

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

**Citrix Systems, Inc.,
Cloud Software Group, Inc**
Petitioners

v.

K.Mizra LLC
Patent Owner

Inter Partes Review No.: IPR2025-01468

**PETITION FOR INTER PARTES REVIEW OF U.S. PATENT NO.
8,234,705 UNDER 35 U.S.C. §§ 311-319 AND 37 C.F.R. §§ 42.1-100, ET SEQ**

TABLE OF CONTENTS

	Page
I. INTRODUCTION	1
II. STATEMENT OF PRECISE RELIEF REQUESTED	1
III. THE '705 PATENT	2
A. Overview of the '705 Patent.....	2
B. Person of Ordinary Skill in the Art	2
C. Claim Construction Under 37 C.F.R. §42.104(b)(3)	3
1. “trusted computing base” (claim 1, 12, 19)	3
2. “trusted platform module” (claims 1, 12, 19)	3
IV. GROUND 1: CLAIMS 1-19 ARE OBVIOUS OVER GLEICHAUF IN VIEW OF OVADIA AND LEWIS.....	4
A. Gleichauf	4
B. Ovadia.....	5
C. Lewis	5
D. Motivation to Combine Gleichauf and Ovadia	6
E. Motivation to Combine Gleichauf, Ovadia, and Lewis	10
F. Claim 1	12
1. [1pre]	12
2. [1a]	12
3. [1b]	14
4. [1c]	20
5. [1d]	21
6. [1e]	24
7. [1f].....	26
8. [1g]	28
9. [1h].....	28
10. [1i].....	30
11. [1j].....	34

G.	Claim 2	35
H.	Claim 3	36
I.	Claim 4	37
J.	Claim 5	38
	1. [5pre]	38
	2. [5a]	38
	3. [5b]	39
K.	Claim 6	39
	1. [6pre]	39
	2. [6a]	39
	3. [6b]	39
L.	Claim 7	40
M.	Claim 8	42
N.	Claim 9	43
O.	Claim 10	43
P.	Claim 11	43
Q.	Claim 12	44
	1. [12pre]	44
	2. [12a]	44
	3. [12b]-[12k]	45
	4. [12l]	45
R.	Claim 13	45
S.	Claim 14	45
T.	Claim 15	45
U.	Claim 16	45
V.	Claim 17	45
W.	Claim 18	45
X.	Claim 19	46
	1. [19pre]	46

2.	[19a]-[19j]	46
V.	GROUND 2: CLAIMS 1-19 ARE OBVIOUS OVER FREUND IN VIEW OF BALL AND DANFORTH.....	46
A.	Freund.....	46
B.	Ball	47
C.	Danforth.....	48
D.	Motivation to Combine	48
E.	Claim 1	52
1.	[1pre]	52
2.	[1a]	52
3.	[1b]	55
4.	[1c]	58
5.	[1d]	58
6.	[1e]	60
7.	[1f]	64
8.	[1g]	65
9.	[1h]	67
10.	[1i]	69
11.	[1j]	75
F.	Claim 2	76
G.	Claim 3	76
H.	Claim 4	77
I.	Claim 5	77
1.	[5pre]	77
2.	[5a]	77
3.	[5b]	79
J.	Claim 6	79
1.	[6pre]	79
2.	[6a]	79

3.	[6b]	79
K.	Claim 7	80
L.	Claim 8	81
M.	Claim 9	81
N.	Claim 10	82
O.	Claim 11	82
P.	Claim 12	83
1.	[12pre]	83
2.	[12a]	83
3.	[12b]-[12k]	84
4.	[12l]	84
Q.	Claim 13	85
R.	Claim 14	85
S.	Claim 15	85
T.	Claim 16	85
U.	Claim 17	85
V.	Claim 18	85
W.	Claim 19	85
1.	[19pre]	85
2.	[19a]-[19j]	86
VI.	MANDATORY NOTICES UNDER 37 C.F.R. §§ 42.8(B)(1)-(4)	86
A.	Real Party-In-Interest	86
B.	Related Matters	86
C.	Lead and Backup Counsel	88
D.	Service Information	89
VII.	CONCLUSION	89

INDEX OF EXHIBITS

Exhibit No.	Description
1001	U.S. Patent No. 8,234,705.
1002	Declaration of Dr. Clay Shields.
1003	File History of U.S. Patent No. 8,234,705.
1004	U.S. Patent No. 9,436,820 (“ Gleichauf ”).
1005	U.S. Patent No. 7,747,862 (“ Ovadia ”).
1006	U.S. Patent No. 7,533,407 (“ Lewis ”).
1007	U.S. Patent Pub. No. 2003/0055962 (“ Freund ”).
1008	U.S. Patent Pub. No. 2006/0005009 (“ Ball ”).
1009	U.S. Patent No. 7,571,460 (“ Danforth ”).
1010	U.S. Patent no. 7,568,107 (“ Rathi ”).
1011	U.S. Patent Pub. No. 2004/0177276 (“ MacKinnon ”).
1012	U.S. Patent no. 7,330,977 (“ Cromer ”).
1013	U.S. Patent No. 6,829,654 (“ Jungck ”).
1014	TCG Specification Architecture Overview (Rev. 1.2 Apr. 28, 2004), available at: https://web.archive.org/web/20061003162035if_/https://www.trustedcomputinggroup.org/groups/TCG_1_0_Architecture_Overview.pdf .
1015	TPM Main – Part 1 Design Principles, Specification Version 1.2 (Rev. 62 Oct. 2, 2003), available at: https://trustedcomputinggroup.org/wp-content/uploads/tpmwg-mainrev62_Part1_Design_Principles.pdf .
1016	U.S. Patent Pub. No. 2003/0061494 (“ Girard ”).
1017	U.S. Patent Pub. No. 2005/0078668 (“ Wittenberg ”).
1018	U.S. Patent Pub. No. 2005/0097199 (“ Woodard ”).
1019	U.S. Patent Pub. No. 2005/0033987 (“ Yan ”).
1020	Microsoft Security Bulletin MS99-008 - Critical Microsoft Learn, https://learn.microsoft.com/en-us/security-updates/securitybulletins/1999/ms99-008
1021	What’s Inside A Router, https://web.archive.org/web/20040112004448/http://www2.ic.uff.br/~michael/kr1999/4-network/4_06-inside.htm (Jan. 12, 2004)
1022	<i>Cisco Systems, Inc. v. K.Mizra LLC</i> , IPR2021-00593, Paper 2 (March 15, 2021).

