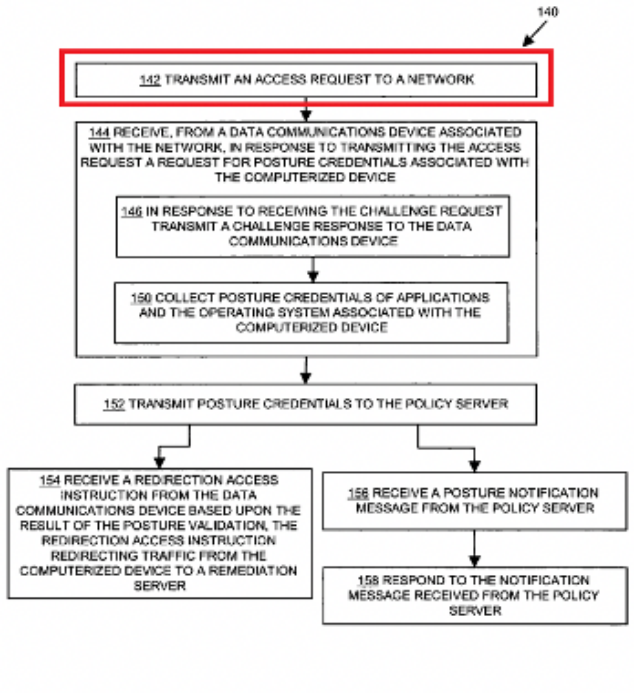
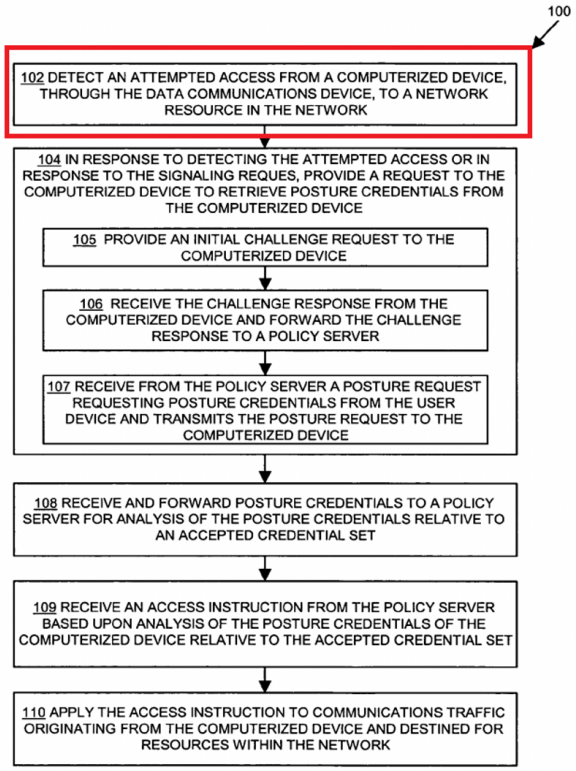


FIG. 5

Ex. 1995 at Fig. 5; Ex. 2009 [Medvidovic Decl.] at ¶ 62. Likewise in Fig. 2, the network device 26 detects the access request in item 102 before receipt and verification of the posture credentials in items 104 and 108:

Ex. 1005 – Gleichauf at Fig. 2



Ex. 1005 at Fig. 2; Ex. 2009 [Medvidovic Decl.] at ¶ 63. Indeed, the Petition at 29-30 also alleges that Gleichauf's "access request" is an initial attempt to connect to a protected network, which in response detects an insecure condition by requesting and confirming posture credentials. Paper 2 at 29-30.

As Gleichauf's access request is an initial request that prompts the Gleichauf system to instruct the endpoint device to collect and transmit posture credentials, it cannot also correspond to the claimed service request of the later step two. Ex. 2009 [Medvidovic Decl.] at ¶ 64. The Petition maps Gleichauf's posture credentials to the

claimed "digitally signed attestation of cleanliness." Paper 2 at 39-42. Claim 1 however requires the service request to be received after "it is determined that the response does not include a valid digitally signed attestation of cleanliness." Ex. 2009 [Medvidovic Decl.] at ¶ 64. In other words, claim 1 requires the network to have validated the digitally signed attestation of cleanliness before the service request is received. *Id*. at 57-64. Gleichauf, by contrast, transmits the access/signaling request first, before posture credentials are collected, received, or verified. *Id*. at 60. While the claimed service request must be received after "digitally signed attestation of cleanliness" are confirmed, Gleichauf's access request is received before the posture credentials are collected and transmitted to the network. *Id*. at 60-64.

