

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

NETSKOPE, INC.,

Petitioner,

v.

K. MIZRA LLC,

Patent Owner.

PTAB Case No. IPR2025-01115

Patent No. 8,234,705 B1

**PETITION FOR *INTER PARTES* REVIEW
OF U.S. PATENT NO. 8,234,705 B1**

TABLE OF CONTENTS

PETITIONER’S EXHIBIT LIST	iii
I. INTRODUCTION	1
II. MANDATORY NOTICES	1
A. Real Party-in-Interest	1
B. Related Matters.....	1
C. Lead and Back-up Counsel and Service Information	2
III. GROUNDS FOR STANDING.....	3
IV. NOTES.....	3
A. Discussion of Prior IPR Proceedings	3
B. Citations and Formatting Notes	4
V. SUMMARY OF THE ’705 PATENT	4
VI. LEVEL OF ORDINARY SKILL IN THE ART	5
VII. CLAIM CONSTRUCTION	6
A. “ <i>trusted computing base</i> ”	6
B. “ <i>trusted platform module</i> ”	8
C. Claim Construction in Related Matters.....	11
VIII. RELIEF REQUESTED AND THE REASONS FOR THE REQUESTED RELIEF	12
IX. IDENTIFICATION OF HOW THE CLAIMS ARE UNPATENTABLE.....	13
A. Challenged Claims	13
B. Statutory Grounds for Challenges.....	13
C. Ground 1	14

1.	Gleichauf.....	14
2.	Ovadia	15
3.	Lewis	16
4.	Reasons to Combine Gleichauf and Ovadia	17
5.	Reasons to Combine Gleichauf, Ovadia, and Lewis	27
6.	Claim 1	32
7.	Claim 2.....	58
8.	Claim 3	61
9.	Claim 5.....	62
10.	Claim 6.....	64
11.	Claim 7.....	66
12.	Claim 8.....	68
13.	Claim 9	69
14.	Claim 10.....	69
15.	Claim 11	70
16.	Claim 12.....	71
17.	Claims 13 and 15-18	72
18.	Claim 19.....	72
X.	CONCLUSION.....	73

PETITIONER’S EXHIBIT LIST

Ex.1001	U.S. 8,234,705
Ex.1002	Prosecution History of U.S. 8,234,705
Ex.1003	Declaration of A.L. Narasimha Reddy, Ph.D., Under 37 C.F.R. § 1.68 in Support of Petition for <i>Inter Partes</i> Review
Ex.1004	<i>Curriculum Vitae</i> of Narasimha Reddy, Ph.D.
Ex.1005	U.S. Patent No. 9,436,820 to Gleichauf et al.
Ex.1006	U.S. Patent No. 7,747,862 to Ovadia
Ex.1007	U.S. Patent No. 7,533,407 to Lewis et al.
Ex.1008	U.S. Patent No. 7,568,107 to Rathi et al.
Ex.1009	U.S. Patent Pub. No. 2004/0177276 to MacKinnon et al.
Ex.1010	U.S. Patent Pub. No. 2006/0005009 to Ball et al.
Ex.1011	U.S. Patent No. 6,829,654 to Jungck
Ex.1012	TCG Specification Architecture Overview (Rev. 1.2 Apr. 28, 2004), <i>available at</i> : https://web.archive.org/web/20061003162035if_/https://www.trustedcomputinggroup.org/groups/TCG_1_0_Architecture_Overview.pdf
Ex.1013	TPM Main – Part 1 Design Principles, Specification Version 1.2 (Rev. 62 Oct. 2, 2003), <i>available at</i> : https://trustedcomputinggroup.org/wp-content/uploads/tpmwg-mainrev62_Part1_Design_Principles.pdf
Ex.1014	U.S. Patent Pub. No. 2003/0061494 to Girard et al.
Ex.1015	U.S. Patent Pub. No. 2005/0078668 to Wittenberg et al.
Ex.1016	U.S. Patent No. 7,571,460 to Danforth et al.
Ex.1017	U.S. Patent Pub. No. 2005/0097199 to Woodard et al.

Ex.1018	TCPA Main Specification (Version 1.1b, 2003), available at: https://trustedcomputinggroup.org/wp-content/uploads/TCPA_Main_TCG_Architecture_v1_1b.pdf
Ex.1019	Plaintiff's Preliminary Infringement Contentions, <i>K.Mizra LLC v. Cisco Systems, Inc.</i> , No. 6-20-cv-01031 (W.D. Tex. Feb. 5, 2021) (without exhibits).
Ex.1020	RESERVED
Ex.1021	U.S. Patent Pub. No. 2005/0033987 to Yan et al.
Ex.1022	RESERVED
Ex.1023	U.S. Patent No. 7,330,977 to Cromer et al.
Ex.1024	IPR2021-00593, Final Written Decision

I. INTRODUCTION

Netskope, Inc. (“Petitioner”) respectfully requests that the Board review and cancel as unpatentable under (pre-AIA) 35 U.S.C. §103(a) claims 1-3, 5-13, and 15-19 (hereinafter, the “Challenged Claims”) of U.S. 8,234,705 (the “’705 patent,” Ex.1001).

II. MANDATORY NOTICES

A. Real Party-in-Interest

Pursuant to 37 C.F.R. § 42.8(b)(1), Petitioner certifies that the real party-in-interest is Netskope, Inc.

B. Related Matters

Pursuant to 37 C.F.R. § 42.8(b)(2), to the best knowledge of the Petitioner, the ’705 patent is or was involved in the following cases:

Case Heading	Number	Court	Filed
<i>K.Mizra LLC v. Google LLC</i>	1-25-cv-00236	WDTX	February 18, 2025
<i>K.Mizra LLC v. SonicWall Inc.</i>	1-25-cv-00047	DDE	January 10, 2025
<i>Hewlett Packard Enterprise Co. v. K.Mizra LLC</i>	23-1183	CAFC	November 23, 2022
<i>Cisco Systems, Inc. v. K.Mizra LLC</i>	22-2290	CAFC	September 29, 2022
<i>Forescout Technologies, Inc. v. K.Mizra LLC</i>	IPR2022-00081	PTAB	October 22, 2021
<i>Hewlett Packard Enterprise Co. v. K.Mizra LLC</i>	IPR2022-00084	PTAB	October 22, 2021

<i>K.Mizra LLC v. Hewlett Packard Enterprise Co. et al</i>	2-21-cv-00305	EDTX	August 9, 2021
<i>K. Mizra LLC v. Fortinet, Inc.</i>	2-21-cv-00249	EDTX	July 8, 2021
<i>K. Mizra LLC v. Forescout Tech., Inc.</i>	2-21-cv-00248	EDTX	July 8, 2021
<i>Cisco Systems, Inc. v. K. Mizra LLC</i>	IPR2021-00593	PTAB	March 15, 2021
<i>Mizra LLC v. Cisco Systems, Inc.</i>	6-20-cv-01031	WDTX	November 6, 2020
<i>Network Security Technologies, LLC v. Bradford Networks, Inc.</i>	1-17-cv-01487	DDE	October 24, 2017
<i>Network Security Technologies, LLC v. ForeScout Technologies, Inc.</i>	1-17-cv-01488	DDE	October 24, 2017
<i>Network Security Technologies, LLC v. McAfee, Inc.</i>	1-17-cv-01489	DDE	October 24, 2017
<i>Network Security Technologies, LLC v. Pulse Secure, LLC</i>	1-17-cv-01490	DDE	October 24, 2017
<i>K.Mizra LLC v. Citrix Systems, Inc. et al</i>	0-25-cv-60803	SDFL	April 24, 2025
<i>Netskope, Inc. v. K.Mizra LLC</i>	3-25-cv-04833	NDCA	June 6, 2025

C. Lead and Back-up Counsel and Service Information

Lead counsel: Thomas Millikan (Reg No. 72,316).

Back-up counsel: Babak Tehrani (Reg. No. 55,937), Hessam Gharaviram

5 (pro hac vice to be requested), Andrew Klein (pro hac vice to be requested).

These attorneys can be reached by mail at Perkins Coie LLP, 11452 El Camino Real, Suite 300, San Diego, CA 92130; by phone at (858) 720-5700; and by fax at (858) 720-5799.

Petitioner consents to electronic service. All services and communications to the attorneys listed above may be sent to:

Netskope-MizraIPRService@perkinscoie.com

A Power of Attorney is being filed concurrently.

5 **III. GROUNDS FOR STANDING**

Petitioner certifies that the '705 patent is eligible for IPR and that Petitioner is not barred or estopped from requesting IPR of the Challenged Claims. 37 C.F.R. § 42.104(a).

IV. NOTES

10 **A. Discussion of Prior IPR Proceedings**

The '705 patent was previously challenged in IPR2021-00593 filed by Cisco Systems ("Cisco") on March 15, 2021. The IPR was instituted on September 24, 2021 based on the combination Gleichauf, Ovadia and Lewis that are also used in this Petition. IPR2021-00593, Paper 9. On October 22, 2021, two other Petitioners, 15 Hewlett Packard Enterprise Co. ("HPE) and Forescout Technologies, Inc. ("Forescout"), filed IPR2022-00084 and IPR2022-00081, based on identical grounds, same arguments and same supporting evidence, along with a joinder motion, which was granted. IPR2022-00084, Papers 1, 3, 10; IPR2022-00081, Papers 1, 3, 14.

20 A final written decision was issued on September 19, 2022, finding the claims not unpatentable. IPR2021-00593, Paper 41. Pursuant to an appeal filed by

Cisco, the U.S. Court of Appeals for the Federal Circuit, on August 16, 2024,

“vacate[d] the Board’s motivation to combine analysis, which was rooted in legal

error and a fact finding un-supported by substantial evidence. ... [and] further

vacate[d] the Board’s ultimate determination that Cisco failed to show the

5 unpatentability of the challenged claims of the ’705 patent and remand[ed] for the

Board to consider the remaining issues regarding the obviousness of the challenged

claims, including Cisco’s non-benefits-based motivation to combine arguments and

Cisco’s fourth rationale.” *Id.*, Paper 44, 13. Upon remand, the IPR was terminated

pursuant to joint motions by Cisco, HPE, Forescout and Patent Owner. *Id.*, Paper

10 51.

The instant Petition, while relying on the combination of the same references

Gleichauf, Ovadia and Lewis, provides further detailed analysis and evidence

regarding motivations to combine, and additional mappings and discussions that

were not part of the prior IPRs, and thus provides compelling reasons for finding

15 the challenged claims unpatentable.

B. Citations and Formatting Notes

Petition cites to exhibits’ original page numbers. All **emphasis** has been

added unless otherwise indicated. Claim terms are presented in *italics*.

V. SUMMARY OF THE ’705 PATENT

20 The ’705 patent describes attempting to defend a computer network against

threats by analyzing the security state of a device that seeks network access. The device is quarantined if its security state is determined to be inadequate. Ex.1001, Abstract. A device in quarantine “is provided only limited access to the protected network.” *Id.*, 3:9-13. A quarantine notification page, i.e., a webpage, is displayed to the device and “provides notification that the computer is quarantined, and/or provides links to remediation sites appropriate to the quarantine such as a link to a site that provides anti-contagion software for removing a virus that the quarantined computer is believed to contain.” *Id.*, 15:66-16:7. The device is “permitted to access the protected network only as required to remedy a condition that caused the quarantine to be imposed.” *Id.*, 3:14-17.

These concepts were well known before the priority date of the '705 patent.

VI. LEVEL OF ORDINARY SKILL IN THE ART

A Person of Ordinary Skill in The Art (“POSITA”) in September 2004 would have had a working knowledge of the network communication art that is pertinent to the '705 patent, including network security methodologies. Ex.1003, ¶ 19. A POSITA would have had a bachelor’s degree in computer science, computer engineering, electrical engineering, or an equivalent training, and approximately two years of professional experience in the field of network communications, and more specifically, network security. *Id.* Lack of professional experience can be remedied by additional education, and vice versa. *Id.*

VII. CLAIM CONSTRUCTION

Claim terms in IPR are construed according to their “ordinary and customary meaning” to those of skill in the art. 37 C.F.R. § 42.100(b). For the purposes of this proceeding and the ground presented herein, Petitioner proposes construing the terms below. Constructions are proposed “only to the extent necessary.” *Nidec Motor Corp. v. Zhongshan Broad Ocean Motor Co.*, 868 F.3d 1013, 1017 (Fed. Cir. 2017).

A. “*trusted computing base*”

Claims 1, 12, and 19 recite in part a “*trusted computing base*.” For example, claim 1 recites “*an attestation that the **trusted computing base** has ascertained the presence of a patch or a patch level associated with a software component on the first host.*”

The ’705 patent describes examples of trusted computing bases, which it distinguishes from other components that could respond to a cleanliness query, such as anti-contagion software or an operating system:

A computer associated with an address identified in a list is queried for a cleanliness assertion (1302), for example by contacting a **trusted computing base** within a computer, and requesting an authenticated infestation scan by trusted software. An example of a **trusted computing base** within a computer is the Paladium security initiative under development by Microsoft and supported by Intel and American Micro Devices. Another example of a **trusted computing base** is

described in various TCG specifications, such as the TCG Architecture Overview, published by the Trusted Computing Group. Trusted code bases may for example execute antivirus scans of the remainder of the computer, including untrusted portions of the disk and/or operating system. In some embodiments, trusted code bases may digitally sign assertions about the cleanliness (e.g. infestation status) and/or state of their computers. In some embodiments, the query for cleanliness (1302) may be responded to by anti-contagion software, such as antivirus software, with assertions about the currency of a scan, such as the last time a scan was performed, or a version associated with a current anti-contagion software or definition file in use, wherein a sufficiently updated software and/or scan may act as a cleanliness assertion. In some embodiments, an operating system may respond with information associated with its patch level, wherein a sufficiently recent patch level may be interpreted as an assertion of cleanliness.

Ex.1001, 13:64-14:22.

During prosecution of the '705 patent, the applicants argued that *trusted computing base* is a term of art whose definition was provided on Wikipedia:

Applicant notes that “trusted computing base” and “trusted platform module[”] are terms of art with specific meanings that are in no way discussed in Liang.... As for the term of art “trusted computing base,” the final paragraph of the “Definition and characterization” section of the Wikipedia entry on “Trusted Computing Base” (http://en.wikipedia.org/wiki/Trusted_computing_base; retrieved January 10, 2010) states that “A given piece of hardware or software is

a part of the TCB if and only if it has been designed to be a part of the mechanism that provides its security to the computer system.”

Ex.1002, pp. 183-184.

The term “*trusted computing base*” should be construed according to the express

5 definition provided by the applicants during prosecution. Ex.1003, ¶¶ 38-41. Thus,

“*trusted computing base*” should be construed as *a piece of hardware or software*

that has been designed to be part of a mechanism that provides security to a

computer system. Id., ¶ 41.

B. “*trusted platform module*”

10 Claims 1, 12, and 19 recite in part a “*trusted platform module*.” For example,

claim 1 recites “*contacting a trusted computing base associated with a **trusted platform module** within the first host.*”

The detailed description does not use the term *trusted platform module*.