Exhibit No.	Description
1023	<i>Cisco Systems, Inc. v. K.Mizra LLC</i> , IPR2021-00593, Paper 41 (September 19, 2022).
1024	<i>Cisco Systems, Inc. v. K.Mizra LLC</i> , 2022-2290, 2023-1183 (Fed. Cir. 2024)
1025	<i>Cisco Systems, Inc. v. K.Mizra LLC</i> , IPR2021-00593, Paper 23 (January 7, 2022).
1026	<i>Netskope, Inc. v. K.Mizra LLC</i> , IPR2025-01115, Paper 2 (April 13, 2025).
1027	Tony Piltzecker et al., How to Cheat at Managing: Windows Server Update Services (2005).

CHART OF CLAIMS
[1pre] A method for protecting a network, comprising:
[1a] detecting an insecure condition on a first host that has connected or is attempting to connect to a protected network,
[1b] wherein detecting the insecure condition includes contacting a trusted computing base associated with a trusted platform module within the first host,
[1c] [wherein detecting the insecure condition includes...] receiving a response, and
[1d] [wherein detecting the insecure condition includes...] determining whether the response includes a valid digitally signed attestation of cleanliness,
[1e] 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;
[1f] 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,
[1g] wherein preventing the first host from sending data to one or more other hosts associated with the protected network includes receiving a service request sent by the first host,
[1h] [wherein preventing the first host from sending data to one or more other hosts associated with the protected network includes...] serving a quarantine notification page to the first host when the service request comprises a web server request, and
[1i] 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
[1j] permitting the first host to communicate with the remediation host.
[2] A method as recited in claim 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.
[3] A method as recited in claim 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.

CHART OF CLAIMS
[4] A method as recited in claim 1, wherein detecting an insecure condition includes configuring an operating system to quarantine the first host upon initial startup after installation of the operating system.
[5pre] A method as recited in claim 1, wherein permitting the first host to communicate with the remediation host includes:
[5a] detecting an outbound communication from the first host; and
[5b] forwarding the outbound communication if it is addressed to the remediation host.
[6pre] A method as recited in claim 1, wherein preventing the first host from sending data to the one or more other hosts includes:
[6a] detecting an outbound communication from the first host; and
[6b] 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.
[7] A method as recited in claim 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.
[8] A method as recited in claim 1, performed at an Internet service provider.
[9] A method as recited in claim 1, wherein the software component on the first host is an operating system.
[10] A method as recited in claim 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.
[11] A method as recited in claim 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.
[12pre] A system for protecting a network, comprising:
[12a] a processor configured to:
[12b] detect an insecure condition on a first host that has connected or is attempting to connect to a protected network,
[12c] wherein detecting the insecure condition includes contacting a trusted computing base associated with a trusted platform module within the first host,
[12d] [wherein detecting the insecure condition includes...] receiving a response, and
[12e] [wherein detecting the insecure condition includes...] determining whether the response includes a valid digitally signed attestation of cleanliness,

CHART OF CLAIMS

[12f] 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;

[12g] when it is determined that the response does not include a valid digitally signed attestation of cleanliness, quarantine the first host, including by preventing the first host from sending data to one or more other hosts associated with the protected network,

[12h] wherein preventing the first host from sending data to one or more other hosts associated with the protected network includes receiving a service request sent by the first host,

[12i] [wherein preventing the first host from sending data to one or more other hosts associated with the protected network includes...] serving a quarantine notification page to the first host when the service request comprises a web server request, and

[12j] 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

[12k] permit the first host to communicate with the remediation host; and

[12l] a memory coupled to the processor and configured to provide instructions to the processor.

[13] A system as recited in claim 12, wherein the processor is configured to detect an insecure condition at least in part by performing one or more of the following: scanning for a vulnerability, scanning for malicious data, determining whether a security software is installed, and detecting anomalous network traffic.

[14] A system as recited in claim 12, wherein the processor is configured to detect an insecure condition at least in part by determining that an initial startup after installation of an operating system is being performed.

[15] A system as recited in claim 12, wherein the processor is configured to quarantine the first host at least in part by preventing the first host from receiving via the protected network data not related to remediation of the insecure condition.

[16] A system as recited in claim 12, wherein the software component on the first host is an operating system.

CHART OF CLAIMS	
[17]	A system as recited in claim 12, 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.
[18]	A system as recited in claim 17, 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.
[19pre]	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:
[19a]	detecting an insecure condition on a first host that has connected or is attempting to connect to a protected network,
[19b]	wherein detecting the insecure condition includes contacting a trusted computing base associated with a trusted platform module within the first host,
[19c]	[wherein detecting the insecure condition includes...] receiving a response, and
[19d]	[wherein detecting the insecure condition includes...] determining whether the response includes a valid digitally signed attestation of cleanliness,
[19e]	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;
[19f]	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,
[19g]	wherein preventing the first host from sending data to one or more other hosts associated with the protected network includes receiving a service request sent by the first host,
[19h]	[wherein preventing the first host from sending data to one or more other hosts associated with the protected network includes...] serving a quarantine notification page to the first host when the service request comprises a web server request, and
[19i]	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

CHART OF CLAIMS
associated with a remediation host configured to provide data usable to remedy the insecure condition; and
[19j] permitting the first host to communicate with the remediation host.

I. INTRODUCTION

Citrix Systems, Inc. and Cloud Software Group, Inc. (“Petitioners”) petition for *inter partes* review of claims 1-19 (“Challenged Claims”) of U.S. 8,234,705 (“’705 patent”) (EX1001).

Petitioners certify that the ’705 patent is available for *inter partes* review and that Petitioners are not barred or estopped from requesting an *inter partes* review challenging the patent claims on the grounds identified in this petition.

The ’705 patent is subject to two previously filed IPRs: IPR2021-00593, IPR2025-01115. EX1022, EX1026. In IPR2021-00593, Cisco Systems requested institution using Ground 1 of the present petition, which was granted. EX1022. In the final written decision, the Board determined no claim was unpatentable. EX1023. The Federal Circuit vacated the Board’s decision and remanded to the Board to consider the remaining obviousness issues. EX1024. The Board never did so because the proceedings were voluntarily terminated.