Because Gleichauf's "access request" or "signaling request" prompts Gleichauf's system to produce the posture credentials that are used to detect the insecure condition as recited in the first claim step, these access/signaling requests cannot teach the claimed "service request" in the second claim step, which Claim 1 requires to be received *after* "it is determined that the response does not include a valid digitally signed attestation of cleanliness." Ex. 2009 [Medvidovic Decl.] at ¶ 64. The Petition therefore fails to show that the claimed "service request" is obvious.

2. The Petition Fails to Demonstrate that it Would Have Been Obvious to "Serv[e] a Quarantine Notification Page to the First Host"

The Petition also fails to show that the claim element to serve a quarantine notification page is obvious. Claim 1 requires "serving a quarantine notification page to the first host when the service request comprises a web server request…" The Petition concedes at 47-49 that Gleichauf discloses a remediation server and does not disclose a quarantine notification page, but nonetheless alleges that it would have been obvious to modify Gleichauf by redirecting HTTP traffic from Gleichauf's remediation page to a quarantine notification page. Paper 2 at 47-49.

The Petition however fails to provide any support for this proposition. In fact, a POSITA would not find such a modification obvious. A POSITA would see no need to redirect traffic away from a remediation server to a notification page, nor does the claim require such action; what the claim requires is redirecting traffic bound for a non-remediation related webpage to a notification page. *See, e.g.*, Ex. 1001 at Claim 1 ("serving a quarantine notification page to the first host when the service request comprises a web server request, and in the event the service request comprises a DNS query, providing in response an IP address of a quarantine server configured to ***serve the quarantine notification page if a host name that is the subject of the DNS query is not associated with a remediation host***"); Ex. 2009 [Medvidovic Decl.] at ¶ 66.

The Petition suggests combining Gleichauf with secondary reference Lewis to satisfy the claim limitation for redirection to a notification page. Lewis, however, only discloses a webpage notifying the user's device is in quarantine, but does not provide any motivation to redirect traffic from a remediation server to this notification webpage. More importantly, a POSITA would understand that Gleichauf discloses that users are directed to the remediation server in ***all*** cases. First, Gleichauf discloses that preexisting traffic to the remediation server is expressly permitted. Ex. 1005 at 14:62-66 ("The access instruction 48 may permit traffic from user device 22 only to certain destination addresses, e.g., IP address of remediation server…"); Ex. 2009 [Medvidovic Decl.] at ¶ 65-66. Second, Gleichauf discloses that all other traffic that is not bound for the remediation server is redirected to the remediation server. Ex. 1005 at 15:55-67. Indeed, Gleichauf's operation—confirmed by Petitioner's expert Dr. Reddy—directs all traffic to the remediation server to remediate insecure devices. Ex. 2010 at 49-53; Ex. 2009 [Medvidovic Decl.] at ¶ 65-66.

A POSITA would understand that the purpose of the remediation server is to remediate the user's device. Ex. 2009 [Medvidovic Decl.] at ¶ 66. As Petitioner's expert Dr. Reddy confirms, once a user device is directed to a remediation server, the user would expect the device to be remediated. Ex. 2010 at 51-53. Gleichauf's system works for its intended purpose of remediating devices, and there is no

rationale to redirect traffic from a remediation server to anywhere else. Therefore, it is not obvious to redirect traffic from a remediation server to a quarantine notification page. Ex. 2009 [Medvidovic Decl.] at ¶ 65-66.