During prosecution of the ’705 patent, the applicants argued that *trusted platform*

15 *module* is a term of art that specifically refers to a secure cryptoprocessor that,

among other things, implements a standard of the same name promulgated by the

Trusted Computing Group:

Applicant notes that “trusted computing base” and “trusted platform module[”] are terms of art with specific meanings that are in no way

20 discussed in Liang. The Wikipedia entry for “Trusted Platform Module

([http://en.wikipedia.org/wiki/Trusted platform module](http://en.wikipedia.org/wiki/Trusted_platform_module); retrieved

January 10, 2010) begins, “In computing, Trusted Platform Module (TPM) is both the name of a published specification detailing a secure cryptoprocessor that can store cryptographic keys that protect information, as well as the general name of implementations of that specification, often called the “TPM chip” or “TPM Security Device” (as designated in certain Dell BIOS settings[1]). The TPM specification is the work of the Trusted Computing Group. The current version of the TPM specification is 1.2 Revision 103, published on July 9, 2007.” Reinforcing the industry adoption of this technology and standardization around the term of art used in the claims, this specification has also been adopted as ISO/IEC standard 11889.

Ex.1002, pp. 183-184.

Consistent with the applicants’ arguments, the Trusted Platform Module specification describes a trusted platform module that includes various components for performing cryptographic operations and storing cryptographic information.

Major components are depicted in figures reproduced below:

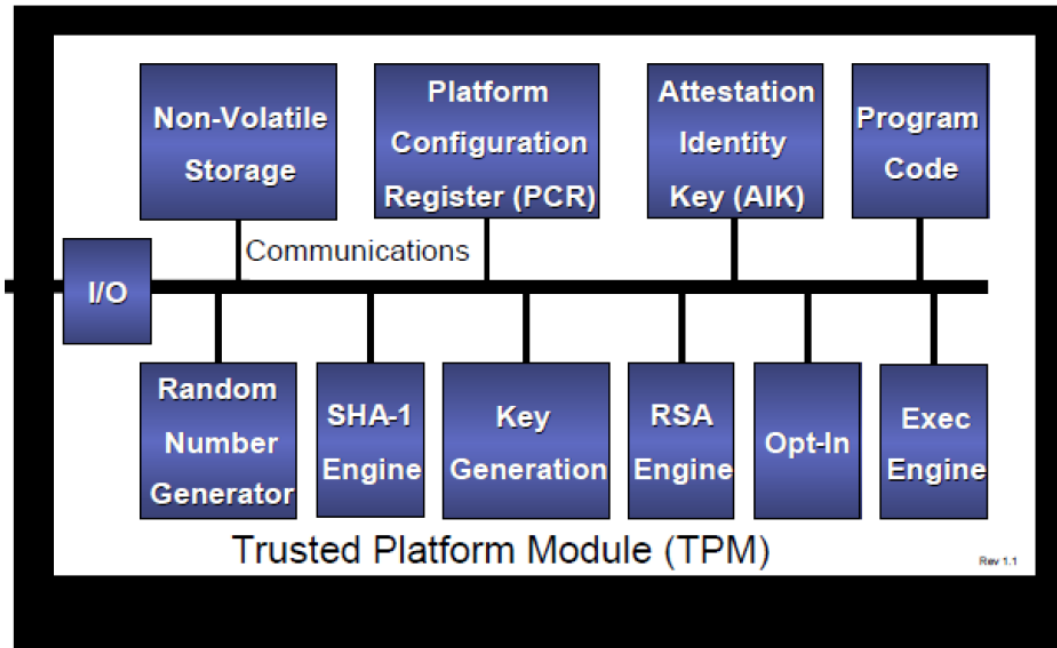
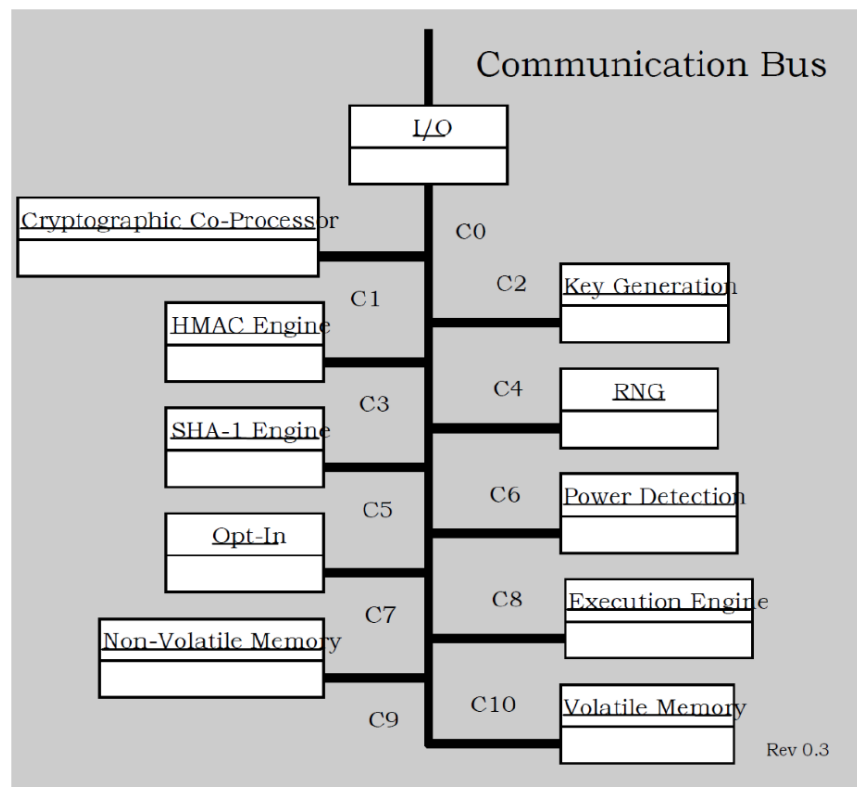


Figure 4.g — TPM Component Architecture

Ex.1012, 19.



Ex.1013, 11.

According to the Trusted Platform Module specification, the components of a trusted platform module include an input/output (I/O) component, a key generation component, a random number generator (RNG), a SHA-1 engine, an opt-in component, an execution engine, and memory (volatile and non-volatile).

5 Ex.1013, 11-22; Ex.1012, 19-20. Consistent with the applicants' statement—that a trusted platform module “store[s] cryptographic keys that protect information”—the Trusted Platform Module specification provides that a compliant trusted platform module stores cryptographic keys in memory. *See* Ex.1012, 19 (“Non-volatile storage is used to store Endorsement Key (EK), Storage Root Key
10 (SRK), owner authorization data and persistent flags.”); *see also* Ex.1013, 19 (“Non-volatile memory component ... is used to store persistent identity and state associated with the TPM.”)

The term “*trusted platform module*” should be construed according to the express definition provided by the applicants during prosecution. Ex.1003, ¶¶ 42-
15 46. Thus, “*trusted platform module*” should be construed as *a secure cryptoprocessor that implements the Trusted Platform Module specification from the Trusted Computing Group. Id.*, ¶ 46.

C. Claim Construction in Related Matters

The terms “*trusted computing base*” and “*trusted platform module*” have
20 previously been addressed in related litigation involving the '705 patent. In

K.Mizra v. Cisco, the parties stipulated that “*trusted computing base*” means “*hardware or software that has been designed to be a part of the mechanism that provides security to a computer.*” See *K.Mizra LLC v. Cisco Systems, Inc.*, No. 6:20-cv-01031 (W.D. Tex. Nov. 6, 2020). Regarding “*trusted platform module*,” while the parties initially submitted a joint claim construction statement proposing competing constructions, the district court ultimately adopted the following construction: “A secure cryptoprocessor that can store cryptographic keys and that implements the Trusted Platform Module specification from the Trusted Computing Group.” *Id.*, Dkt. No. 46, at 1.

In a related IPR (IPR2021-00593), the parties again addressed these same terms. Petitioner urged that “trusted computing base” should be construed as “a piece of hardware or software that has been designed to be part of a mechanism that provides security to a computer system,” pointing to the patentee’s prosecution statements. IPR2021-00593, Petition at 11-13. Patent Owner did not contest that construction. IPR2021-00593, Patent Owner Response at 5. The parties disputed construction of “trusted platform module,” but the Board found that they the parties did “not identify any issue that would turn on the construction of this term” and therefore determined that “no claim terms required express construction.” IPR2021-00593, Final Written Decision (Ex.102), at 14.

VIII. RELIEF REQUESTED AND THE REASONS FOR THE REQUESTED RELIEF

Petitioner asks that the Board institute a trial for *inter partes* review and cancel the Challenged Claims in view of the analysis below.

IX. IDENTIFICATION OF HOW THE CLAIMS ARE UNPATENTABLE

A. Challenged Claims

5 The Challenged Claims are claims 1-3, 5-13, and 15-19.

B. Statutory Grounds for Challenges

Ground	Claims	Basis
#1	1-3, 5-13, 15-19	35 U.S.C. § 103 (Pre-AIA) over Gleichauf in view of Ovadia and Lewis

U.S. Patent No. 9,436,820 to **Gleichauf** et al. (Ex.1005, “Gleichauf”) was filed on August 2, 2004.

10 U.S. Patent No. 7,747,862 to **Ovadia** (Ex.1006, “Ovadia”) was filed on June 28, 2004.

U.S. Patent No. 7,533,407 to **Lewis** et al. (Ex.1007 “Lewis”) was filed on April 14, 2004.

Gleichauf, Ovadia, and Lewis are prior art under 35 U.S.C. § 102(e).

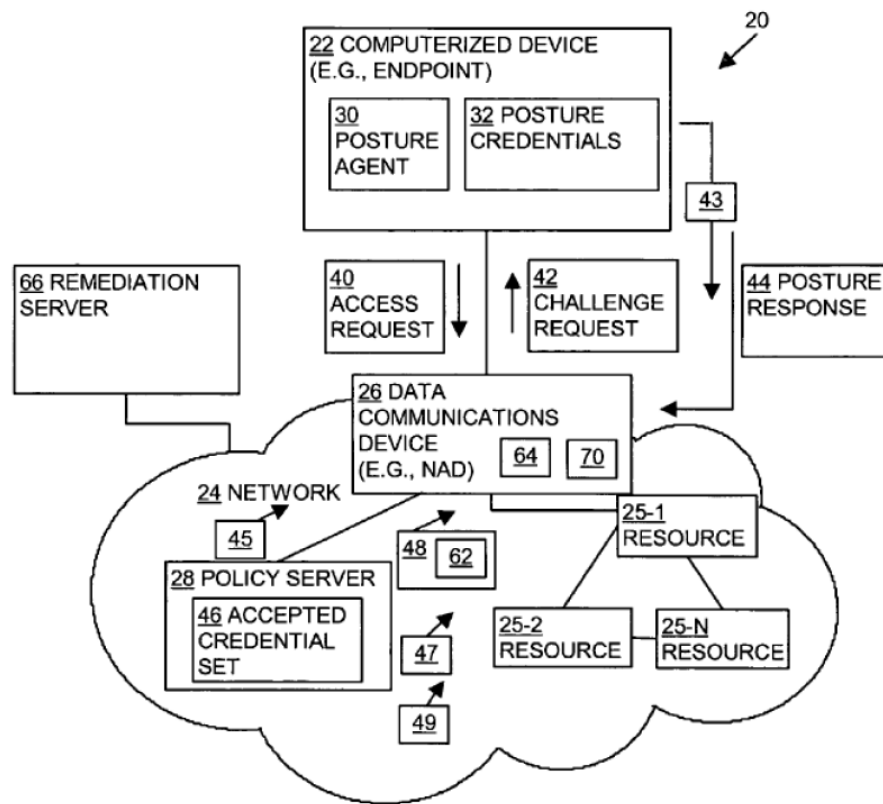
15 Petitioner’s analysis also cites additional prior art to demonstrate the knowledge of a POSITA and to provide contemporaneous context to support Petitioner’s assertions regarding, what a POSITA would have understood from the asserted prior art. *See Yeda Research v. Mylan Pharm. Inc.*, 906 F.3d 1031, 1041–1042 (Fed. Cir. 2018) (affirming the use of “supporting evidence relied upon

to support the challenge”): 37 C.F.R. § 42.104(b): *see also K/S HIMPP v. Hear-Wear Techs., LLC*, 751 F.3d 1362, 1365-66 (Fed. Cir. 2014); *Arendi S.A.R.L. v. Apple Inc.*, 832 F.3d 1355, 1363 (Fed. Cir. 2016).

C. Ground 1

1. Gleichauf

Gleichauf generally relates to the field of network security. Ex.1005, 7:55 61; Ex.1003, ¶ 53. Gleichauf teaches controlling a computerized device’s access to network resources based on the security posture of the device. Ex.1005, 3:4-9; Ex.1003 ¶ 53. Gleichauf illustrates a data communications system in Figure 1 that includes a device that is attempting to access a network resource on the network:



Ex.1005, Fig. 1.

Gleichauf's policy server controls the device's access to the network.

Ex.1005, 7:4-8, 10:20-23; Ex.1003, ¶ 54. When the policy server detects a security issue with a device, e.g., an out-of-date anti-virus application, the device is quarantined. Ex.1005, 15:55-61; Ex.1003, ¶ 55. While in quarantine, the device has limited network access but is still permitted to access a remediation server with security updates suitable to bring the device into compliance with network access security protocols. Ex.1005, 16:13-21; Ex.1003, ¶ 55. Normal access is generally granted after the device is updated. Ex.1005, 16:40-43; Ex.1003, ¶ 55.

Gleichauf was not before the Examiner during prosecution of the '705 patent.

2. Ovadia

Like Gleichauf, Ovadia discloses network security measures and describes steps to ensure that a device seeking to access a network has an authorized configuration. Ex.1006, 1:8-11, Abstract; Ex.1003, ¶ 57. Ovadia's security scheme generates security keys, including attestation identity keys ("AIKs"), using a trusted platform module. Ex.1006, Abstract; Ex.1003, ¶ 57. The trusted platform module is used "to verify [that] a current configuration of a subscriber station platform is an authorized configuration." Ex.1006, Abstract; Ex.1003, ¶ 57. A subscriber station seeking network access "sends authentication information including a manifest signed with the SS's [subscriber station's] private AIK key."

Ex.1006, Abstract; Ex.1003, ¶ 57. An authentication server receives the information and authenticates the subscriber station via the subscriber station's public AIK key. Ex.1006, Abstract; Ex.1003, ¶ 57.

Ovadia was not before the Examiner during prosecution of the '705 patent.

5 **3. Lewis**

Like Gleichauf, Lewis describes network security measures, including verifying the configuration state of a device that seeks to access a network. Ex.1007, 4:7-9; Ex.1003, ¶ 59. Lewis provides “a system for ensuring that machines having invalid or corrupt states are restricted from accessing network
10 resources.” Ex.1007, 4:7-9; Ex.1003, ¶ 59. Lewis describes how the device must provide a “bill of health” to a quarantine server before gaining access to the network. Ex.1007, 4:13-15; Ex.1003, ¶ 59. The quarantine server determines if the bill of health is valid or invalid. Ex.1007, 4:16-23; Ex.1003, ¶ 59. A valid bill of health indicates that the device has up-to-date security software, and the device is
15 granted access to the network. Ex.1007, 4:18-21; Ex.1003, ¶ 59. An invalid bill of health indicates that the device lacks security measures sufficient for access to the network, and the device's network configuration is put in a lock-down configuration. Ex.1007, 13:44-52; Ex.1003, ¶ 59.