IPR2025-01115, filed 08/13/2025 by Netskope, Inc., requests institution using Ground 1 of the present petition and is pending pre-institution.

II. STATEMENT OF PRECISE RELIEF REQUESTED

Petitioners request cancellation of the Challenged Claims in view of the following:

Ground	Claims	Stat. Basis	Prior Art
1	1-19	35 U.S.C. § 103	Gleichauf in view of Ovadia and Lewis
2	1-19	35 U.S.C. § 103	Freund in view of Ball and Danforth

III. THE '705 PATENT

A. Overview of the '705 Patent

The '705 patent issued from U.S.P.N. 11/237,003, filed 9/27/2005, and claims priority to U.S.P.N. 60/613,909, filed 9/27/2004. The '705 patent's priority date is no earlier than 9/27/2004. EX1002, ¶40.

The '705 patent describes defending computer networks against threats by analyzing the security state of a device that seeks network access. The device is quarantined if its security state is inadequate. EX1001, Abstract. A device in quarantine "is provided only limited access to the protected network," such as to remedy the condition that caused the quarantine to be imposed. EX1001, 3:9-17. EX1002, ¶41.

B. Person of Ordinary Skill in the Art

A person of ordinary skill in the art at the time of the alleged invention (9/27/2004) ("POSITA") would have had a bachelors' degree in electrical engineering, computer science, computer engineering or a comparable field of study, plus approximately two years of professional experience in the field of network security. Additional graduate education could substitute for professional

experience, and significant professional experience could substitute for formal education. EX1002, ¶¶43-45.

C. Claim Construction Under 37 C.F.R. §42.104(b)(3)

The Challenged Claims are interpreted using the same claim construction standard used to construe the claims in a civil action in district court. 37 C.F.R. §42.100(b). Claim terms not addressed below should be construed according to their plain and ordinary meaning to a POSITA at the time of the claimed invention.

Petitioners submit that no claim term requires an express construction and all terms should be construed according to their plain and ordinary meaning.

Nevertheless, to the extent there is a dispute among the parties or the Board requires an express construction of terms, Petitioners adopt Cisco's construction from IPR2021-00593, as set forth below:

1. “trusted computing base” (claim 1, 12, 19)

The previous parties agreed this term 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.” EX1022, 12-13; EX1025, 5-6; EX1023, 14. ¶¶46-47.

2. “trusted platform module” (claims 1, 12, 19)

Cisco proposed this term be construed as “*a secure cryptoprocessor that implements a Trusted Platform Module specification from the Trusted Computing Group.*” EX1022, 13-16; EX1025, 5-6. Although the PO disputed this

construction, the Board did not identify any issue that would turn on this construction and determined that no claim terms required express construction. EX1023, 14. Nevertheless, should a similar dispute arise, Petitioners adopt Cisco's construction of this term. EX1002, ¶46, ¶48.

IV. GROUND 1: CLAIMS 1-19 ARE OBVIOUS OVER GLEICHAUF IN VIEW OF OVADIA AND LEWIS

A. Gleichauf

U.S. 9,436,820 ("Gleichauf") was filed 8/2/2004, and published 9/6/2016. EX1004, cover page. Gleichauf is prior art under 35 U.S.C. §102(e) (pre-AIA). EX1002, ¶49.

Gleichauf relates to the field of network security and teaches controlling a computerized device's access to network resources based on the security posture of the device. EX1004, 7:55-61, 3:4-9, FIG. 1. EX1002, ¶50.

Gleichauf is analogous art in the same field of endeavor as the '705 patent, and is reasonably pertinent to at least one problem with which the '705 patent was concerned as they both address network security by controlling access to network resources based on the security posture or condition of a device attempting to connect to resources. EX1004, 3:4-9; EX1001, 3:13-20. Gleichauf was not considered during prosecution of the '705 patent. EX1002, ¶¶51-52.

B. Ovadia

U.S. 7,747,862 (“Ovadia”) was filed 6/28/2004, and published 6/29/2010.

EX1005, cover page. Ovadia is prior art under 35 U.S.C. §102(e) (pre-AIA).

EX1002, ¶53.

Ovadia is analogous art in the same field of endeavor as the ’705 patent, and is reasonably pertinent to at least one problem with which the ’705 patent was concerned as they both concern network security by controlling access to network resources. Ovadia discloses network security measures and describes steps to ensure that a device seeking to access a network has an authorized configuration.

EX1005, 1:8-11, Abstract. Ovadia’s security scheme generates security keys, including attestation identity keys (“AIKs”), using a trusted platform module (TPM). EX1005, Abstract. The TPM is used “to verify [that] a current configuration of a subscriber station platform is an authorized configuration.”

EX1005, Abstract. Ovadia was not considered during prosecution of the ’705 patent. EX1002, ¶¶54-56.

C. Lewis

U.S. 7,533,407 (“Lewis”) was filed 4/14/2004, and published 5/12/2009.

EX1006, cover page. Lewis is prior art under 35 U.S.C. 102(e) (pre-AIA).

EX1002, ¶57.

Lewis is analogous prior art in the same field of endeavor as the '705 patent and, is reasonably pertinent to at least one problem with which the '705 patent was concerned as they both concern network security measures. Lewis describes verifying the configuration state of a device that seeks to access a network. EX1006, 4:7-23. 13:44-52. EX1002, ¶¶58-59, ¶61.

During prosecution, Lewis's corresponding publication, U.S. 2005/0131997, was cited by the Examiner for its teaching of a quarantine server providing a quarantine notification page. EX1003, 249-250. Applicants did not dispute this, arguing that the art failed to teach other features recited in then-pending claim 1. EX1003, 224-225. Petitioners rely on Lewis for the same point asserted by the Examiner and conceded by applicants. Hence, 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) discretion on these facts. EX1002, ¶60.

D. Motivation 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 a computerized device connects to a network. EX1004, 3:4-9; EX1005, 1:8-11. 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. The combination would have been obvious because it merely uses known techniques

(e.g., Ovadia’s TPM infrastructure and digital signatures) to improve a similar method (e.g., Gleichauf’s authentication and authorization of devices seeking network access) in the same way. EX1002, ¶62.