3. The Petition Fails to Demonstrate That the Combination Would Yield Predictable Results or That a POSITA Would Have Had a Reasonable Expectation of Success

Petitioner fails to show that combining Gleichauf with Ovadia and Lewis would yield predictable results or that it would have a reasonable expectation of success. At a minimum, Gleichauf and Ovadia are fundamentally different and directed towards opposite goals. Gleichauf is directed to ensuring that the most up-to-date patches of antivirus software are installed on a computer. In contrast, Ovadia is directed towards ensuring that the configuration of a device has *not* changed. Thus, a POSITA would not have looked to combine Gleichauf and Ovadia let alone have any reasonable expectation of success in combining them or that their combination would yield predictable results.

Additionally, Gleichauf's posture agent could not have been combined with Ovadia's Trusted Platform Module (TPM) in the way that the Petition asserts by having the posture agent software executed inside the TPM's execution engine. As detailed below, a POSITA would have understood that not only does the prior art fail to provide an enabling disclosure for this combination, it would have been

technologically unworkable to implement Gleichauf's posture agent software inside the TPM in the manner that Petitioner suggests.[1]

(a) Gleichauf and Ovadia are Fundamentally Different and Directed Towards Opposite Goals

Ovadia is fundamentally different from Gleichauf, and Ovadia's teachings have nothing to do with how to implement Gleichauf's techniques. Gleichauf is directed to assessing and reporting the health or security state of a computer to ensure that the device has the most current and "up-to-date patches to the operating system or applications, such as anti-virus applications, associated with the computerized device." Ex. 1005 at 5:4-16; *see also, e.g.*, *id*., 8:65-9:1 ("the user device 22 includes a posture plug-in 32 that indicates the version of operating

[1] Petitioner also asserts that there was a motivation to combine Gleichauf and Ovadia to ensure interoperability with respect to network security functionality. Paper 2 at 22. The fact remains, however, that the proposed combination would not have been technologically possible and a POSITA would not have found an enabling disclosure in the combination. Moreover, a POSITA would not have looked to Ovadia to standardize the protocols of Gleichauf to ensure interoperability because Gleichauf already taught the use of a sufficiently secure and trustworthy standardized protocol, the Extensible Authentication Protocol (EAP) to securely transmit posture information. Ex. 1005 at 11:3-12.; Ex. 2009 [Medvidovic Decl.] at ¶ 87.

systems used by the user device 22 or the presence of updated 'patches' associated with the operating systems"); *id*., 11:18-23 ("upgrade or provide up-to-date patches to applications, such as anti-virus applications . . . to minimize the susceptibility of the user device 22 to an unauthorized application, such as a virus, thereby minimizing the potential for the user device 22 to receive malicious content or introduce an unauthorized application onto the network 24 at a later time").

In contrast, Ovadia is directed towards device *integrity*—i.e., ensuring that device configurations remain *unchanged* by way of "integrity measurements." The word "integrity" is a well understood term of art in the computer security field; POSITAs would understand it to mean "[a]n indication that a message *has not been altered* during storage or transmission." Ex. 2009 [Medvidovic Decl.] at ¶ 70 (emphasis added); Ex. 2011 at 11 (page 19 of 375). A TPM "integrity measurement" corresponds to the platform environment at the time of measurement. *See* Ex. 1006 at 10:49-53 ("More specifically, an integrity metric corresponding to a platform environment may reflect a single environment component (i.e., firmware/software component), or a combination of components used to form an environment that exists at the time the integrity metric is measured.").

Ovadia teaches the use of a "static CRTM" as part of the integrity measurement because this value *does not change*, noting that this measurement "cannot be partially replaced" and should be *immutable*. Ex. 1006 at 10:61-66 ("In

one embodiment, a static CRTM is created via a measurement of the platform's base firmware. The base firmware is a portion of firmware provided by the platform that cannot be partially replaced. The CRTM should be an immutable portion of the platform's initialization code that executes upon any platform reset").