The lock-down configuration puts the device in quarantine. Ex.1007,
20 13:63-65; Ex.1003, ¶ 60. A quarantined device's communications are restricted,

but it is permitted to access the quarantine server that provides a webpage to inform the “user that the client [device] is in quarantine and corrective action must be taken.” Ex.1007, 10:61-66; Ex.1003, ¶ 60. If the user performs the corrective action and submits an updated bill of health that is valid, the device is permitted
5 access to the network. Ex.1007, 13:12-15; Ex.1003, ¶ 60.

During prosecution of the ’705 patent, Lewis’s corresponding application publication, U.S. Patent Pub. No. 2005/0131997, was cited by the Examiner for its teaching of a quarantine server providing a quarantine notification page. Ex.1002, 98-99. The applicants did not dispute this, arguing instead that the Examiner’s
10 cited art (including Lewis) failed to render obvious other features recited in then-pending claim 1. Ex.1002 at 73-74; Ex.1003, ¶ 61. Petitioner is relying on Lewis for the same point asserted by the Examiner and conceded by the applicants. As such, Lewis is cited for a point of agreement that was not overcome during prosecution. There is no basis for the Board to exercise its 35 U.S.C. § 325(d)
15 discretion on these facts.

4. Reasons to Combine Gleichauf and Ovadia

A POSITA when considering the teachings of Gleichauf would have also considered the teachings of Ovadia, as both relate to providing secure access when an individual computerized device connects to a network. Ex.1005, 3:4-9; Ex.1006,
20 1:8-11; Ex.1003, ¶ 62. A POSITA would have been motivated to combine the

teachings of Gleichauf and Ovadia for multiple reasons and doing so would have been within the POSITA's skillset. Ex.1003, ¶ 63. The combination would have been obvious because it merely uses known techniques (e.g., Ovadia's trusted platform module security infrastructure and digital signatures, which were

5 industry-standard security techniques) to improve a similar method (e.g., Gleichauf's authentication and authorization of devices seeking network access) in the same way. Ex.1003, ¶ 63.

Notably, this Petition relies on two related and specific teachings of Ovadia in the combination: (1) the use of the TPM and (2) digitally signing the attestation
10 of cleanliness. Both aspects relate to enhancing the security of communications between Gleichauf's electronic device 22 and its data communications device 26, and reducing the vulnerability of Gleichauf's system to malicious attacks, viruses, worms, and the like which conform to Gleichauf's goal of "limiting vulnerable computerized devices from accessing the network resources and minimizing the
15 risk that the network resources receive or transmit malware." Ex.1001, Abstract; Ex.1003, ¶ 64.

A POSITA would have been motivated to take advantage of Ovadia's use of a trusted platform module—as described in the Trusted Computing Group (TCG) "platform-independent industry specification"—for multiple reasons including
20 because it would standardize Gleichauf's posture credentials. Ex.1006, 4:32-35;

Ex.1003, ¶ 65; *see also* Ex.1012, 2 (TCG was formed as an “industry-standards organization with the aim of enhancing the security of the computing environment in disparate computer platforms.”). A POSITA would have known the benefits of using a standardized security platform in implementing or augmenting the protocols of Gleichauf for several reasons.

First, a standardized security platform would have been less vulnerable to design flaws and security backdoors since such platforms were subject to extensive security testing by experts in the field before becoming an industry standard.

Ex.1003, ¶¶ 33, 66. Accordingly, a POSITA would have found it beneficial to use the standardized TPM in Gleichauf to reduce security vulnerabilities of non-standardized platforms.

Second, a POSITA would have found motivation in Ovadia’s description of the TPM that was adopted by the TCG with its main specification defining “a trusted platform subsystem that employs cryptographic methods when establishing trust.” Thus, a POSITA would have been guided by the teachings of Ovadia and would have understood that using the standardized TPM would have enhanced the trustworthiness of the information stored at Gleichauf’s electronic devices and the exchange of messages between Gleichauf’s electronic device and its data communication device. As one example, the TPM included tamper-detection features both as physical markers and as software-implemented features that

enhanced the trustworthiness of the electronic device. Ex.1013, 97 (“If the TPM discovers tampering with the tick count (through timing changes etc.) the TPM MUST treat this as an attack and shut down further TPM processing as if a self-test had failed.”); *see also id.*, 45 (“Any tamper-resistance markers... The tests on the

5 tamper-resistance or tamper-evident markers are under programmable control.

There is no requirement to check tamper-evident tape or the status of epoxy surrounding the case.”); Ex.1003, ¶ 67. Such features of the TPM would have identified electronic devices that were tampered with, and would have therefore aided with Gleichauf’s objective of “limiting vulnerable computerized devices from accessing the network resources and minimizing the risk that the network resources receive or transmit malware.” Ex.1005, Abstract, *see also id.*, 12:7-12, 12:27-31; Ex.1003, ¶ 67.

Third, a POSITA would have understood that the TCG-defined TPM can advantageously be used to provide additional features and functionalities that included asset tracking functionalities, collect and report (in a trusted manner) platform configuration information, and secure client-server communications.

Ex.1012, 2-4; *see also* Ex.1014, [0011] (“The TCPA architecture includes an isolated computing engine, or trusted platform module (TPM) (not shown), whose processes can be trusted because they cannot be altered. ... Additionally, the TPM

20 may encrypt or wrap data, such as a key, and may bind or seal an internally

generated asymmetric key pair to a particular TPM and/or a particular platform configuration.”); Ex.1023, 2:8-11 (TCPA is the former name of TCG); Ex.1021, [0003]-[0004] (same); Ex.1003, ¶ 68.

Fourth, following the industry standard, as described in Ovadia, would have benefited Gleichauf’s system by formatting a device’s posture credentials consistently and with substantially the same information, regardless of the device’s manufacturer or components. Ex.1003, ¶ 69. This would also allow Gleichauf’s policy server to process posture credentials that were received, in a trusted manner, more efficiently and in substantially the same way regardless of the device returning the credentials. *Id*; see also *id.* ¶ 33; Ex.1018, 1. Following the industry standard would also promote the interoperability of devices from diverse manufacturers. Ex.1003, ¶ 69. Ensuring the interoperability of devices is a common and important goal in the computer networking arts. *Id.*

Ovadia describes applying technology from the “Trusted Computing Group (TCG)” which addresses “network security” and “covers trust in computing platforms in general.” Ex.1006, 4:16-35. A POSITA would thus have understood that the trusted computing and authentication technologies described in Ovadia are not limited to the wireless networks Ovadia describes but instead are generally applicable to networks—like Gleichauf’s—that condition network access on authentication and authorization of client devices. Ex.1003, ¶ 70.

Fifth, the combination also would have been obvious because it applies the known technique of digitally signing an attestation about the state of a device to make its authentication more secure (as in Ovadia) with the known network authorization process of analyzing a device's security posture credentials,

5 including its authentication (as in Gleichauf) to yield predictable results. Ex.1003,

¶ 71. A POSITA would have been familiar with the use of digital signatures to prove the integrity and authenticity of network communications and would have recognized that Ovadia's digital signature technique would improve Gleichauf's process by increasing resistance to forged or tampered-with posture credentials.

10 Ex.1010, [0033]; Ex.1003, ¶ 71. In fact, at the time of the alleged invention in 2004, it was well-known by POSITAs that digital signatures were routinely used to verify the authenticity of messages or documents. *Id.* And, therefore, a POSITA also would have recognized that use of a digital signature would improve Gleichauf's process by allowing the policy server to confirm that the posture
15 credentials originated with the device attempting to access the network. *Id.*

Sixth, a POSITA also would have been motivated to combine the teachings of Gleichauf and Ovadia because Ovadia provides additional details regarding implementation of Gleichauf's techniques. Ex.1003, ¶ 72. Gleichauf provides that "policy server 28 authenticates an identity of the computerized device 22 to
20 generate an authentication result." Ex.1005, 22:3-5. Ovadia explains that one way

to authenticate a device is to check that device's digital signature. Ex.1006, 16:30-32. It would have been obvious to a POSITA to use Ovadia's digital signatures as a way of performing Gleichauf's device authentication. Ex.1003, ¶ 72.

5 **Reasonable Expectation of Success:** A POSITA would have had a reasonable expectation of success in augmenting and improving Gleichauf's system based on the teachings of Ovadia that related to the use of TPM and digital signatures. Ex.1003, ¶ 73. Notably, Ovadia explains "[t]he trusted platform ... may be integrated into some existing platform component or components." Ex.1006, 10 4:38-40. Therefore, Ovadia already described that the TPM can be incorporated into an existing platform, such as Gleichauf's system. *Id.*

Furthermore, components of Ovadia's trusted computing architecture had corresponding components in Gleichauf's architecture, and a POSITA would have recognized how details provided for components in Ovadia would be applicable to 15 the corresponding components in Gleichauf. Ex.1003, ¶ 74. For example, similar to the posture agent in Gleichauf, which gathers information about the device's "configuration state," Ovadia uses "an authentication agent to determine the state of a platform environment" and generate an "integrity measurement." Ex.1005, 3:8, 12:21-31; Ex.1006, 4:40-43, 14:1-6; Fig. 8. Ovadia's integrity measurement is 20 stored in the trusted platform module. Ex.1006, 13:61-63, Fig. 8. The integrity

measurement is then provided to an authentication server as part of authentication data in response to an authentication challenge, which is analogous to Gleichauf's sending the posture credentials of a device to a policy server in response to requests to authenticate the device. Ex.1006, 4:43-45, 13:61-63, Fig. 8; Ex.1005, 5 10:63-11:3; Ex.1003, ¶ 75. Ovadia's integrity measurement is "digitally signed with appropriate keys" before it is sent to the authentication server. Ex.1006, 13:63-64, Fig. 8; Ex.1003, ¶ 75. A POSITA would have understood that Ovadia's trusted platform module generates and sends the integrity measurement because the integrity measurements are digitally signed with an Attestation Integrity Key 10 (AIK) and "AIKs are only permitted to sign data generated by a TPM." Ex.1006, 14:9, 5:35-36; Ex.1003, ¶ 76. It would have been obvious to a POSITA to implement Gleichauf's posture agent with trusted platform module functionality, e.g., the ability to store, provide, and digitally sign integrity measurements, as taught in Ovadia, to provide the greater security assurances available with digital 15 signatures and limited-use integrity keys. Ex.1003, ¶ 76.

Both references also describe providing a message with an explanation to the user of the device if the posture credentials or integrity measurement is not accepted. Ex.1005, 4:50-67; Ex.1006, 16:62-65; Ex.1003, ¶ 77. Therefore, not only Ovadia contemplated using the TPM in existing systems or components (Ex.1006, 20 4:38-40), the analogous architectures of Gleichauf's and Ovadia's respective

systems—together with the fact that they both solve similar problems within the same field of network access authentication—would have provided a POSITA with a reasonable expectation of success in making the combination. Ex.1003, ¶ 77. The predictability of their combination is also consistent with, and supported by,

5 Gleichauf’s and Ovadia’s mutual acknowledgements that a POSITA would have had the skills to make “various changes in form and details” to their described embodiments. Ex.1005, 27:24-29; Ex.1006, 18:64-19:4; Ex.1003, ¶ 77.

Petitioner’s proposed combinations permit but do not require physical incorporation of elements from Ovadia into Gleichauf. *See In re Mouttet*, 686 F.3d 10 1322, 1332 (Fed. Cir. 2012); *In re Etter*, 756 F.2d 852, 859 (Fed. Cir. 1985).

Notably, the TPM Main document makes it evident that the TPM specification provides design guidelines and generic information, and a POSITA designer would be able to use the specified guidelines to implement the TPM for specific platforms. Ex.1013, 12 (“The design principles give the basic concepts of the TPM 15 and generic information relative to TPM functionality. A TPM designer MUST review and implement the information in the TPM Main specification (parts 1-4) and review the platform specific document for the intended platform. The platform specific document will contain normative statements that affect the design and implementation of a TPM.”); Ex.1003, ¶ 78. Therefore, while this Petition does 20 provide specific combination details, an implementation-level explanation of the

combination is neither technically, nor legally, required to illustrate a reasonable expectation of success. Ex.1003, ¶¶ 78-79. As a prior panel has opined, “[t]he test for obviousness is not whether the features of a secondary reference may be bodily incorporated into the structure of the primary reference Rather, the test is what the combined teachings of the references would have suggested to those of ordinary skill in the art.” *MCM Portfolio LLC v. Hewlett-Packard Co.*, 812 F.3d 1284, 1294 (Fed. Cir. 2015) (quoting *In re Keller*, 642 F.2d 413, 425 (CCPA 1981).” IPR2022-00787, Paper 25, 55.

Additionally, Ovadia’s teachings are compatible with, and are directly relevant to the Gleichauf’s system.¹ Notably, Ovadia’s invention is related to “methods and apparatus to authenticate base and subscriber stations and secure sessions for broadband wireless networks ... [that] employs a trusted platform

¹ Patent Owner in a previous IPR proceeding erroneously alleges the goal of Ovadia is ensure that device configurations remain unchanged, which is purportedly opposite to Gleichauf’s goal of ensuring that the most up-to-date patches of antivirus software are installed on a computer. IPR2021-00593, Paper 23, 18-21. This assertion is wrong because it mischaracterizes the goal of Ovadia, and incorrectly implies that implementing security updates somehow contradicts integrity verification. Ex.1003. ¶ 80.

module (TPM) to generate security keys.” Ex.1006, Abstract. Ovadia’s process of establishing a secure communication session included establishing a broadband connection, using an authentication protocol that involved sending subscriber authentication information and authenticating it via a digital signature, and

5 culminated in establishing a secure channel for communication. *See e.g.*, Ex.1006, claim 1, Figures 3a-4b. As explained above, Ovadia, as part of its authentication procedures, employed a trusted platform module, and used a digital signature to ensure authenticity of the exchanged information. As also explained above, these were complementary to and enhanced the operations of Gleichauf’s system that

10 also relied on exchange of information between the policy server and the computing device that requested services, wherein “policy server 28 authenticate[d] an identity of the computerized device 22 to generate an authentication result.” Ex.1005, 22:3-5. Accordingly, a POSITA would have understood that Gleichauf’s system benefited from the complementary teachings of
15 Ovadia regarding the use of a TPM and digital signatures. Ex.1003, ¶ 80.

5. Reasons to Combine Gleichauf, Ovadia, and Lewis

A POSITA when considering the teachings of Gleichauf and Ovadia would have also considered the teachings of Lewis, as all of the references relate to providing secure access when an individual computerized device connects to a
20 network. Ex.1005, 3:4-9; Ex.1006, 1:8-11; Ex.1007, 4:7-9; Ex.1003, ¶ 81. A

POSITA would have been motivated to combine the teachings of Gleichauf, Ovadia, and Lewis for multiple reasons and doing so would have been within the POSITA's skillset. *Id.* The combination would have been obvious because it uses the known technique of redirection of device traffic comprising a DNS query to a quarantine server that serves a webpage to the device, as in Lewis, to improve a similar method of traffic redirection, as in Gleichauf, in the same way. *Id.*

Notably, while DNS queries were ubiquitous at the time of the alleged invention of the '705 patent in 2004 (*id.*, ¶¶ 28-29, 82), Gleichauf was silent about the specific steps to take when a request comprising a DNS query is received from a non-compliant electronic device. In this regard Lewis provided more specific details by explaining that a DNS query would result in resolution (or translation) of the query to an IP address of the quarantine server. A POSITA would have known that a DNS query included a name that needed to be resolved to an IP address, as this was (and is) the basic functionality of DNS resolution. *Id.* In response to a request from non-compliant electronic device, an obvious name resolution for a DNS query that did not specify a remediation host would have been, as Lewis explains, providing the IP address of the quarantine server. A POSITA would have found the combination of Gleichauf and Ovadia with Lewis obvious for several reasons. *Id.*

First, given that DNS queries were very common, a POSITA would have

expected to receive such queries, and would have benefited from Lewis's explicit teachings about responding to DNS queries in a manner that did not compromise the security of the Gleichauf-Ovadia system. *Id.*, ¶ 83.