A POSITA would have been motivated to take advantage of Ovadia’s use of a TPM—as described in the Trusted Computing Group (TCG) “platform-independent industry specification”—because it would standardize Gleichauf’s posture credentials. EX1005, 4:32-35; EX1014, 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 TCG-defined TPM can be used to provide asset tracking functionalities, collect and report (in a trusted manner) platform configuration information, and secure client-server communications. EX1014, 2-4; EX1016, ¶11; EX1012, 2:8-11; EX1019, ¶¶3-4. Put differently, following the industry standard, as described in Ovadia, would benefit 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. 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 providing the credentials. Following the industry standard would also promote the interoperability of devices from different manufacturers. Ensuring interoperability

of devices is a common and important goal in the computer networking arts.

EX1002, ¶63.

Ovadia describes applying technology from the “TCG” which addresses “network security” and “covers trust in computing platforms in general.” EX1005, 4:16-35. A POSITA would have understood that the trusted computing and authentication technologies described in Ovadia are not limited to wireless networks Ovadia describes but instead are generally applicable to networks—like Gleichauf’s—that condition access on authentication and authorization of client devices. EX1002, ¶64.

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 secure (as in Ovadia) with the known network authorization process of analyzing a device’s security posture credentials, including its authentication (as in Gleichauf) to yield predictable results. A POSITA would have been familiar with the use of digital signatures to prove the 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. EX1008, ¶33. A POSITA also would have recognized that use of a digital signature would improve Gleichauf’s process by

enabling the policy server to confirm that posture credentials originated with the device attempting to access the network. EX1002, ¶65.

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. Gleichauf provides that "policy server 28 authenticates an identity of the computerized device 22 to generate an authentication result." EX1004, 22:3-5. Ovadia explains that one way to authenticate a device is to check that device's digital signature. EX1005, 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. EX1002, ¶66.

A POSITA would have had a reasonable expectation of success in combining Gleichauf and Ovadia. Notably, components of Ovadia's trusted computing architecture have corresponding components in Gleichauf's architecture, and a POSITA would have recognized how component descriptions in Ovadia would be applicable to corresponding components in Gleichauf. 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." EX1004, 3:8, 12:21-31; EX1005, 4:40-43, 14:1-6; FIG. 8. A POSITA would have understood that Ovadia's TPM generates and sends the

integrity measurement because the integrity measurements are digitally signed with an AIK and “AIKs are only permitted to sign data generated by a TPM.” EX1005, 14:9, 5:35-36. It would have been obvious to a POSITA to implement Gleichauf’s posture agent with TPM functionality, e.g., the ability to store, provide, and digitally sign integrity measurements, as taught in Ovadia, to provide security assurances available with digital signatures and limited-use integrity keys. EX1002, ¶¶67-68.

Petitioners’ proposed combination permits but does not require physical incorporation of elements from Ovadia into Gleichauf. *See In re Mouttet*, 686 F.3d 1322, 1332 (Fed. Cir. 2012); *In re Etter*, 756 F.2d 852, 859 (Fed. Cir. 1985). EX1002, ¶69.

E. Motivation 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 references relate to providing secure access when a device connects to a network. EX1004, 3:4-9; EX1005, 1:8-11; EX1006, 4:7-9. A POSITA would have been motivated to combine the teachings of Gleichauf, Ovadia, and Lewis and doing so would have been within the POSITA’s skillset. The combination would have been obvious because it uses the known technique of redirection of device traffic 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. EX1002, ¶70.

The combination would also have been obvious because it is merely the application of Lewis's known technique of serving a webpage to a quarantined device with information and remediation instructions to Gleichauf's and Ovadia's known methods of providing a notification message identifying reasons for quarantine, yielding predictable results. A POSITA would have known that serving a webpage with remediation instructions to a quarantined device from a quarantine server would beneficially provide information to the user while simultaneously preventing the device from accessing protected network resources. EX1006, 10:61-66, 13:63-14:1. EX1002, ¶¶71-73.

A POSITA would have had a reasonable expectation of success in combining the teachings of Gleichauf and Ovadia with Lewis. Gleichauf's policy server sends a notification message to a quarantined device with information and remediation instructions. EX1004, 5:4-11. A POSITA would have found it requires only a straight-forward software change to provide the contents of Gleichauf's notification 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. EX1002, ¶74.

Petitioners’ proposed combinations permit but do not require physical incorporation of elements from Lewis and Ovadia into Gleichauf. *See In re Mouttet*, 686 F.3d at 1332; *In re Etter*, 756 F.2d at 859. EX1002, ¶75.

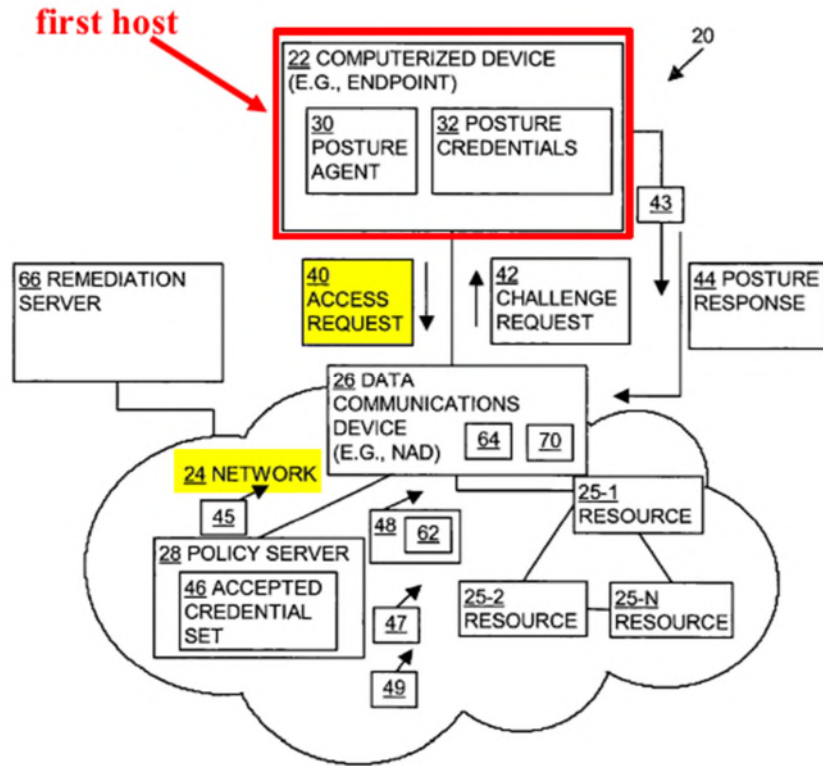
F. Claim 1

1. [1pre]

Gleichauf discloses a “computer-implemented method to control access to resources in a network.” EX1004, 27:61-62. 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. EX1004, 5:55-60, 7:56-61, 8:16-31, 12:57-67. EX1002, ¶77.