A POSITA would understand that the integrity measurement is NOT an attestation of a computer's security state indicating whether the computer has the most current and "up-to-date patches to the operating system or applications, such as anti-virus applications, associated with the computerized device." Ex. 2009 [Medvidovic Decl.] at ¶ 73. The integrity measurement only asserts that the platform had a certain state at a certain time—making no assertions regarding security states. *Id*. Indeed, the TCG specification plainly states that the platform can even be in an *insecure state*—directly contravening the goal in Gleichauf: "The philosophy of integrity measurement, storage and reporting is that *a platform may be permitted to enter any state possible including undesirable or insecure states*, but that it may not be permitted to lie about states that is was or was not in." Ex. 1012 at 6 (page 12 of 54); Ex. 2009 [Medvidovic Decl.] at ¶ 73.

Therefore, Gleichauf's posture credentials are NOT analogous to Ovadia's integrity measurement, and it would not have been obvious to combine the two references. Ex. 2009 [Medvidovic Decl.] at ¶ 74. Gleichauf's posture credentials and Ovadia's integrity measurements are directed to fundamentally different

objectives—Gleichauf's posture credentials are for ensuring that the current security state has the most up-to-date patch levels while Ovadia's integrity measurements are for ensuring that the integrity metrics have remain unchanged. Ex. 2009 [Medvidovic Decl.] at ¶ 74.

(b) Gleichauf's Posture Agent Could Not Have Been Run by the TPM Execution Engine of Ovadia and this Combination has no Enabling Disclosure Otherwise

Petitioner asserts that "one obvious way to implement Gleichauf's posture agent with trusted platform module functionality would be for the posture agent software to be executed by the execution engine in Ovadia's trusted platform module." Paper 2 at 36. Petitioner also asserts that "[a]s another example implementation of Gleichauf's posture agent with trusted platform module functionality, the posture agent software would simply be integrated with a software-implemented trusted platform module" based on Ovadia's narrow disclosure that TPM functionality can be embodied in software. Paper 2 at 36; Ex. 1003, ¶ 94.

As detailed below, a POSITA would understand that this implementation, with either a hardware or software based TPM, was fundamentally impossible—i.e., the posture agent software could not have been run by the TPM's execution engine. Ex. 2009 [Medvidovic Decl.] at ¶ 75. Note, however, that Petitioner had no other choice but to combine Gleichauf and Ovadia in this implausible manner in order to satisfy its articulation of the "digitally signed" claim limitation in claim element

[1.4]. That is, Petitioner's sole identification for this element is using the TPM's Attestation Integrity Key (AIK) to digitally sign the attestation of cleanliness. *See, e.g.*, Paper 2 at 36. The Petition fails to propose any other possible modification of the prior art for the combination to render this element obvious. But under this theory, only data generated by the TPM itself can be digitally signed with this TPM key. *See, e.g.*, Paper 2 at 36 ("With the posture agent executing in the trusted platform module execution engine, the information collected by the posture agent would be 'data generated by a TPM' and thus eligible for digital signature under the Trusted Platform Module main specification"); *see also id.*, 39-40. Thus, Petitioner is forced to place the posture agent within the TPM (whether software or hardware based), in order for any output to be "data generated by a TPM" for signing with the TPM key. No other combination is possible under Petitioner's theory. Nevertheless, this theory fails because the posture agent software could not have been run by the TPM's execution engine.

A TPM is not a general-purpose processor and is not simply fungible or extensible with code. Ex. 2009 [Medvidovic Decl.] at ¶ 76. Indeed, Petitioner proposes that a TPM be construed at least as a "cryptoprocessor that implements the Trusted Platform Module specification from the Trusted Computing Group." While Patent Owner proposes a broader construction, the TPM specification and specified firmware are governed by the TCG consortium of companies in order to establish a

consensus-built set of standards for establishing "trust" under both constructions. *See* Ex. 1012 at 2 (page 8 of 54). At the foundation of this "trust" lie "roots of trust," such as the TPM—which are "elements that *must* be trusted." *See* Ex. 1012 at 6 (page 12 of 54) (emphasis added). *Id*. As a building block of a trusted platform, TPM components are trusted to work properly without additional oversight. *See* Ex. 1012 at 19 (page 25 of 54); Ex. 2009 [Medvidovic Decl.] at ¶ 77. That is because, for example, the TCG further requires that a TPM be "physically protected from tampering" and that software TPM implementations "must justify a hardware-equivalent interpretation for tamper-protection." *See* Ex. 1012 at 21 (page 27 of 54); Ex. 2009 [Medvidovic Decl.] at ¶ 77. Thus, a POSITA would have understood that altering the TPM specification to accommodate the type of combination that Petitioner proposes would have been self-defeating—i.e., so that the TPM execution engine can run arbitrary user space application software in a complete end run around the defined set of commands and interfaces.[2] Ex. 2009 [Medvidovic Decl.] at ¶ 77.