Second, redirecting a DNS query to a quarantine server (part of the policy server in the Gleichauf-Ovadia-Lewis combination, as further explained below) would have been one of only a limited number of choices that a POSITA would have had when receiving a request from a compromised electronic device. Another choice would have been to resolve the DNS query to the IP address of the remediation server. However, redirection to the quarantine sever would have provided the quarantine server (policy server) an opportunity to re-assess the electronic device's lack of trustworthiness before deciding on the appropriate course of remediation.² *Id.*, ¶ 84. For example, a redirection to the quarantine server (policy server) would have allowed the policy server to determine if the electronic device were infected with additional viruses before providing

² In fact, Gleichauf describes that its policy server is able "to re-assess the posture credentials and apply a new access policy on the data communications device." Ex.1005, 5:60-67; see also *id.*, 5:13-16 ("the device will be again identified as being at some level of non-compliance and the appropriate response will be invoked by the policy server.").

instructions to access a particular remediation server appropriate for remediating the detected vulnerabilities. *Id.*

Third, a POSITA would have found the combination with Lewis advantageous because Lewis described serving a webpage with remediation instructions to a quarantined device. Notably, Lewis described redirection to the quarantine server would be triggered by the user opening a browser and attempting to access network resources. Ex.1007, 10:63-66; Ex.1003, ¶ 85. The webpage served by the quarantine server would be displayed in the browser that the user already has opened, which would advantageously avoid necessitating additional software components running on the device to receive and display the notification message to the user. Ex.1005, 4:56-60; Ex.1006, 16:62-65; Ex.1003, ¶¶ 85-86.

The Combination and Reasonable Expectation of Success: Lewis

explains that the quarantine server can be implemented “on a trusted machine in a network,” and specifically, on the “network server 203”. Ex.1007, 4:9-11, 11:16-

19. Consistent with Lewis’ teachings, in the Gleichauf-Ovadia-Lewis combination a POSITA would have understood that the relevant functionality of Lewis’s quarantine server would be similarly incorporated into Gleichauf’s policy server; notably, both Lewis’s network server and Gleichauf’s policy server were responsible for verifying the posture credentials (or the bill of health) of the user device that intended to access the network resources. Ex.1005, 10:30-33; Ex.1007,

12:1-15; Ex.1003, ¶ 87.

A POSITA would have had a reasonable expectation of success in combining the teachings of Gleichauf and Ovadia with Lewis. Ex.1003, ¶ 87. As explained, Gleichauf's policy server sends a notification message to a quarantined
5 device with information and remediation instructions. Ex.1005, 5:4-11. Put differently, the policy server identifies whether a device is to be quarantined and what actions should be taken to remediate the security issue before the device is allowed access to the network. Ex.1005, 7:3-8, 4:60-63. A POSITA would have found it straight-forward to provide the contents of Gleichauf's notification
10 message in a webpage since doing so would display the contents of the message to the user in the browser the user is already using. Ex.1003, ¶¶ 87-88.

The combination would have involved nominal changes to Gleichauf's system because the quarantine server's relevant functionalities would be incorporated into Gleichauf's policy server, which already had the capability to
15 verify the user device's posture credentials, and to generate various access restriction or redirect instructions. Ex.1005, 15:50-54.³ Therefore, incorporation of

³ Notably, the '705 patent describes the functionalities of its quarantine server primarily as "provid[ing] a notice and/or other information and/or instructions to a user of the quarantined host." Ex.1001, 12:9-11; *see also id.*, 15:1 ("provides

functionalities to provide DNS redirect and to serve a webpage did not include any significant architectural modifications to Gleichauf's system and would have been well within the capabilities of a POSITA. Ex.1003, ¶ 89. The combination would have also incorporated Ovadia's trusted platform module, as explained in Section IX.C.4.

6. Claim 1

[1.0] *A method for protecting a network, comprising:*

Gleichauf discloses a “computer-implemented **method** to control access to resources in a network.” Ex.1005, 27:61-62; Ex.1003, ¶¶ 91-92. Figure 1

reproduced in connection with Element 1.1 below illustrates a protected network

24. Ex.1005, 8:32-36; Fig. 1; Ex.1003, ¶ 98. Gleichauf's method protects the

network by decreasing the likelihood of malware or virus infection and ensuring

that the device attempting to access the network is authorized and is the device it

claims to be. Ex.1005, 5:55-60, 7:56-61, 8:16-31, 12:57-67; Ex.1003, ¶¶ 91-92. As

explained below, the combination of Gleichauf, Ovadia, and Lewis renders

obvious a method “*comprising*” the claimed steps.

[1.1] *detecting an insecure condition on a first host that has connected or is attempting to connect to a protected network, wherein detecting the insecure condition includes*

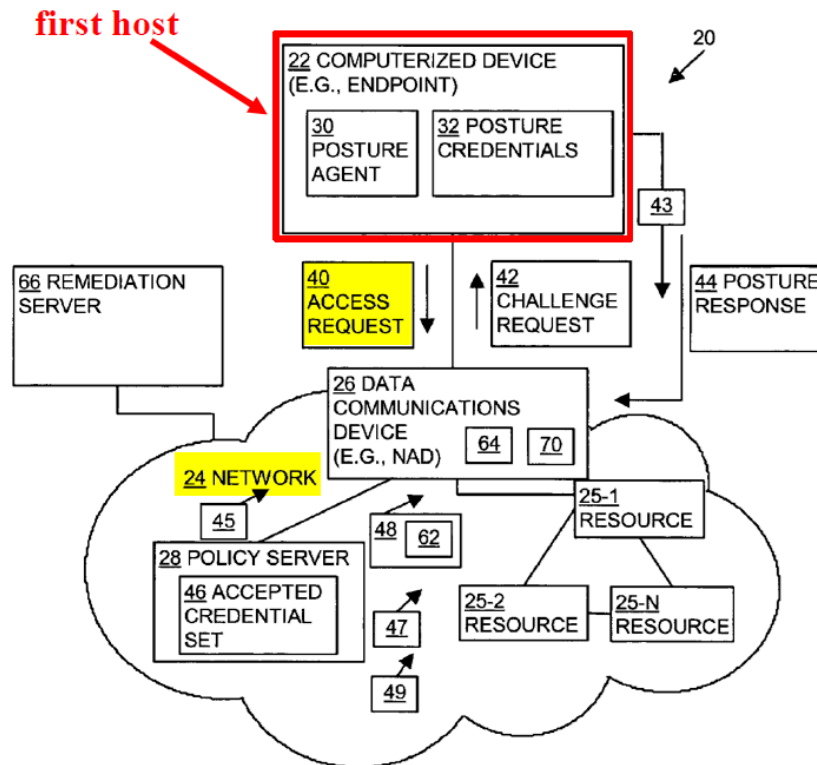
information and links to assist in remediation”). Ex.1003, ¶X.

First, Gleichauf describes a “computerized device, such as a personal computer attempting to communicate with restricted resources in the network.”

Ex.1005, 3:50-56; Ex.1005, 8:1-4; Ex.1003, ¶¶ 93-99. Figure 1 of Gleichauf

illustrates the computerized device 22 (*first host*) sending an access request to the

5 network 24 (“attempting to connect to a protected network”):



Ex.1005, Fig. 1 (annotated); Ex.1003, ¶ 98.

Gleichauf’s computerized device 22 is a *first host* that is attempting to access network resources 25 (Ex.1005, 8:38-40) that are protected by access control

10 measures; the network resources are part of a *protected network*. *Id.*, 6:39-43, 15:5-19; Ex.1003, ¶ 95. The access request, packet transmission, and attempt to access the restricted network resources show that the device has *connected* or *is*

attempting to connect to a protected network. Ex.1003, ¶ 95.

Second, Gleichauf's policy server *detects an insecure condition* in the computerized device. Ex.1003, ¶ 96. The device's communication attempt triggers a "posture based challenge-response" exchange where the device transmits its posture credentials to the network. Ex.1005, 11:29-32. The posture credentials contain "critical security information" including information about the "particular virus software and virus definition versions" installed on the device. Ex.1005, 3:36-45. The policy server analyzes these posture credentials and *detects* the absence "of an anti-virus application (e.g., or an up-to-date anti-virus application or virus definition file)." Ex.1005, 14:16-34; Ex.1003, ¶ 97. Devices that "are not configured with anti-virus or up-to-date anti-virus applications" are considered "vulnerable" because "network resources [may] become 'infected' with malware carried by the 'vulnerable' [computerized] user device 22." Ex.1005, 22:38-59, 11:58-12:13. A device would also be vulnerable to any malware that might be transmitted by network resources. Ex.1005, 1:10-14, 8:52-56. Thus, absence of an anti-virus application or up-to-date anti-virus application is an example of an *insecure condition*. Ex.1003, ¶ 97.

Gleichauf also describes other examples of a potential *insecure condition*, such as "the version of operating systems used by the user device 22 or the presence of updated 'patches' associated with the operating systems" and "when

was the last time a local antivirus scan was executed.” Ex.1005, 1:65-2:3, 8:65-9:1, 3:36-45, 31:1-7; Ex.1003, ¶ 98. Gleichauf explains that there may be “known vulnerabilities in the device operating system, resident applications, associated security programs, and their respective configurations.” Ex.1005, 1:65-2:3.

5 Additional examples of a potential *insecure condition* are discussed *infra* at [1.4] in relation to the *attestation of cleanliness*.

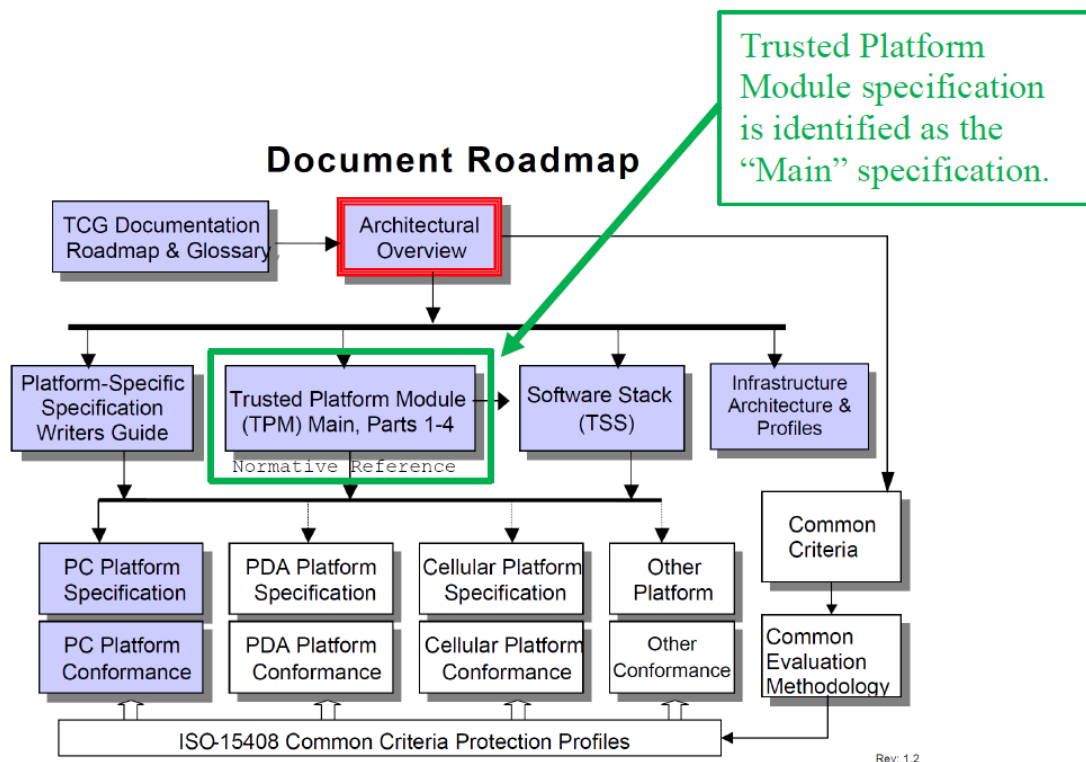
By gathering information relating to a computerized device’s vulnerabilities—such as the presence of up-to-date anti-virus application and operating system patches or when an anti-virus scan was last run—Gleichauf
10 renders obvious *detecting an insecure condition*. Ex.1003, ¶¶ 98-100.

[1.2] *contacting a trusted computing base associated with a trusted platform module within the first host,*

First, Gleichauf’s computerized device includes a “posture agent” and one or more “posture plug-ins” installed on the device that collect information about
15 the device’s security posture. Ex.1005, 3:9-16; Ex.1003, ¶¶ 101-102. The posture agent accesses information collected by each plug-in using an API implemented by each plug-in manufacturer and prepares posture credentials that it transmits to the policy server for analysis and access control. Ex.1005, 9:24-39; Ex.1003, ¶¶ 103-104. Gleichauf’s posture agent is “embodied as a software or hardware
20 application.” Ex.1005, 9:29-39. Since the posture credentials are used to secure the restricted resources, a POSITA would have understood that the combined

functionality of the posture agent and plug-ins is part of the mechanism that provides such security. Ex.1003, ¶ 105. Thus, the combined functionality of the posture agent and plug-ins is a *trusted computing base*. Ex.1003, ¶ 105; *see also supra* Section VII.A.

5 **Second**, Ovadia discloses a security apparatus that employs a trusted platform module implementing the Trusted Computing Group “main specification (Version 1.2, October 2003...)” Ex.1006, 4:16-47. The main specification from TCG is the Trusted Platform Module Main Specification (Ex.1003, ¶ 106):

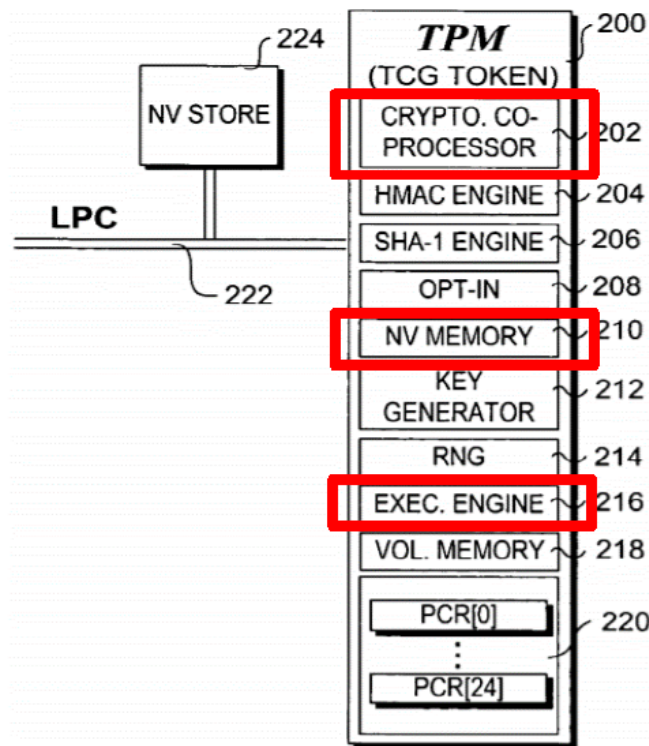


10 Ex.1012, iv (annotated); Ex.1013; Ex.1003, ¶ 106.

Ovadia’s trusted platform module implements the Trusted Platform Module main specification from the TCG and includes a “cryptographic co-processor,”

“non-volatile (NV) memory,” and an “execution engine.” Ex.1006, 4:46-55, Fig. 2; Ex.1003, ¶ 107. Since Ovadia’s trusted platform module implements the Trusted Platform Module main specification, it would have been obvious to a POSITA that the non-volatile memory stores cryptographic keys because that is how it is

5 described in the main specification. Ex.1013, 19; Ex.1012, 19; Ex.1003, ¶ 107.