2. [1a]

Gleichauf describes a “computerized device, such as a personal computer attempting to communicate with restricted resources in the network.” EX1004, 3:50-56, 8:1-4. Figure 1 illustrates the computerized device sending an access request to the network (“*attempting to connect to a protected network*”):



EX1004, FIG. 1. Gleichauf's device is a *first host*, and, the network, which contains resources protected by access control measures, is a *protected network*. The device's attempts to access the restricted network resources through an access request shows that the device has *connected or is attempting to connect* to the network. EX1002, ¶¶78-79.

Gleichauf's policy server *detects an insecure condition* in the device. The device's attempt to connect to the network triggers a "posture based challenge-response" exchange where the device transmits its posture credentials to the network. EX1004, 11:29-32. Posture credentials contain "critical security information" including information about the "particular virus software and virus

definition versions” installed on the device. EX1004, 3:36-45. The policy server analyzes these posture credentials and *detects* the absence “of an anti-virus application.” EX1004, 14:16-34. 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’ user device 22.” EX1004, 22:38-59, 11:58-12:13. A device would also be vulnerable to any malware that might be transmitted by network resources. EX1004, 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*. EX1002, ¶80.

Gleichauf describes other examples of an *insecure condition*, such as an outdated version of an operating system or the absence of updated patches associated with the operating system. EX1004, 1:65-2:3, 8:65-9:1, 3:36-45, 31:1-7; EX1004, 1:65-2:3. Additional examples of *insecure condition[s]* are discussed *infra* at [1d]. EX1002, ¶¶81-82.

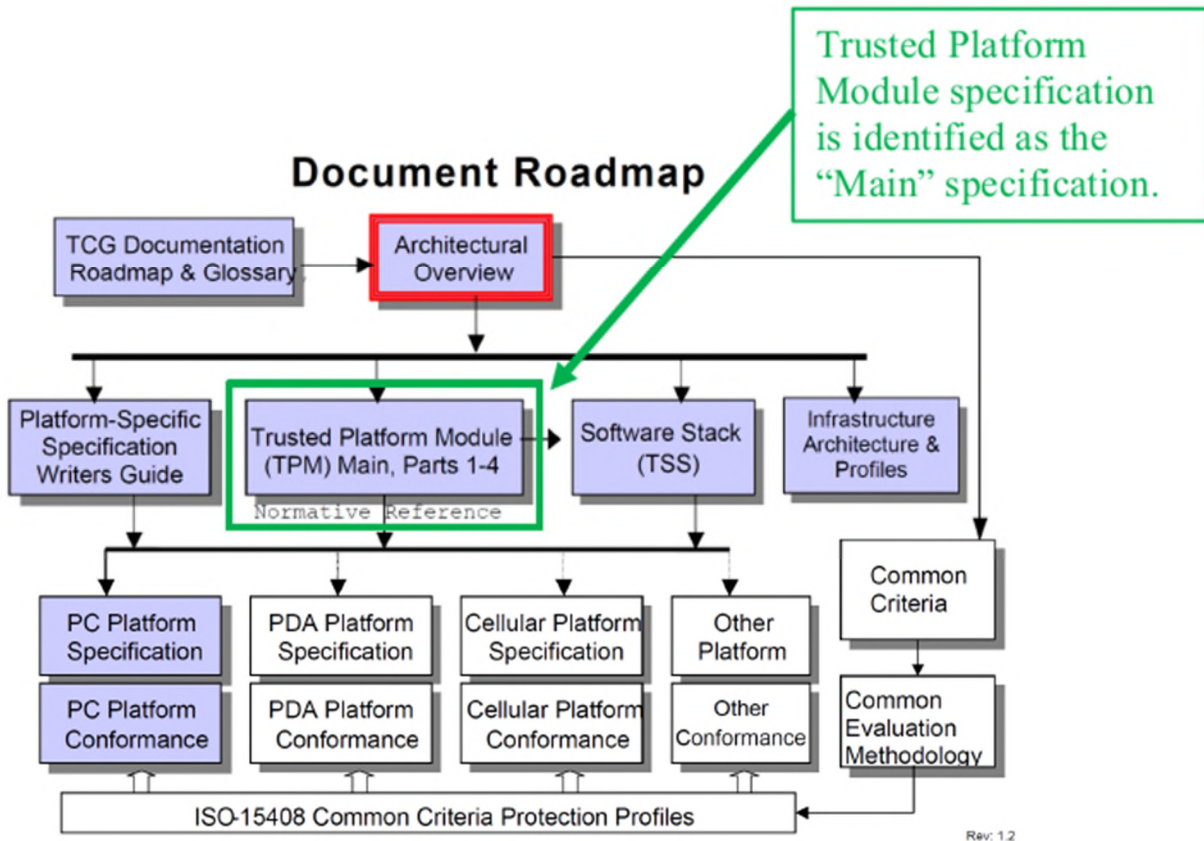
3. [1b]

Gleichauf’s computerized device includes a “posture agent” and one or more “posture plug-ins” installed on the device that collect information about the device’s security posture. EX1004, 3:9-16. The posture agent accesses information collected by each plug-in using an API implemented by each plug-in manufacturer and prepares posture credentials it transmits to the policy server for analysis and

access control. EX1004, 9:24-39. Gleichauf's posture agent is "embodied as a software or hardware application." EX1004, 9:29-39. Since the posture credentials are used to secure access to 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. Thus, the combined functionality of the posture agent and plug-ins is a *trusted computing base* within the computerized device that provides security for Gleichauf's data communication system.