[2] The Institution Decision states that "[t]o the extent Patent Owner's argument is premised on physical incorporation of the TPM execution engine or a TMP chip into Gleichauf's device, we disagree with Patent Owner because '[t]he test for

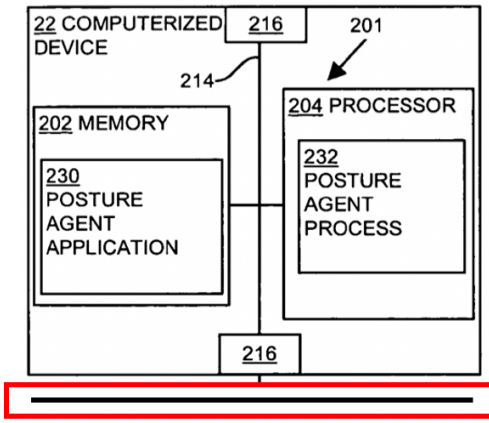
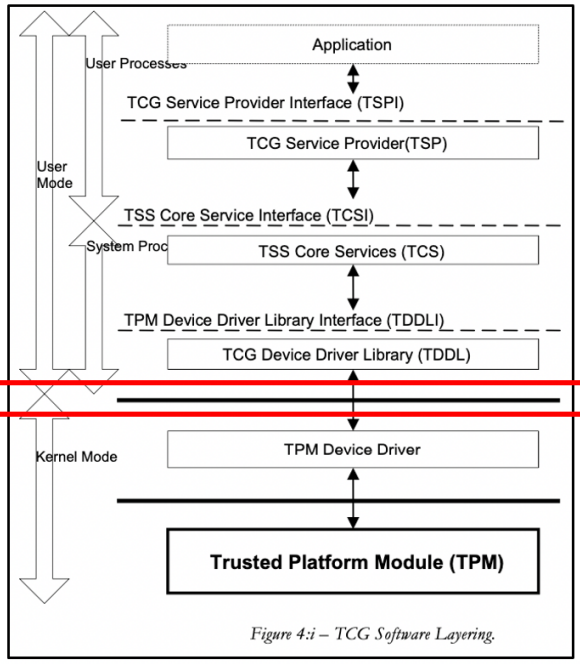
Rather, the TCG precisely defined the exact APIs for interacting with the TPM—rendering the TPM inextensible. *See, e.g.,* Ex. 1012 at 35-40 (pages 41-46) (Section 4.6.3 TCG Commands and Interfaces). The TPM was only designed to support a limited set of specific TPM commands such as "Seal" and "Unseal." *Id*; Ex. 2009 [Medvidovic Decl.] at ¶ 78. Indeed, the TPM execution engine can only "run program code to execute TPM commands received from the I/O port." *See, e.g.*, Ex. 1013 at 19 (page 30 of 161) (Section 2.2.9, Execution Engine); Ex. 1012 at 35-37 (page) 41-43 (Section 4.6.3.1 TPM Commands); Ex. 2009 [Medvidovic Decl.] at ¶ 78. Thus, a POSITA would have understood that with this highly secure and dedicated set of TPM-specific commands, Gleichauf's posture software certainly

---

obviousness is not whether the features of a secondary reference may be bodily incorporated into the structure of the primary reference . . . .'" To be clear, however, Patent Owner's argument is not premised on any physical incorporation of the TPM into Gleichauf's device. Rather, the argument is directed to the fact that it would not be possible, much less obvious to a POSITA, to combine Ovadia's TPM with Gleichauf's user application and plug-ins in the manner that Petitioner proposes as further explained here. Indeed, the asserted combination of prior art is not sufficiently enabled to meet the Federal Circuit's test for obviousness. *Raytheon Techs. Corp.*, 993 F.3d at 1380.