Ex.1006, Fig. 2 (annotated): Ex.1003, ¶ 107.

Therefore, Ovadia’s trusted platform module is a *trusted platform module* as claimed. Ex.1003, ¶ 108: *see supra* Section VII.B. Ovadia’s trusted platform

10 module “may be embodied as a hardware device (most common) or via software.” Ex.1006, 4:24-26. Like Gleichauf’s posture agent, Ovadia’s trusted platform module is used “to verify [that] a current configuration of a subscriber station

platform is an authorized configuration.” Ex.1006 Abstract: Ex.1003, ¶ 108. The configuration (called the integrity measurement) including information about the subscriber station, including the “current measurement of the firmware and/or software” of the station. Ex.1006, 13:57-14:12; Ex.1003, ¶ 108.

5 As described in more detail in Section IX.C.4, a POSITA would have been motivated to combine the teachings of Gleichauf and Ovadia and doing so would have been within the POSITA’s skillset. Ex.1003, ¶ 112. A POSITA would have appreciated the value in implementing Gleichauf’s posture agent with trusted platform module functionality. Ex.1003, ¶ 112. The TCG, which prepares the TCG
10 specification discussed above, is “an **industry consortium** concerned with platform and network security.” Ex.1006, 4:31-32. The TCG specification is a “**platform-independent industry specification** that covers trust in computing platforms” as well as how to implement and use trusted platform modules. Ex.1006, 4:32-35, 4:22-24. Implementing Gleichauf’s posture agent with trusted
15 platform module functionality would provide consistency among posture credentials generated by such posture agents operating on various computerized devices. Ex.1003, ¶ 112.

 On the device side (e.g., device 22 of Gleichauf), a posture agent implemented with trusted platform module functionality would operate consistent
20 with the TCG specification, regardless of the device it is running on (given that

TPM is “platform independent,” Ex.1006, 4:32-35), and operation of such trusted platform modules would be consistent throughout the adopting industry. Ex.1003, ¶ 113. Additionally, because the TPM specification was an industry standard, it had gone through extensive security testing and vetting by security experts, and was particularly suited for implementations, such as those in Gleichauf, where security was an important consideration. *Id.* Accordingly, the implementation of the TPM in Gleichauf’s device would have ensured (or greatly increased) the likelihood of having a stable and secure platform without security backdoors or other vulnerabilities. *Id.*

On the network side, a policy server receiving Gleichauf’s posture credentials would process them more quickly and with more confidence that adequate information is provided. Ex.1003, ¶ 114. Indeed, the policy server would be able to process all received posture credentials in substantially the same way because such credentials would have been gathered in substantially the same way. *Id.*

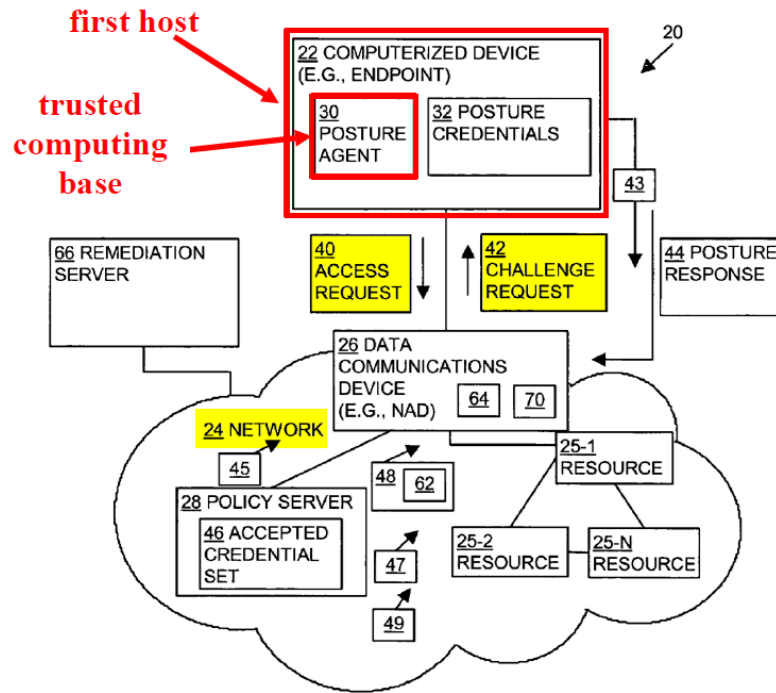
In the combination, Gleichauf’s posture agent would be augmented to implement the trusted platform module functionality, i.e., a *trusted platform module*, per Ovadia. *Id.* For example, one obvious way to implement Gleichauf’s posture agent with trusted platform module functionality would be for the posture agent software to be executed by the execution engine in Ovadia’s trusted platform

module. Ex.1003, ¶ 115; Ex.1006, Fig. 2. The posture agent implemented with trusted platform module functionality would interface with Gleichauf's posture plug-ins in the same way disclosed in Gleichauf by using the "posture plug-in API [application programming interface]" to collect information requested by

5 Gleichauf's policy server. Ex.1005, 9:24-28; Ex.1003, ¶ 115. 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. Ex.1006, 5:36; Ex.1003, ¶ 115. As another example implementation of Gleichauf's
10 posture agent with trusted platform module functionality, the posture agent software would simply be integrated with a software-implemented trusted platform module. *Id.*; see Ex.1006, 4:24-26 ("TPM functionality may be embodied... via software."). With the posture agent (part of the *trusted computing base*) executing on the TPM hardware or integrated with the TPM software, the functionality of the
15 posture agent and posture plug-ins would be *associated with a trusted platform module*. Ex.1003, ¶ 115.

Third, Gleichauf's policy server *contacts* the computerized device's posture agent, which prompts the posture agent to collect the device's posture credentials from the posture plug-ins. Ex.1005, 4:7-11, 3:28-36; Ex.1003, ¶ 109. Gleichauf's
20 Figure 1, below, shows the posture agent *within* the device. *Id.*, ¶ 110. It would

have been obvious to a POSITA that the *trusted platform module* would likewise be *within* the device. Ex.1003, ¶ 111.

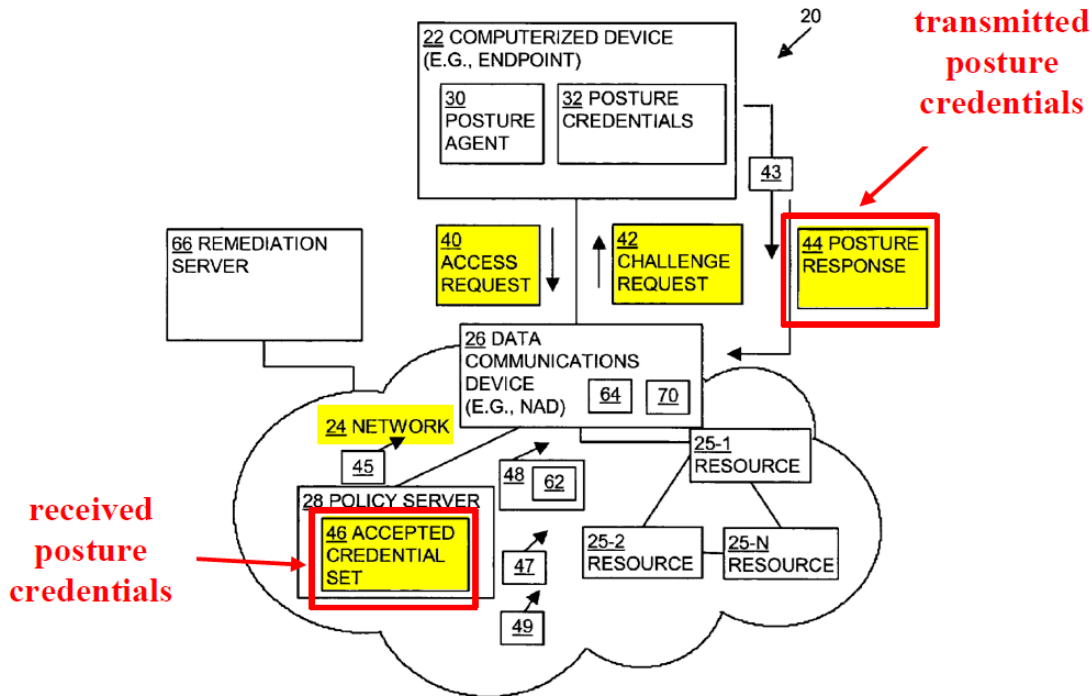


Ex.1005, Fig. 1 (annotated); Ex.1003, ¶ 110.

5 [1.3] *receiving a response, and*

Gleichauf's posture agent responds to the policy server's request by collecting the computerized device's posture credentials from the one or more posture plug-ins and transmitting the credentials in a posture response to the policy server. Ex.1005, 3:28-36, 4:11-15; Ex.1003, ¶¶ 117-118. Gleichauf's Figure 1

10 illustrates the transmission and reception of posture credentials:



Ex.1005, Fig. 1 (annotated); Ex.1003, ¶ 119.

Thus, the policy server “receiv[es] a response” to its request for the posture credentials. Ex.1003, ¶¶ 119-120.

5 **[1.4] determining whether the response includes a valid digitally signed attestation of cleanliness,**

First, as discussed at [1.3], Gleichauf’s posture response includes posture credentials. Ex.1003, ¶¶ 121-122. The posture credentials received by the policy server contain information such as “what particular virus software and virus
10 definition versions are installed,” “when was the last time a local antivirus scan was executed,” “what operating system patches are installed,” and “types and

versions of software applications” on the device, each individually and collectively providing an *attestation of cleanliness*.⁴ Ex.1005, 3:36-49; Ex.1003, ¶ 122.

Second, the policy server analyzes and “validate[s] the posture credentials” (i.e., “*determining whether the response includes a valid ... attestation of cleanliness*”). Ex.1005, 10:30-33; Ex.1003, ¶ 123. For example, the policy server checks for (i.e., “*determin[es]*”) the presence of an anti-virus application. Ex.1005, 10:30-33; Ex.1003, ¶ 123. If a current anti-virus is present, the computerized device will be allowed to access the network resources, which demonstrates that the policy server deemed the response to include “*a valid ... attestation of cleanliness*.” Ex.1005, 11:45-58; Ex.1003, ¶¶ 124-125. If a current anti-virus is absent, the device’s access to the network will be restricted, which demonstrates that the policy server deemed the response to lack “*a valid ... attestation of*

⁴ This is consistent with examples of attestation of cleanliness described in the ’705 patent point: “In some embodiments, the query for cleanliness (1302) may be responded to by anti-contagion software, such as antivirus software, with *assertions about the currency of a scan, such as the last time a scan was performed, or a version associated with a current anti-contagion Software or definition file in use, wherein a sufficiently updated software and/or scan* may act as a cleanliness assertion.” Ex.1001, 14:15-19.

cleanliness.” Ex.1005, 11:58-67; Ex.1003, ¶ 125.

Third, the posture agent “securely transmit[s]” posture credentials to the policy server. Ex.1005, 11:3-12; Ex.1003, ¶ 126.

Fourth, Ovadia provides further details about secure transmission of a
5 device’s configuration information for the purpose of requesting access to a network. Ex.1003, ¶ 127. Like the device in Gleichauf, Ovadia’s subscriber station seeking network access uses its trusted platform module to respond to an authentication request by taking a “current measurement of the firmware and/or software configuration” and uses this as a “current integrity measurement.”
10 Ex.1006, 14:1-6; Ex.1003, ¶ 127. The integrity measurement is stored in the trusted platform module. Ex.1006, 13:61-63, Fig. 8; Ex.1003, ¶ 127. The integrity measurement is then used to produce a value that is “**digitally signed** with the SS’s [subscriber station’s] AIK [Attestation Integrity Key] private key, and sent to the AS [Authentication Server].” Ex.1006, 14:6-10; Ex.1003, ¶ 127. The subscriber
15 station “is authenticated” and allowed access to the network if the integrity measurement is acceptable and “the digital signature is authentic.” Ex.1006, 14:13–23, 16:66-17:6; Ex.1003, ¶ 127.

As described in more detail in Section IX.C.4., a POSITA would have been motivated to combine the teachings of Gleichauf and Ovadia and would have had
20 the skills to do so. Ex.1003, ¶ 128. A POSITA would have understood that

Ovadia's digital signature decreases the risk of accepting credentials that have been altered by a malicious actor. Ex.1003, ¶ 128. Additionally, at the time of the alleged invention in 2004, digital signatures were widely known and used as a means for security authentication and validation, dating back to the 1980s.

5 Ex.1003, ¶¶ 71, 128. As discussed at [1.2], in the combination, Gleichauf's posture credentials would be prepared by a posture agent implemented with trusted platform module functionality (as taught by Ovadia). Ex.1003, ¶ 128.

A POSITA would have recognized that digitally signing Gleichauf's posture credentials (as taught by Ovadia) would provide a mechanism for Gleichauf's
10 policy server (which evaluates those credentials) to be assured that the posture credentials have not been forged or tampered with. *Id.* Ovadia explains that a device's integrity measurements (*e.g.*, posture credentials) are digitally signed with an Attestation Integrity Key (AIK), and that "AIKs are only permitted to sign data generated by a TPM." Ex.1006, 14:9, 5:35-36. Since only the trusted platform
15 module (Gleichauf's posture agent implemented with trusted platform module functionality as discussed at [1.2]) could produce a valid digital signature and the trusted platform module would only sign posture credentials that it prepares itself, Gleichauf's policy server would have an assurance that posture credentials with a valid digital signature were authentic and correct. Ex.1003, ¶ 128. Put differently,
20 the policy server would be assured that a virus or other malware on the computing

device had not tampered with or forged the posture credentials to hide their presence on the computing device. *Id.* It would have been obvious to a POSITA to apply Ovadia's teaching of digitally signing an integrity measurement of a device's configuration information to Gleichauf's transmission of a device's posture credentials. *Id.* In the combination, Gleichauf's posture credentials, prepared by the posture agent implemented with trusted platform module functionality, would be digitally signed, as taught by Ovadia. Ex.1003, ¶¶ 128-130.