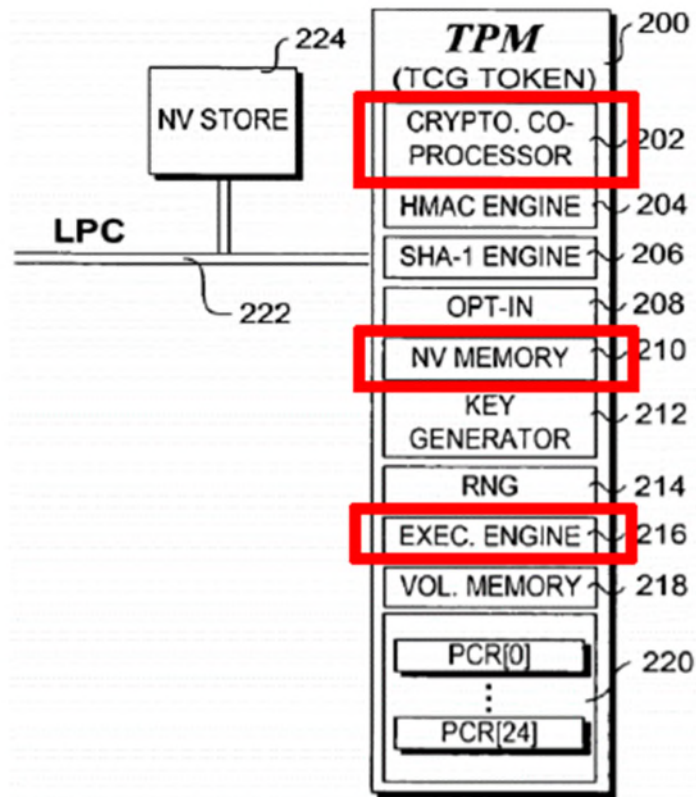
§III(C)(1) ("trusted computing base" construction). EX1002, ¶84.

Ovadia discloses a security apparatus that employs a TPM implementing the TCG "main specification (Version 1.2, October 2003...)." EX1005, 4:16-47. The main specification from TCG is the TPM Main Specification:



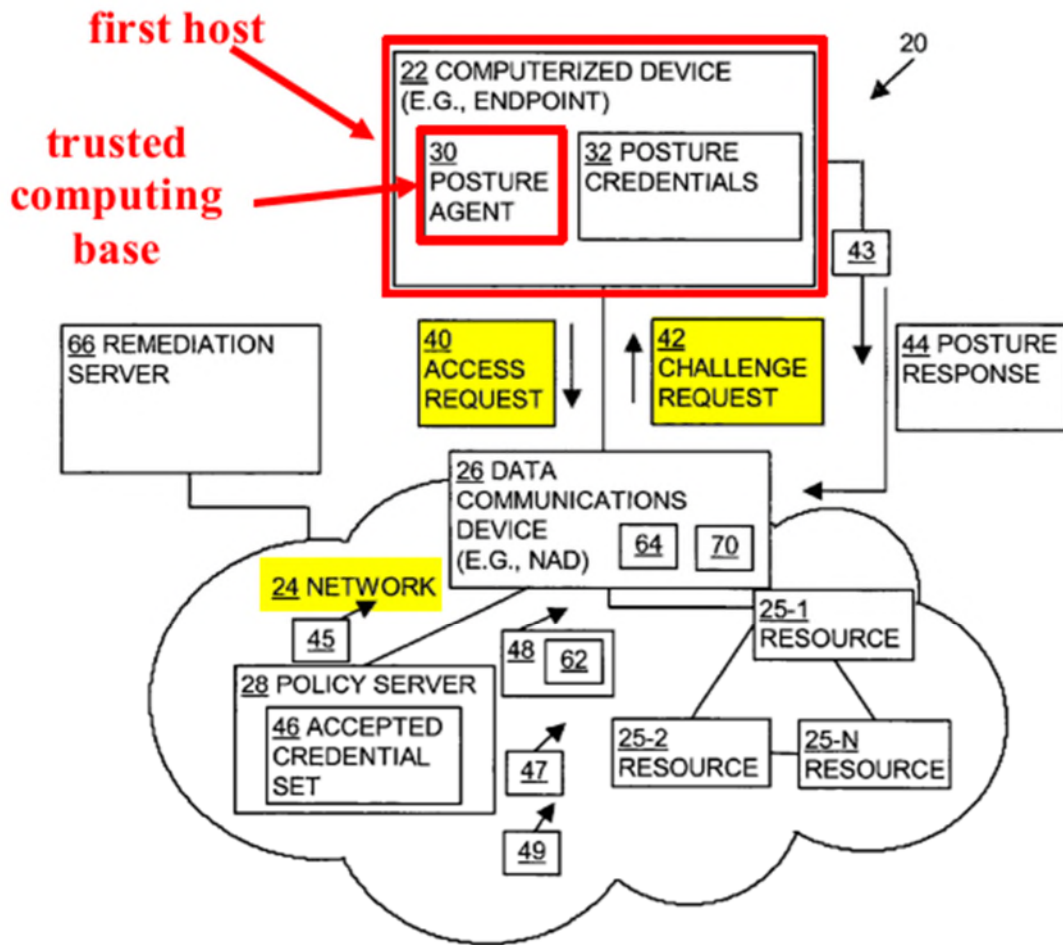
EX1014, iv. EX1015. EX1002, ¶85.

Ovadia’s TPM implements the TPM main specification from the TCG and includes a “cryptographic co-processor,” “non-volatile...memory,” and an “execution engine.” EX1005, 4:46-55, FIG. 2. Since Ovadia’s TPM implements the TPM main specification, it would have been obvious to a POSITA that Ovadia’s system includes a cryptographic co-processor. §III(C)(2) (“trusted platform module” construction); EX1013, 19; EX1008, 19; EX1005, FIG. 2:



EX1002, ¶86.

Gleichauf's policy server *contacts* the device's posture agent, which prompts the posture agent to collect the device's posture credentials from the posture plugins. EX1004, 4:7-11, 3:28-36. Gleichauf's Figure 1, below, shows the posture agent *within* the device:



EX1004, FIG. 1. It would have been obvious to a POSITA that the *trusted platform module* would likewise be *within* the device. Therefore, Ovadia’s TPM is a *trusted platform module* as claimed. §III(C)(2) (“trusted platform module” construction).

EX1002, ¶87.

Ovadia’s TPM “may be embodied as a hardware device (most common) or via software.” EX1005, 4:24-26. Like Gleichauf’s posture agent, Ovadia’s TPM is used “to verify [that] a current configuration of a subscriber station platform is an authorized configuration.” EX1006, Abstract. The configuration (called the

integrity measurement) includes information about the subscriber station, including the “current measurement of the firmware and/or software” of the station. EX1006, 13:57-14:12. EX1002, ¶88.