could not have been executed by the TPM execution engine. Ex. 2009 [Medvidovic Decl.] at ¶ 79. Because TPM commands are specific and defined, a set of TPM commands capable of implementing Gleichauf's posture software simply did not and would not exist. And because the set of TPM commands are intentionally limited to preserve the security of the TPM, new TPMs to implement the posture software would not be created. For these reasons, integrating the posture agent software in the TPM execution engine simply would not have been a matter of design choice or implementation, much less an obvious one. *Id*.

In particular, according to the TCG specification, software *applications* such as Gleichauf's posture agent application are run in the user space external to the TPM, while system level code, such as device drivers and the TPM execution engine are run in the kernel space as shown below.

Ex. 1005 – Gleichauf at Fig. 8	Ex. 1012 – TCG Arch. Spec. at 26
	

Ex. 1005 at 24:8-12, 25:62-26:19, Fig. 8; Ex. 1012 at 26 (page 32 of 54); Ex. 2009 [Medvidovic Decl.] at ¶ 79. Generally, computer memory is typically divided into the kernel space and the user space. *Id*. at ¶ 80. The kernel space is dedicated for running system level operations such as device drivers and behind-the-scenes tasks needed to run a computer, such as memory allocation and process management. *Id*. The user space is for running software applications. *Id*. Indeed, running a user space software application in the kernel space is simply not possible, much less a matter of obvious implementation. *Id*. at ¶ 81. Therefore, a POSITA would have understood that Gleichauf's posture agent application could not have been executed by the TPM execution engine in the kernel space in either hardware or software. *Id*.

Petitioner further suggests that the posture agent with so-called TPM functionality would in turn interface with Gleichauf's posture plug-ins in the same way as disclosed in Gleichauf by using "posture plug-in API" to collect posture information requested by Gleichauf's policy server. Paper 2 at 36; Ex. 1003, ¶ 94. A POSITA would have understood, however, that this would not have been technologically possible, as this would additionally require the TPM execution engine to invoke a posture plug-in that exists at the application layer in the user space. Ex. 1005 at 3:36-45, 8:47-52 ("Each posture plug-in 32 indicates a security posture with respect to a particular manufacturer's application"); Ex. 2009 [Medvidovic Decl.] at ¶ 82. And there is simply no disclosure in the TPM specification or Ovadia for the TPM to invoke user space software applications such as posture plug-ins. *See, e.g.*, Ex. 1012 at 35-40 (pages 41-46) (Section 4.6.3, Summary of TCG Commands and Interfaces); Ex. 2009 [Medvidovic Decl.] at ¶ 82. Nor has Petitioner's expert Dr. Reddy offered any opinion in any level of implementation detail on how TPM commands would be implemented in the proposed combination of Gleichauf and Ovadia. Ex. 2010 at 67-71.

Again, the TPM is instead designed to run function-specific firmware with a limited set of TPM commands and interfaces. *Id*; Ex. 2009 [Medvidovic Decl.] at ¶ 83. A POSITA would have understood that user application APIs, such as Gleichauf's posture plug-in API, were not supported by the limited set of TPM

interfaces, nor would alteration of the defined set of TPM interfaces be feasible. *Id*. at ¶ 84. As Petitioner's expert Dr. Reddy states, the TPM does not support general purpose processing. Ex. 2010 at 63. Tampering with the TPM in such a way would invite breaches in security of the TPM and be grossly contrary to the purpose of the TPM to have a separate and protected memory space. *Id*. Petitioner's Expert Dr. Reddy also testified that "the intent is that TPM is not going to allow outside software or hardware to access those locations." Ex. 2010 at 60.