[1.5] *wherein the valid digitally signed attestation of cleanliness includes at least one of an attestation that the trusted computing base has ascertained that the first host is not infested, and an attestation that the trusted computing base has ascertained the presence of a patch or a patch level associated with a software component on the first host;*

First, the computerized device's posture credentials transmitted to the policy server (the "*attestation of cleanliness*" in [1.4]), contain information about "the last time a local antivirus scan was executed." Ex.1005, 3:41; Ex.1003, ¶¶ 131-132.

Gleichauf expressly contemplates that failure to show "timely execution of the anti-virus application" is a failure to show cleanliness and results in restrictions on network access. Ex.1005, 22:38-47; Ex.1003, ¶ 132. It would have been obvious to a POSITA that the opposite showing would have the opposite effect. Ex.1003, ¶

132. It would have been obvious for a posture credential that does show timely execution of the anti-virus to be treated as "*an attestation that the trusted computing base has ascertained that the first host is not infested.*" *Id.*

Second, Gleichauf's posture credentials also contain information about "what particular virus software and virus definition versions are installed," "what operating system patches are installed," and "types and versions of software applications" on the device. Ex.1005, 3:36-49; Ex.1003, ¶ 133. Such information
5 on software and patches is "*an attestation that the trusted computing base has ascertained the presence of a patch or a patch level associated with a software component on the first host.*" *Id.*, ¶ 134.

Gleichauf's posture agent prepares posture credentials from information it collects from posture plug-ins. Ex.1005, 3:9-16, 9:24-39; Ex.1003, ¶ 135. The
10 posture agent "is also responsible for determining whether the posture of the computer device 22 has changed by learning of any changes from the posture plug-ins 32 through the posture plug-in API." Ex.1005, 9:50-53. The combined functionality of the posture agent and plug-ins, i.e., *trusted computing base*, thus "*ascertains*" the security posture of the device. Ex.1003, ¶ 135.

15 **Third**, Gleichauf's disclosures regarding the claimed *attestation* are substantially the same as those of the '705 patent specification. Ex.1003, ¶ 136. The '705 patent states. Ex.1001, 14:12-19; *see also* FN4, *supra*.

A POSITA would have understood Gleichauf's disclosure that posture credentials contain information about "the last time a local antivirus scan was
20 executed," Ex.1005, 3:36-49, to equate to the '705 patent's note about the currency

of a virus scan. Ex.1003, ¶ 137.

[1.6] *when it is determined that the response does not include a valid digitally signed attestation of cleanliness, quarantining the first host, including by preventing the first host from sending data to one or more other hosts associated with the protected network, wherein preventing the first host from sending data to one or more other hosts associated with the protected network includes*

First, as explained in [1.4], Gleichauf's policy server reviews and validates posture credentials. Ex.1005, 10:30-33, 11:45-58; Ex.1003, ¶¶ 139-140. The policy server reviews the posture credentials for compliance with network admission policy, e.g., for the presence of an anti-virus application and/or the timeliness of the last virus scan. Ex.1005, 4:21-27, 11:45-58, 22:38-47, 23:56-65; *see also id.*, 3:28-45 (describing examples of posture credentials); Ex.1003, ¶ 140.

Second, Gleichauf describes using the posture credentials to determine whether to place the device in quarantine ("*quarantining the first host*"). Ex.1005, 4:24-27, 4:50-53, 7:3-8; 21:5-8, Ex.1003, ¶ 141. There are many reasons why a device might be placed into quarantine, including if it is found to be out of compliance with a network admission policy, Ex.1005, 4:50-56, if its behavior is found to be anomalous, suggesting that it may have been compromised, *see* Ex.1017, [0039], or if it is found to be infected with a virus or has been hijacked by an unauthorized entity. *See, e.g.*, Ex.1016, 4:8-14; Ex.1003, ¶ 142.

If the policy server "fails to detect the presence of, or a listing of, the anti-virus" application or definition in the device's posture credentials, the device

is out of compliance with network admission policy (i.e., “*when it is determined that the response does not include a valid digitally signed attestation of cleanliness*”). Ex.1005, 11:59-60, 22:38-47; Ex.1003, ¶ 143. Gleichauf provides

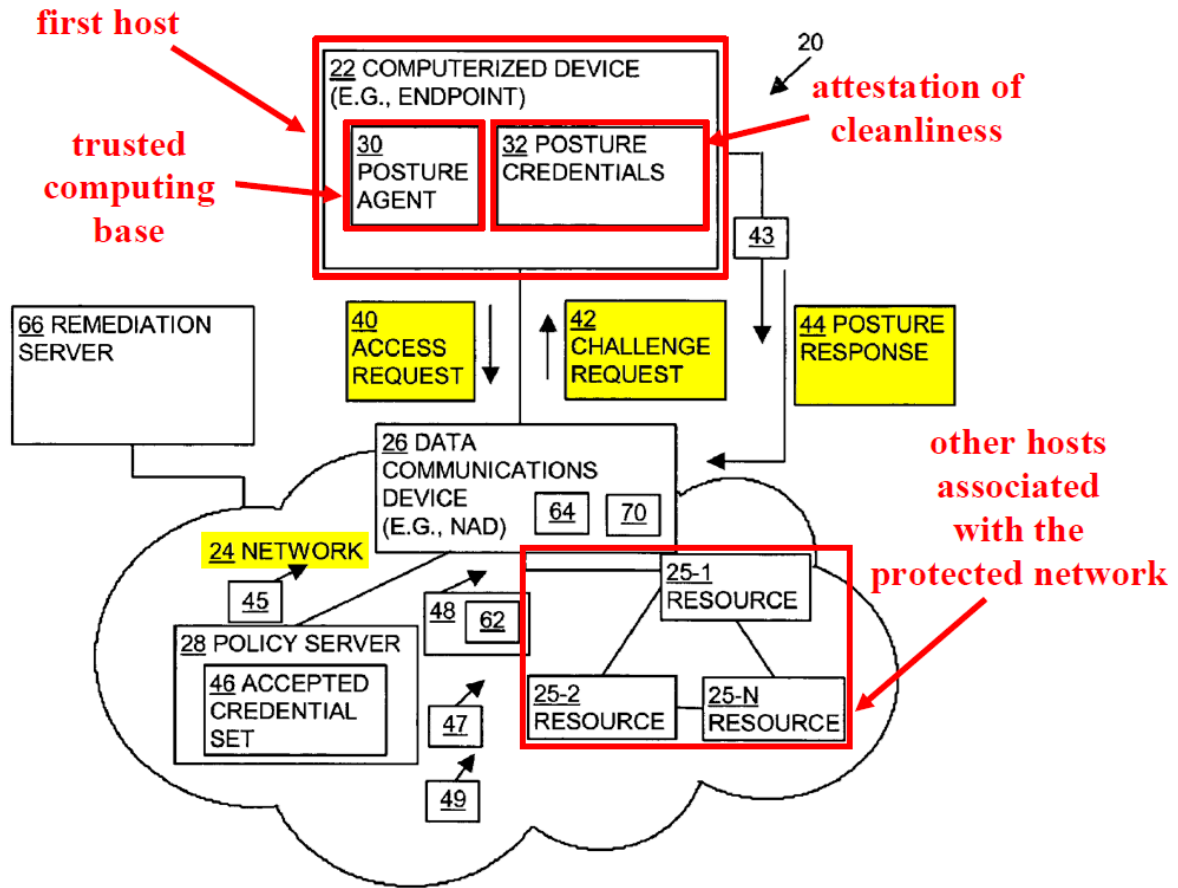
that “devices that are out of compliance with admission requirements [are

5 assigned] to a quarantine.” Ex.1005, 4:50-56. Placement in quarantine causes the network (and more specifically, data communication device 26 acting on directions from the policy server) to “restrict or completely reject communications from the user device 22 to the resources 25 within the network 24.” Ex.1005, 11:58-67,

22:38-47; Ex.1003, ¶ 143. Gleichauf describes that the resources 25 can be, for

10 example, “**servers** that contain content accessible by the user device 22,” Ex.1005, 9:65-67, which is consistent with the ’705 patent’s characterization that a *host* can be a server. Ex.1001, 12:7-11: “In some embodiments, **requests to connection to**

a host other than the remediation server 1032 are redirected to...”; see also, *id.*, 20:48-50, “**remediation host**.” Ex.1003, ¶ 144.



Ex.1005, Fig. 1 (annotated); Ex.1003, ¶ 145.

Restricting or completely rejecting communications from the computerized device to network resources, e.g., resources 25-1 through 25-N, renders obvious preventing the first host from sending data to one or more other hosts associated with the protected network. Ex.1003, ¶¶ 145-146.

[1.7] receiving a service request sent by the first host,

Gleichauf describes how the computerized device transmits an “access request” or “signaling request” in an “attempt to access the network resources

within the network.” Ex.1005, 3:50-56, 8:1-4; Ex.1003, ¶¶ 147-148. Such access

attempts can occur even after the device has been placed in quarantine. For example, Gleichauf describes that to ensure that a device attempting to access the network is the same device as a previously authorized one, the posture query procedure of Figure 1 is repeated. Ex.1005, 16:46-55; Ex.1003, ¶¶ 149-150. As
5 another example, Gleichauf explains that for a device that has been directed to a remediation server but has failed to be remediated, “the device *will be again identified as being at some level of non-compliance* and the appropriate response will be invoked by the policy server.” Ex.1005, 5:13-16; Ex.1003, ¶ 150.

A request to access network resources is a “*service request*.” Ex.1003, ¶ 151.

10 Gleichauf describes in some arrangements, the resources 25 can be “servers that contain content accessible by the user device.” Ex.1005, 9:65-67. Thus, a POSITA would have found it obvious that a request to access such a server to obtain content would be a request for a *service*. Ex.1003, ¶ 151. The request is received and acted upon by a data communications device. Ex.1005, 3:60-65; Ex.1003, ¶ 151.

15 Therefore, Gleichauf discloses “*receiving a service request sent by the first host*.” Ex.1003, ¶ 152.

[1.8] *serving a quarantine notification page to the first host when the service request comprises a web server request,*

First, as explained in [1.6], Gleichauf discloses that a computerized device
20 that is out of compliance with network admission policy is quarantined. Ex.1005, 4:50-56; Ex.1003, ¶¶ 153-154. Communications from a quarantined device are

restricted or rejected. Ex.1005, 11:58-67; Ex.1003, ¶ 154. Limited communications to certain destinations, such as a remediation server, may be permitted. Ex.1005, 14:62-66; Ex.1003, ¶ 154. Notably, Gleichauf explains that in such circumstances:

...the data communications device 26 **limits the user device's 22**
5 **access to only a portion of the network resources 25** (i.e., for
example, of the network resources 25-1, 25-2, and 25-N, the data
communications device 26 allows the user device 22 to
communicate only with resources 25-1, and 25-2). **The access**
instruction 48 may permit traffic from user device 22 only to
10 **certain destination addresses, e.g. IP address of remediation**
server, and only to certain protocols or ports, e.g. HTTP traffic.
Therefore, as the data communications device 26 applies the access
instruction 48 to communications traffic originating from the
computerized device 22, **the data communications device 26**
15 **limits or restricts user device access to a subset of the network**
resources 25 based upon the results of the examination.

Ex.1005, 14:55-15:4.

For example, Gleichauf provides that “a redirect access instruction” is
applied to “HTTP traffic.” Ex.1005, 15:55-67; Ex.1003, ¶ 155. A POSITA would
20 have found it obvious that HTTP traffic includes a *web server request* because
hyper-text transfer protocol (HTTP) is the protocol used for communications
between web browsers and web servers. Ex.1003, ¶ 155. This is evidenced by the
fact that Gleichauf refers to the redirection as a “URL Redirect.” Ex.1005, 15:45,

Ex.1003, ¶ 156. A POSITA would have been familiar with URL redirection, which was a common technique used to forcibly provide an alternative response to a client's request for a webpage. Ex.1015, [0003]; Ex.1003, ¶ 156. This technique is commonly used to redirect a user's web browser to a login page at a hotel or airport. Ex.1008, 4:56-57; Ex.1009, [0044]; Ex.1003, ¶ 156.

Second, Gleichauf describes providing "one or more notification messages" for display to a user "indicating that the device has been quarantined." Ex.1005, 4:56-63, 21:1-12. The message may include "a link to a remediation server or the IP address of the remediation server." Ex.1005, 21:1-12; Ex.1003, ¶ 157.

While strictly not needed to satisfy [1.8], in an alternate mapping, this claim limitation is rendered obvious based on the combination of Gleichauf, Ovadia and Lewis. Notably, Lewis also discloses the use of redirection with a client device that is quarantined. Ex.1007, 4:24-31; Ex.1003, ¶ 159. Therefore, in the Gleichauf-Ovadia-Lewis combination, the functionality of providing a *quarantine notification* can be additionally or alternatively provided by Lewis's quarantine server.

Ex.1003, ¶ 158. Providing the quarantine notification by the quarantine server would have been beneficial for the same reasons as those described above in Section IX.C.5. In the combination, when a user of the device opens a web browser (which would generate a "*service request compris[ing] a web server request*"), an initial domain name service (DNS) request from the device is redirected to a

quarantine server. Ex.1007, 13:7-12; Ex.1003, ¶ 159. The quarantine server receives the DNS request and supplies a default webpage on the quarantine server to the device (“*serving a quarantine notification page to the first host*”). Ex.1007, 13:12-15; Ex.1003, ¶ 159. The webpage informs the user that the device “is in quarantine and corrective action must be taken.” Ex.1007, 10:61-66; Ex.1003, ¶ 159.

As described in more detail in Section IX.C.5, a POSITA would have been motivated to combine the teachings of Gleichauf, Ovadia, and Lewis and would have had the skill to do so. Ex.1003, ¶ 160. A POSITA would have appreciated the security benefits of redirecting HTTP traffic from Gleichauf’s device to a quarantine server (per Lewis), rather than a remediation server. Ex.1003, ¶ 160. For example, as explained in Section IX.C.5, the combination would have allowed the quarantine server (part of policy server) to re-assess the electronic device’s posture credentials before re-directing the request to the appropriate remediation server. Further, the user of the quarantined device would receive information about the quarantine reasons via a webpage served by the quarantine server. Ex.1003, ¶ 160.