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. §IV(D) (motivation to combine). The TCG, which prepares the TCG specification discussed above, is “an **industry consortium** concerned with platform and network security.” EX1005, 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 TPMs. EX1005, 4:32-35, 4:22-24.

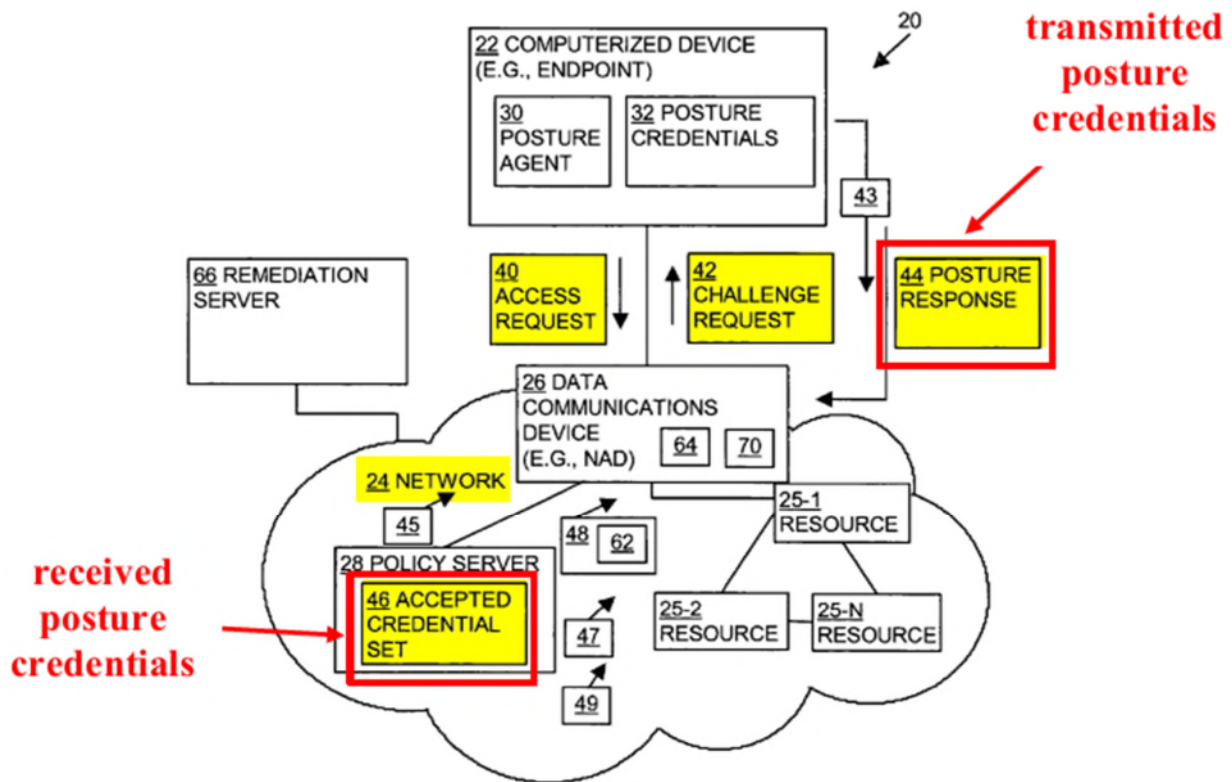
Implementing Gleichauf’s posture agent with TPM functionality would provide consistency among posture credentials generated by such posture agents operating on different devices. On the device side, a posture agent implemented with TPM functionality would operate consistent with the TCG specification, regardless of the device it is running on, and operation of such TPMs would be consistent throughout the adopting industry. On the network side, a policy server receiving Gleichauf’s posture credentials would process them more quickly and efficiently. The policy server would be able to process all posture credentials in substantially the same way because such credentials would have been gathered in substantially

the same way. In the combination, Gleichauf's posture agent would be implemented with TPM, i.e., a *trusted platform module*, per Ovadia. EX1002, ¶89.

For example, one way a POSITA would implement Gleichauf's posture agent with TPM functionality would be for the posture agent software to be executed by the execution engine in Ovadia's TPM. EX1005, Fig. 2. The posture agent implemented with TPM 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 Gleichauf's policy server. EX1004, 9:24-28. With the posture agent executing in combination with the TPM execution engine, the information collected by the posture agent would be "data generated by a TPM" and thus eligible for digital signature under the TPM main specification. EX1005, 5:36. EX1002, ¶¶90-91.

4. [1c]

Gleichauf's posture agent responds to the policy server's request by collecting the device's posture credentials from one or more posture plug-ins and transmitting the credentials in a posture response to the policy server. EX1004, 3:28-36, 4:11-15. Gleichauf's Figure 1 illustrates the transmission and reception of posture credentials:



EX1004, FIG. 1. Thus, the policy server “receiv[es] a response” to its request for the posture credentials. EX1002, ¶92.

5. [1d]

As discussed above, Gleichauf’s posture response includes posture credentials. §IV(F)(4) (limitation 1c). The posture credentials received by the policy server contain information such as “what particular virus software and virus 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*. EX1004, 3:36-49. EX1002, ¶94.

Policy server analyzes and “validate[s] the posture credentials” (i.e., “*determining whether the response includes a valid ... attestation of cleanliness*”). EX1004, 10:30-33. For example, the policy server checks for (i.e., “*determin[es]*”) the presence of an anti-virus application. EX1004, 10:30-33. If a current anti-virus application is present, the 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.*” EX1004, 11:45-58. If a current anti-virus application 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 cleanliness.*” EX1004, 11:58-67. EX1002, ¶95.

Posture agent “securely transmit[s]” posture credentials to the policy server. EX1004, 11:3-12. EX1002, ¶96.

Ovadia provides details about secure transmission of a device’s configuration information for the purpose of requesting access to a network. Like the device in Gleichauf, Ovadia’s subscriber station seeking network access uses its TPM 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.” EX1005, 14:1-6. The integrity measurement is stored in the TPM. EX1005, 13:61-63, Fig. 8. The integrity measurement is 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].” EX1005, 14:6-10. The subscriber station “is authenticated” and allowed access to the network if the integrity measurement is acceptable and “the digital signature is authentic.” EX1005, 14:13-23, 16:66-17:6. EX1002, ¶97.