Specifically, Petitioner's proposed combination would further require the TPM to invoke Gleichauf's plug-in APIs provided by yet another party, which would open up even greater risks for tampering with the security of the TPM. *See, e.g.*, Ex. 1005 at 12:21-37 ("the posture agent 30 […] request[s] additional types of posture credentials from an existing posture plug-in 32 or from an additional posture plug-in produced by a new security application or third-party […] using a posture plug-in to query each vendor specific application"); Ex. 2009 [Medvidovic Decl.] at ¶ 85. Thus, the prior art combination as a whole does not and would not provide an enabling disclosure for further invoking posture plug-ins from the TPM. *See Raytheon Techs. Corp.*, 993 F.3d at 1380 (prior art must be enabled to render an invention obvious). And Petitioner's expert Dr. Reddy has not offered any opinion in any level of implementation detail on how TPM commands would be

implemented in the proposed combination of Gleichauf and Ovadia. Ex. 2010 at 67-71 ("I have not offered that kind of detail here on how that could be done").

In sum, even if Petitioner's implementation was technologically workable, which it is not, it places a user space software application into a kernel space execution engine, which is otherwise intended to be securely locked down with a segregated memory space dedicated for a limited set of low-level system commands. A POSITA would have well understood not to employ such an implementation because this would squarely defeat the clear purpose of the TPM. Ex. 2009 [Medvidovic Decl.] at ¶ 86. Therefore, Gleichauf's posture agent could not have been combined with Ovadia's TPM in the manner that Petitioner asserts.

C. Claims 2-3, 5-11, 13, and 15-18 are Patentable Over the Combination of Gleichauf, Lewis, and Ovadia

Claims 2-3 and 5-11 depend on independent Claim 1, and Claims 13 and 15-18 depend on independent Claim 12. For the same reasons set forth for independent Claims 1 and 12 above, dependent claims 2-3, 5-11, 13, and 15-18 are also patentable over the combination of Gleichauf, Lewis, and Ovadia.

V. CONCLUSION

Petitioner has not established a reasonable likelihood that it will prevail on Ground 1 to establish claims 1-3, 5-14, and 15-19 of the '705 Patent are unpatentable, and therefore the claims should be affirmed.

Respectfully Submitted,

Dated: January 7, 2022 */Palani P. Rathinasamy/*

James T. Carmichael, Reg. No. 45,306
CARMICHAEL IP, PLLC
8000 Towers Crescent Dr., 13th Floor
Tysons Corner, VA 22182
Tel: (703) 646-9255

David D. Schumann, Reg. No. 53,569
Palani P. Rathinasamy, Reg. No. 63,541
Cliff Win, Jr. (*pro hac vice)*
Folio Law Group PLLC
1200 Westlake Ave. N.
Seattle, WA 09109
Tel: (206) 880-1802

Attorneys for Patent Owner

(Case No. IPR2021-00593)

CERTIFICATION UNDER 37 C.F.R. § 42.24(d)

Under the provisions of 37 C.F.R. § 42.24(d), the undersigned hereby certifies that the word count for all portions of the foregoing Patent Owner's Response totals 6,127 words per Microsoft Word, which complies with the requirement of 14,000 words allowed under 37 C.F.R. § 42.24(d).

Respectfully Submitted,

Dated: January 7, 2022

/Palani P. Rathinasamy/

Palani P. Rathinasamy, Reg. No. 63,541
Folio Law Group PLLC
1200 Westlake Ave. N.
Seattle, WA 09109
Tel: (206) 880-1802

Attorneys for Patent Owner

CERTIFICATE OF SERVICE

As authorized by Patent Owner's Mandatory Notice, I hereby certify that on January 7, 2022, a copy of this document has been served in its entirety by electronic mail on Petitioner's lead and backup counsel.

Theodore M. Foster
ipr.theo.foster@haynesboone.com
David L. McCombs
david.mccombs.ipr@haynesboone.com
Gregory P. Huh
gregory.huh.ipr@haynesboone.com
Eugene Goryunov
eugene.goryunov.ipr@haynesboone.com

Dated: January 7, 2022

/Palani P. Rathinasamy/
Palani P. Rathinasamy, Reg. No. 63,541
Folio Law Group PLLC
1200 Westlake Ave. N.
Seattle, WA 09109
Tel: (206) 880-1802

Attorneys for Patent Owner