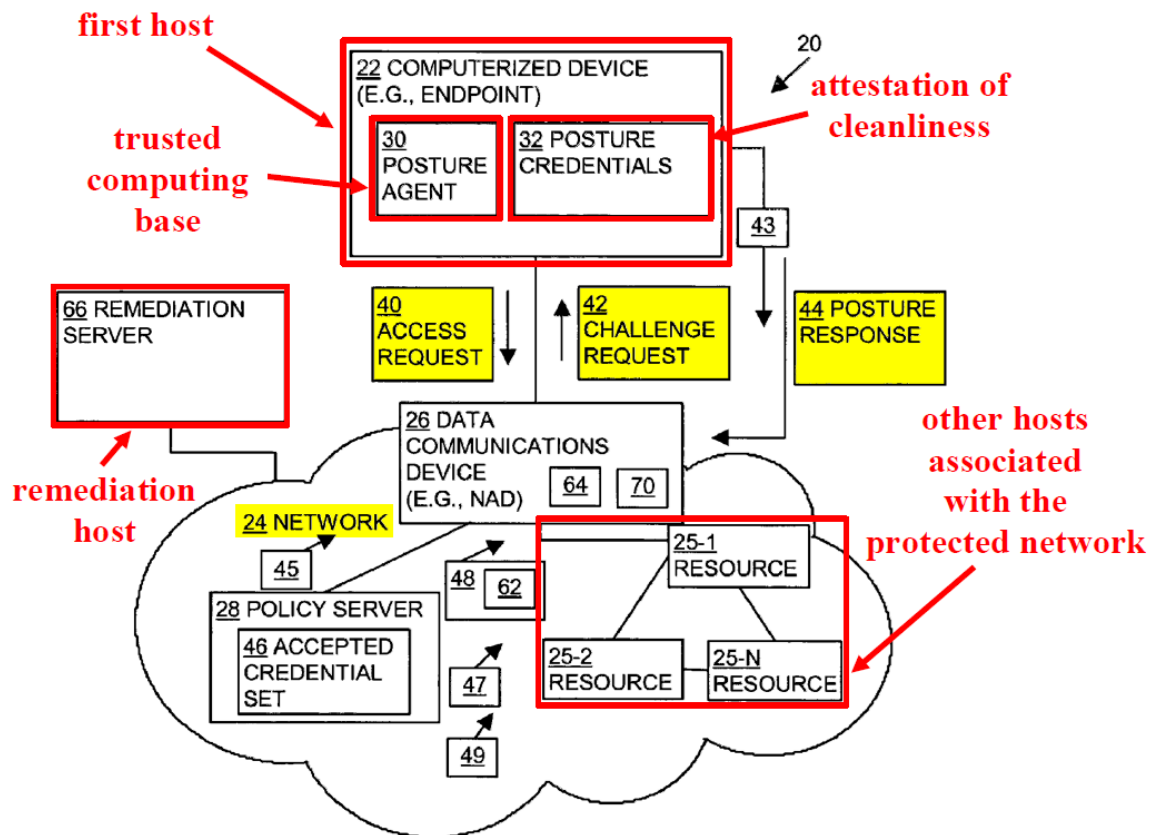
In the combination with Lewis, HTTP traffic from a quarantined device would be redirected to a quarantine server as taught by Lewis. Ex.1003, ¶ 160. The quarantine server (part of policy server) would then serve a webpage to the user, as

taught by Lewis. Ex.1003, ¶ 160. The webpage would inform the user that the device is in quarantine (a “quarantine notification page”). Ex.1003, ¶¶ 160-161.

[1.9] *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 configured to provide data usable to remedy the insecure condition; and*

First, as discussed in [1.6], Gleichauf discloses that a computerized device that is out of compliance with network admission policy is quarantined and that communications from a quarantined device to protected resources in the network are restricted or rejected. Ex.1005, 4:50-56, 11:58-67; Ex.1003, ¶¶ 162-163.

Second, Gleichauf discloses a remediation server, shown in Fig. 1:



Ex.1005, Fig. 1 (annotated); Ex.1003, ¶ 164.

The remediation server “is configured to upgrade or provide up-to-date patches to the operating system or applications, such as anti-virus applications, associated with the computerized device” (a “*remediation host configured to provide data*
5 *usable to remedy the insecure condition*”). Ex.1005, 5:4-11; Ex.1003, ¶ 164. HTTP traffic from the device directed to the remediation server is permitted while HTTP traffic directed to other destinations is redirected. Ex.1005, 14:62-66, 15:5567; Ex.1003, ¶¶ 166-167.

It would have been obvious to a POSITA that HTTP traffic, such as that
10 from Gleichauf’s device, is usually preceded by a DNS query (i.e., “*and in the event the service request comprises a DNS query*”). Ex.1003, ¶ 168. The DNS query is used to obtain the internet protocol (IP) address of a named server, which would be a web server in the case of HTTP traffic. *Id.* The device accesses a webpage by sending an HTTP request to the DNS-resolved IP address. *Id.* These
15 steps were commonly known and ubiquitously employed by web browsers in the prior art. Ex.1008, 5:49-63; Ex.1003, ¶ 168.

Third, Lewis describes a “hijack DNS component 421 for intercepting DNS queries from quarantined clients.” Ex.1007, 11:15-17. The operation of the hijack DNS component is such that “[w]henever the client performs name resolution on
20 any address, it will receive the IP [address] of QS [quarantine server].” Ex.1007,

14:22-23. Thus, the IP address of Lewis's quarantine server (part of the policy server in the combination) is *an IP address of a quarantine server configured to serve the quarantine notification*. Ex.1003, ¶ 169.

Fourth, both Gleichauf and Lewis describe allowing a device placed in quarantine to communicate with one or more servers to fix the issues that led to it being quarantined. Ex.1005, 5:4-13; Ex.1007, 13:12-17, 14:17-31; Ex.1003, ¶ 170. It would have been obvious to a POSITA that communications with such a remediation server would be preceded by a DNS request to resolve the remediation server's name into its IP address. Ex.1003, ¶ 170. This is because servers are frequently known and referred to by their domain name; however, the domain name cannot directly be used to address an IP packet to the server. *Id.* The domain name must first be resolved, using DNS, into the server's IP address. *Id.* The IP address is then specified as the destination for the IP packet. *Id.* A POSITA would have been familiar with the use and operation of DNS. *Id.*

Gleichauf describes allowing a device in quarantine to communicate with a remediation server "to upgrade or provide up-to-date patches to applications, such as anti-virus applications." Ex.1005, 16:14-16. Hence, it would have been obvious to a POSITA for a DNS server (responding to a request to resolve the IP address of the remediation server) to provide the correct IP address of the remediation server.

Ex.1003, ¶ 170. It would have been obvious to a POSITA to adapt the functionality

of a hijack DNS component, e.g., that of Lewis, to provide the correct IP address resolution to DNS queries relating to servers involved with device remediation. *Id.*

For all other queries, the hijack DNS component would function as described in Lewis by answering such DNS queries with the IP address of a quarantine server.

5 *Id.* Thus, the combined teachings of Gleichauf and Lewis render obvious responding to a DNS query with the IP address of a quarantine server *if a host name that is the subject of the DNS query is not associated with a remediation host configured to provide data usable to remedy the insecure condition.* Ex.1003, ¶ 172.

10 As described in more detail in Section IX.C.5, a POSITA would have been motivated to combine the teachings of Gleichauf, Ovadia, and Lewis and would have had the skills to do so. Ex.1003, ¶ 171.

[1.10] *permitting the first host to communicate with the remediation host.*

As explained in [1.6], Gleichauf discloses that a computerized device out of
15 compliance with network admission policy is quarantined and communications from the device to network resources are restricted or rejected. Ex.1005, 4:50-56, 11:58-67; Ex.1003, ¶¶ 173-174. HTTP traffic from the device to a remediation server is permitted, however (i.e., “*permitting the first host to communicate with the remediation host*”). Ex.1005, 14:62-66; Ex.1003, ¶¶ 173-176.

20 **7. Claim 2**

[2.0] A method as recited in claim 1,

See analysis at [1.0]-[1.10].

[2.1] wherein detecting an insecure condition further includes at least one of the following: scanning for a vulnerability, scanning for malicious data, determining whether a security software is installed, and detecting anomalous network traffic.

First, as explained in [1.1], Gleichauf teaches that a policy server *detects an insecure condition* in the computerized device by reviewing the device's posture credentials. Ex.1005, 3:36-45, 11:29-32, 11:58-12:3, 14:16-34, 22:39-59; Ex.1003, ¶¶ 178-179.

Second, the posture credentials include information about "the last time a local antivirus scan was executed," and a device that is "not configured with anti-virus or up-to-date anti-virus applications" is "vulnerable." Ex.1005, 3:36-49, 22:54-56; Ex.1003, ¶ 180. The failure to show "timely execution of the anti-virus application" is another indication that the device is "vulnerable" and in an *insecure condition*. Ex.1005, 22:38-59; Ex.1003, ¶ 180. A POSITA would have understood that checking for an anti-virus application and the recency of an anti-virus scan each constitute "*scanning for a vulnerability*." *Id.*

The posture credentials also contain information about whether the device can "resist reception or transmission of... content (spam and/or data theft)."

Ex.1005, 8:52-56; Ex.1003, ¶ 181. A POSITA would have understood that spam is a type of "*malicious data*" because it is unsolicited. Ex.1003, ¶ 181. A POSITA

would also have understood that the device would be “*scanned*” to determine whether the device can resist reception or transmission of spam. Ex.1003, ¶ 181.

The posture credentials further include information about the presence or absence of an anti-virus software, recentness of an anti-virus definition, and the status of a firewall application. Ex.1005, 3:36-49, 8:62-65; Ex.1003, ¶ 182. A POSITA would have understood that an anti-virus and intrusion prevent software (e.g., a firewall) are types of “*security software*” because they can detect and prevent virus and other security infestations. Ex.1003, ¶ 182. Reviewing the posture credentials for information about the presence or absence of anti-virus software or firewall application involves “*determining whether a security software is installed.*” Ex.1003, ¶¶ 182.

Furthermore, Gleichauf’s “invention helps 'dampen' the propagation of worms and viruses, and related side effects, by **preventing systems capable of participating in such an outbreak from gaining access to the network.**” Ex.1005, 8:28-31. Gleichauf further explains that “security posture of the user device 22 relates, for example, to **the user device's 22 ability to resist reception or transmission of malware (e.g., viruses)**, content (spam and/or data theft), or access by unauthorized users.” Ex.1005, 19:64-67. It would have been obvious that preventing an outbreak of worms and viruses and determining a user device’s ability to resist the reception or transmission of malware involved *detecting*

anomalous network traffic, such as monitoring the reception/transmission of malware by/to the user device. Ex.1003, ¶¶ 183-184.

8. Claim 3

[3.0] *A method as recited in claim 1,*

5 See analysis at [1.0]-[1.10].

[3.1] *wherein detecting an insecure condition includes determining that the first host should be quarantined until an update to an operating system has been installed.*

First, as explained in [1.1], Gleichauf teaches that a policy server *detects an*
 10 *insecure condition* in the computerized device by reviewing posture credentials. Ex.1005, 3:36-45, 11:29-32, 11:58-12:3, 14:16-34, 22:39-59; Ex.1003, ¶¶ 186-187.

Second, the posture credentials contain information about “operating system patches [that] are installed within the computing system” (i.e., “*updates to an operating system*”). Ex.1005, 3:36-49; Ex.1003, ¶ 188. Gleichauf explains, and a
 15 POSITA would have known, that an outdated operating system may make the device “vulnerable” to network threats. Ex.1005, 1:65-2:3, 4:50-56; Ex.1003, ¶¶ 188-189. When the device is quarantined, the policy server can transmit a notification message to the device that may “include a link to a remediation server” where the remediation server “is configured to upgrade or provide up-to-date
 20 patches to the operating system.” Ex.1005, 21:8-12, 5:8-11; Ex.1003, ¶ 190-191.

“The device should be admitted into the original network if the remediation process was successful.” Ex.1005, 5:11-13; Ex.1003, ¶¶ 188-189. A POSITA

would have found it obvious from Gleichauf's description of quarantining devices with outdated operating systems and of providing operating system updates, to quarantine a device "*until an update to an operating system has been installed.*" Ex.1003, ¶¶ 191-192.

5 **9. Claim 5**

[5.0] *A method as recited in claim 1,*

See analysis at [1.0]-[1.10].

[5.1] *wherein permitting the first host to communicate with the remediation host includes:*

10 See analysis at [1.10]. Ex.1003, ¶ 194.

[5.2] *detecting an outbound communication from the first host; and*

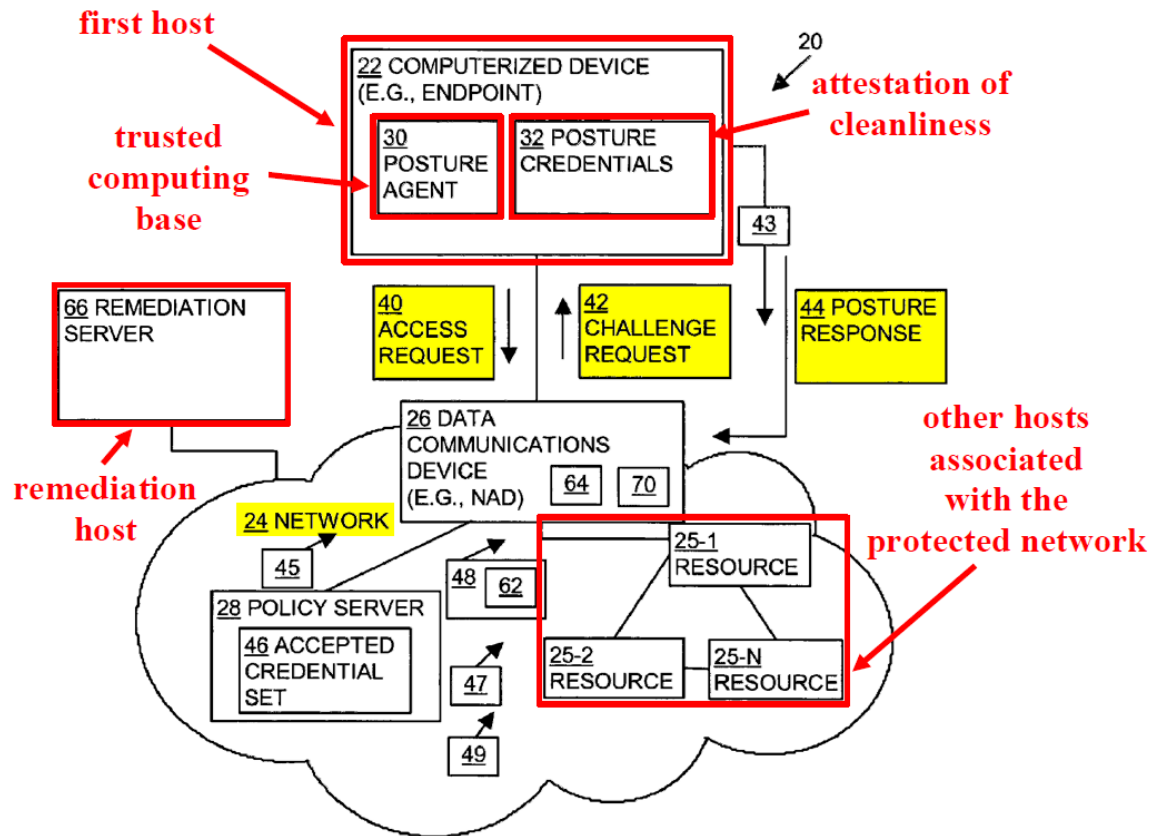
Gleichauf's quarantined device receives a notification message from the policy server "indicating that the device has been quarantined and needs to be remediated" and including "a link to a remediation server." Ex.1005, 21:5-12;

15 Ex.1003, ¶¶ 195-196. The device responds to the message by activating the link and being directed to the remediation server. Ex.1005, 21:20-23; Ex.1003, ¶ 196.

A POSITA would have understood that the device activating the link directing it to the remediation server includes an "*outbound communication from the first host*" to the remediation server. Ex.1003, ¶ 197. Gleichauf explains that
20 data communications device 26 "is configured to **detect attempted connections** to the network 24 by the user device." Ex.1005, 10:6-11, 12:57-67. This is further

illustrated in Figure 2 of Gleichauf, illustrating that outbound access requests from the electronic device 22 are received at the data communications device 26.

Ex.1003, ¶ 197.



5 A POSITA would have understood that directing the device to the remediation server—a server that is not part of the device—would be an “*outbound communication from the first host.*” Ex.1003, ¶ 197. A POSITA would also have understood that the data communications device 26 would “*detect[]*” an outbound communication when it is applying access instructions that govern the device’s

10 permitted level of network access. *Id.*, ¶¶ 198-199; Ex.1005, 13:52-65, 14:3-7, 14:20-34, 22:38-47.

[5.3] *forwarding the outbound communication if it is addressed to the remediation host.*

As discussed in [5.2], Gleichauf discloses that communications from a quarantined device are restricted or rejected. Ex.1005, 22:38-47; Ex.1003, ¶¶ 200-

5 201. A quarantined device is generally permitted to engage in HTTP communication with a remediation server. Ex.1005, 14:62-66. A POSITA would have understood that permitting HTTP traffic from a device to a remediation server, per Gleichauf, would entail the data communications device 26

“*forwarding the outbound communication if it is addressed to the remediation*
10 *host.*” Ex.1003, ¶¶ 201-202.

10. Claim 6

[6.0] *A method as recited in claim 1,*

See analysis at [1.0]-[1.10].

[6.1] *wherein preventing the first host from sending data to the one or more other hosts includes:*

See analysis at [1.6].

[6.2] *detecting an outbound communication from the first host; and*

See analysis at [5.2].

[6.3] *redirecting the outbound communication to a quarantine server if it comprises a request for an approved service and is not addressed to a remediation host.*

First, Gleichauf restricts or rejects communications from a quarantined device. Ex.1005, 11:58-67; Ex.1003, ¶¶ 206-209. Gleichauf explains that

communications are permitted “only to certain destination addresses, e.g., IP address of remediation server, **and only to certain protocols or ports**, e.g., HTTP traffic.” Ex.1005, 14:62-66. A POSITA would have understood Gleichauf to teach that a quarantined device may be limited to communications with a remediation server using only HTTP (an “*approved service*”)⁵. Ex.1003, ¶¶ 206-208.

Second, as discussed in [1.9], Gleichauf discloses that HTTP traffic (an “*approved service*”) from the device to a remediation server is permitted. Ex.1005, 14:62-66; Ex.1003, ¶ 209. However, HTTP traffic from the device to other destinations (“*not addressed to a remediation host*”) will be subject to a “**redirect access instruction**” that will “direct the computerized device 22 (e.g., HTTP traffic from the computerized device 22) to a remediation server.” Ex.1005, 15:55-16:6; *see also id.*, 28:34-40: “applying the redirect access instruction at the intercepting device by operation of one or more computer processors, to communications traffic originating from the computerized device and destined for resources within the

⁵ This interpretation is consistent with the ’705 patent’s description that an approved service request can be a request for a web page. Ex.1001, 14:62-65. A POSITA would have understood that communications using HTTP would be a request for a web page because HTTP was ubiquitously used for communications between web browsers and web servers. *See* explanations in [1.8].

network, in order to grant the computerized device with access to the remediation server without including access to the resources in the network.” Ex.1003, ¶ 209.

As discussed at [1.8]-[1.9], it would have been obvious to a POSITA, in view of Lewis, to *redirect* unauthorized access requests to a *quarantine server* that would

5 inform the user of their device’s quarantine status via a quarantine notification page. Ex.1007, 11:15-17, 14:22-23; Ex.1003, ¶¶ 209-210.

11. Claim 7

[7.0] *A method as recited in claim 1,*

See analysis at [1.0]-[1.10].

10 [7.1] *wherein quarantining the first host further includes preventing the first host from receiving via the protected network data not related to remediation of the insecure condition.*

As explained in [1.6], Gleichauf discloses that a computerized device that is out of compliance with network admission policy is quarantined. Ex.1005,

15 4:50-56; Ex.1003, ¶¶ 212-215. Outbound communications from a quarantined device are not permitted to destinations that are not the remediation server.

Ex.1005, 11:58-67, 14:62-66, 28:34-40; Ex.1003, ¶ 214. It would have been

obvious to a POSITA that inbound communications would be similarly subject to access restrictions. Ex.1003, ¶ 214. If outbound requests from the quarantined

20 device are not permitted to non-remediation destinations, the quarantined device is effectively prevented from receiving information that it would have received in

response to such requests. Ex.1003, ¶ 214. Since Gleichauf teaches that a device in

quarantine may only communicate with remediation-related servers, it would have been obvious to a POSITA that Gleichauf's system prevents the quarantined device from receiving information not related to remediation (*"preventing ... from receiving via the protected network data not related to remediation of the insecure condition"*). Ex.1003, ¶ 214.

To the extent that this claim is interpreted as requiring the blocking of all non-remediation-related traffic from reaching the device, this restriction would have been obvious to a POSITA. Ex.1003, ¶ 215. Gleichauf is concerned with limiting or preventing the risk that network resources "receive or transmit malware" **from or to** "vulnerable computerized devices." Ex.1005, Abstract, 8:52-55; Ex.1003, ¶ 215. That is why vulnerable devices are quarantined in an "isolated ... network segment" until the vulnerability has been remedied so that the device cannot receive such malware. Ex.1005, 4:25-26; Ex.1003, ¶ 215. It would thus have been obvious to a POSITA that a device in quarantine would be restricted from receiving data not related to remediation in the same way that the device is restricted from transmitting data to non-remediation servers. Ex.1003, ¶ 215. A POSITA would have had a reasonable expectation of success in restricting communications to or from a quarantined device because Gleichauf describes allowing or preventing communications "between" the device and the network. Ex.1003, ¶¶ 215-216; Ex.1005, 14:3-7, 14:20-25.

12. Claim 8

[8.0] *A method as recited in claim 1,*

See analysis at [1.0]-[1.10].

[8.1] *performed at an Internet service provider.*

5 Gleichauf's computerized device attempts to communicate using the network, where the network is "a computer network, such an enterprise network or the **Internet**, which includes the network resources 25, the data communications device 26, and the policy server 28." Ex.1005, 3:50-56, 9:57-64; Ex.1003, ¶¶ 218-220. Network resources 25 are "servers that contain content accessible by the . . .
10 device." Ex.1005, 9:65-67.

A POSITA would have understood that the combination of the data communications device and policy server is an *Internet service provider* because the combination selectively provides access to the network, which Gleichauf explains is the Internet. Ex.1003, ¶ 220. The policy server determines whether to
15 quarantine the device (by means of instructions to the data communications device) and protect the network. Ex.1005, 4:50-56, 10:30-33, 11:45-67; Ex.1003, ¶ 220. While in quarantine, the device is unable to communicate with network resources. Ex.1005, 11:58-67,14:62-66. A POSITA would have therefore appreciated that the data communications device and policy server determine whether the device can
20 access resources on the network, e.g., on the Internet. Ex.1003, ¶ 220. Thus, the data communications device and the policy server together operate to provide the

device Internet access, which a POSITA would have understood to be the function of an “*Internet service provider*.” Ex.1003, ¶¶ 220-221. Moreover, it would have been obvious for the data communications device and policy server to be owned by a company that provides Internet service because they operate together to provide the computerized device Internet access. *Id.*, ¶¶ 221.

13. Claim 9

[9.0] *A method as recited in claim 1,*

See analysis at [1.0]-[1.10].

[9.1] *wherein the software component on the first host is an operating system.*

See analysis at [3.1]. Ex.1003, ¶ 223.

14. Claim 10

[10.0] *A method as recited in claim 1,*

See analysis at [1.0]-[1.10].

[10.1] *wherein determining that the response does not include a valid digitally signed attestation of cleanliness includes determining that the software component on the first host is not sufficiently updated.*

See analysis at [1.4]. Gleichauf’s computerized device’s posture response includes posture credentials. Ex.1003, ¶ 227. The posture credentials contain information about the “particular **virus software and virus definition versions**” installed on the device. Ex.1005, 3:36-49, 8:56-9:5. If the current anti-virus definition is not present, the device is quarantined and communications from it to the network will be restricted or completely rejected, showing that the response is

deemed not to include “*a valid . . . attestation of cleanliness.*” Ex.1005, 11:58-67; Ex.1003, ¶ 225.

15. Claim 11

[11.0] *A method as recited in claim 1,*

5 See analysis at [1.0]-[1.10].

[11.1] *wherein determining that the software component on the first host is not sufficiently updated includes determining that a patch level associated with the software component on the first host is not sufficiently recent.*

The computerized device’s posture credentials contain information about the
 10 “particular virus software and virus definition versions” installed on the device.
 Ex.1005, 3:36-49, 8:56-9:5; Ex.1003, ¶¶ 228-229. If a review of the posture
 credentials “fails to detect the presence of an anti-virus application (e.g., or **an up-
 to-date anti-virus application or virus definition file**),” the device will be
 quarantined. Ex.1005, 14:16-34; Ex.1003, ¶ 228. A POSITA would have
 15 understood that failure to detect an up-to-date anti-virus application or definition
 file indicates that the existing anti-virus application or definition file on the device
 “*is not sufficiently recent.*” Ex.1003, ¶ 228. A POSITA would have understood that
 updates to anti-virus applications or definition files were often provided in the
 form of patches that updated the out-of-date applications or virus definition files.
 20 *Id.* Accordingly, it would have been obvious that failure to detect the presence of
 an up-to-date anti-virus application or virus definition file would be *determining*
that a patch level associated with the software component is not sufficiently recent.

Ex.1003, ¶¶ 228-229.

16. Claim 12

[12.0] *A system for protecting a network, comprising:*

Gleichauf teaches a “**robust network security architecture and system for**
5 **controlling access to a network** depending on the configuration state or ‘security posture’ of a device when it attempts to connect to that network.” Ex.1005, 3:4-9; Ex.1003, ¶¶ 230-231. Gleichauf’s system, and specifically its policy server, protects the network by reducing the risk of the network becoming infected with malware or suffering a virus outbreak. Ex.1005, 8:16-31; Ex.1003, ¶ 231. Thus,
10 Gleichauf discloses “[a] system for protecting a network.” Ex.1003, ¶ 232. As explained below, the combination of Gleichauf, Ovadia, and Lewis renders obvious a system “*comprising*” the claimed steps.

[12.1] *a processor configured to:*

As discussed in [1.0]-[1.10], Gleichauf’s policy server operates to protect the
15 network. The policy server “includes a controller 209 formed of a memory 210 and a **processor 212**.” Ex.1005, 25:44-46; Ex.1003, ¶¶ 233-235.

[12.2]-[12.11] (elements recited in claim 12)

Elements [12.2]-[12.11] are substantively identical to the steps of claim 1 and are rendered obvious for the same reasons.

20 [12.12] *a memory coupled to the processor and configured to provide instructions to the processor.*

As discussed in [12.1], Gleichauf's policy server operates to protect the network. Gleichauf's policy server "includes a controller 209 formed of a **memory 210** and a processor 212." Ex.1005, 25:41-46; Ex.1003, ¶¶ 246-247. A POSITA would have understood that memory 210 is coupled to processor 212 because they
5 are part of the same controller 209. Ex.1003, ¶ 247.

Memory 210 is "*configured to provide instructions to the processor*" of the policy server because it is "encoded with **logic instructions (e.g., software code)** and/or data that form a credential analysis application 246" where "application 246 represents **software code, instructions and/or data ... that reside within**
10 **memory** ... accessible to the policy server 28." Ex.1005, 26:66-27:8; Ex.1003, ¶¶ 248-249.

17. Claims 13 and 15-18

Claims 13 and 15-18 depend from claim 12 and recite substantively identical limitations as claims 2, 7, and 9-11 that depend from claim 1. Accordingly, claims
15 13 and 15-18 are rendered obvious for the same reasons.

18. Claim 19

[19.0] A computer program product for protecting a network, the computer program product being embodied in a non-transitory computer readable medium and comprising computer instructions for:

20 Gleichauf describes a "**computer program product 200** includes an application or logic instructions that are loaded into the computerized device 22, data communications device 26, and policy server 28 to configure the devices 22,

24, 26 to operate as part of the data communications system 20” described throughout the reference. Ex.1005, 25:46-51, 3:4-9; Ex.1003, ¶¶ 260-261. A POSITA would have understood that Gleichauf’s computer program product *comprises computer instructions*. Ex.1003, ¶ 261.

5 As described for claim 12, Gleichauf’s policy server operates to protect the network and directs the operation of the data communications device 26. The policy server includes “*non-transitory computer readable medium*” in the form of memory 210 that stores “**logic instructions (e.g., software code) and/or data**” for operating the disclosed security architecture. Ex.1005, 26:66-7:8; Ex.1003, ¶ 262.

10 Gleichauf’s system protects the network by reducing the risk of the network becoming infected with malware or suffering a virus outbreak. Ex.1005, 8:16-31; Ex.1003, ¶¶ 262-264.

[19.1]-[19.10] (elements recited in claim 19)

15 Elements [19.1]-[19.10] are substantively identical to the steps of claim 1 and are rendered obvious for the same reasons.

X. CONCLUSION

Petitioner has established a reasonable likelihood that the Challenged Claims are unpatentable.

Dated: August 13, 2025

PERKINS COIE LLP
11452 El Camino Real, Suite 300
San Diego, CA 92130
Telephone: (858) 720-5700
Fax: (858) 720-5834

Respectfully submitted,

/Thomas Millikan/

Lead Counsel
Thomas Millikan, Reg. No. 72,316

Back-up Counsel
Babak Tehrani, Reg. No. 55,937
Andrew Klein, *Pro Hac Vice*
Hessam Gharaviram, *Pro Hac Vice*

Attorneys for Petitioner

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

Pursuant to 37 C.F.R. § 42.24(d), the undersigned hereby certifies that the word count for the foregoing *Petition for Inter Partes Review* of U.S. Patent No. 8,234,705 totals 13,894 words, excluding the parts exempted by 37 C.F.R. § 42.24(a). Accordingly, this Petition is under the word count limit of 14,000 words.

This word count was calculated by using the built-in word-count tool in Microsoft Word 365, the software used to prepare the document.

Dated: August 13, 2025

Respectfully submitted,

PERKINS COIE LLP
11452 El Camino Real, Suite 300
San Diego, CA 92130
Telephone: (858) 720-5700
Fax: (858) 720-5834

/Thomas Millikan/

Lead Counsel
Thomas Millikan, Reg. No. 72,316

Back-up Counsel
Babak Tehranchi, Reg. No. 55,937
Andrew Klein, *Pro Hac Vice*
Hessam Gharaviram, *Pro Hac Vice*

Attorneys for Petitioner

CERTIFICATE OF SERVICE

The undersigned hereby certifies that a true copy of the foregoing PETITION FOR *INTER PARTES* REVIEW OF U.S. PATENT NO. 8,234,705 and supporting materials (Exhibits and Power of Attorney) have been served this 13th day of August 2025, by FedEx Mail delivery service on Patent Owner at the correspondence address for the attorney of record for the '705 patent shown in USPTO Patent Center:

KINNEY & LANGE, P.A.
333 S. 7TH ST., SUITE 2700
MINNEAPOLIS, MN 55402-2438
UNITED STATES

Dated: August 13, 2025

By: /Meghan Bright/
Meghan Bright
Paralegal