A POSITA would have been motivated to combine the teachings of Gleichauf and Ovadia and would have had the skillset to do so. §IV(D) (motivation to combine). A POSITA would have understood that Ovadia’s digital signature decreases the risk of accepting credentials that have been altered by a malicious actor. Gleichauf’s posture credentials would be prepared by a posture agent implemented with TPM functionality (as taught by Ovadia). §IV(F)(3) (limitation 1b). A POSITA would have recognized that digitally signing Gleichauf’s posture credentials (as taught by Ovadia) would provide a mechanism for Gleichauf’s policy server (which evaluates those credentials) to confirm that the posture credentials have not been forged or tampered with. Ovadia explains that a device’s integrity measurements (*e.g.*, posture credentials) are digitally signed with an AIK, and that “AIKs are only permitted to sign data generated by a TPM.” EX1005, 14:9, 5:35-36. Since only the TPM (Gleichauf’s posture agent implemented with TPM functionality (§IV(F)(3) (limitation 1b))) could produce a valid digital signature and the TPM 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. Put differently, the policy server would be assured that a virus or other malware on the device had not tampered with or forged the posture credentials. 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. In the combination, Gleichauf's posture credentials, prepared by the posture agent implemented with TPM functionality, would be digitally signed, as taught by Ovadia. A POSITA would have further found it obvious to substitute Ovadia's teachings of digitally signing the integrity measurement for Gleichauf's recitation of secure transmission. §IV(D) (motivation to combine). EX1002, ¶98.

6. [1e]

The device's posture credentials transmitted to the policy server (the "attestation of cleanliness" in §IV(F)(5) (limitation 1d)), contain information about "the last time a local antivirus scan was executed." EX1004, 3:41. 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. EX1004, 22:38-47. A POSITA would have known from this disclosure that the opposite showing would have the opposite effect. Thus, a posture credential that does show timely execution of the anti-virus application would be treated as

“an attestation that the trusted computing base has ascertained that the first host is not infested.” EX1002, ¶100.

Posture credentials also contain information about “what particular virus software and virus definition versions are installed,” “what operating system patches are installed,” and what “types and versions of software applications” are on the device. EX1004, 3:36-49. This information 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.” EX1002, ¶101.

Gleichauf’s posture agent prepares posture credentials from information it collects from posture plug-ins. EX1004, 3:9-16, 9:24-39. The 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.” EX1004, 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. EX1002, ¶102.

Gleichauf’s disclosures regarding the claimed *attestation* are substantially the same as those of the ’705 patent specification. EX1001, 14:12-19. A POSITA would have understood Gleichauf’s disclosure that posture credentials contain information about “the last time a local antivirus scan was executed,” EX1004,

3:36-49, equates to the '705 patent's note about the currency of a virus scan.

EX1002, ¶¶103-104.

7. [1f]

Gleichauf's policy server reviews and validates posture credentials.

§IV(F)(5) (limitation 1d); EX1004, 10:30-33, 11:45-58. The policy server reviews the 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.

EX1004, 11:45-58, 22:38-42. EX1002, ¶106.

Gleichauf describes using posture credentials to determine whether to place the device in quarantine ("*quarantining the first host*"). EX1004, 4:24-27, 7:3-8. There are many reasons why a device might be placed into quarantine, including if it is out of compliance with a network admission policy (EX1004, 4:50-56), if its behavior is anomalous, suggesting that it may have been compromised (EX1018, ¶39), or if it is infected with a virus or has been hijacked by an unauthorized entity. EX1009, 4:8-14. EX1002, ¶107.

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., "*it is determined that the response does not include a valid digitally signed attestation of cleanliness*"). EX1004, 11:59-60. Gleichauf's out-of-compliance devices are assigned to

quarantine.” EX1004, 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.” EX1004, 11:58-67,

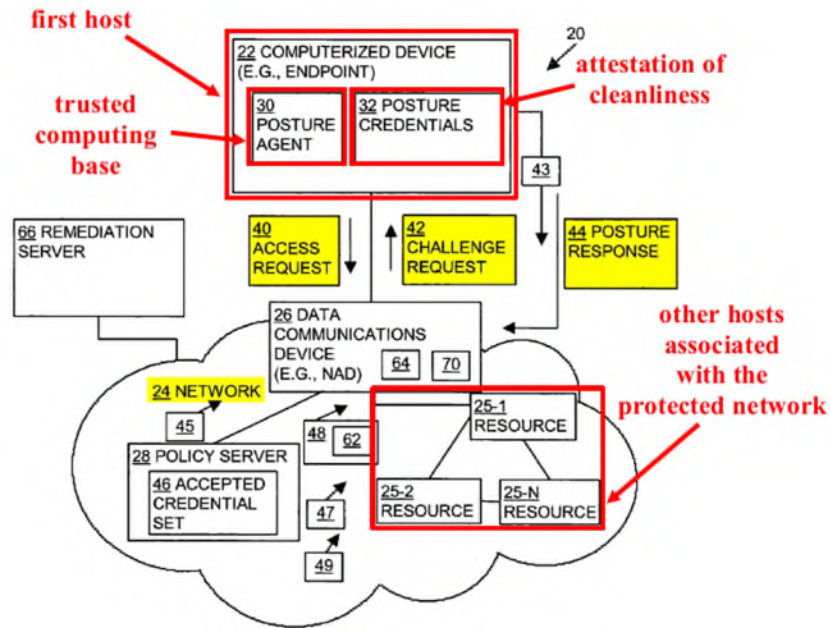


FIG. 1. Resources 25-1 through 25-N are other hosts within the network. EX1004, 9:65-67. Unless the first host is quarantined, network resources 25-1 through 25-N run the risk of becoming infected through the first host. EX1004, 8:21-28 (“By detecting the real-time security state of the computerized device prior to providing the computerized device with access to the network resources, thereby limiting at-risk or vulnerable computerized devices from accessing the network resources and minimizing the risk that the network resources receive or transmit malware”).

Therefore, Gleichauf discloses *preventing the first host from sending data to one or more other hosts associated with the protected network*. EX1002, ¶108.

8. [1g]

Gleichauf describes how the device transmits an “access request” or “signaling request” in an “attempt to access the network resources within the network.” EX1004, 3:50-56, 8:1-4. Access attempts can occur even after the device has been placed in quarantine. EX1004, 13:52-65. A device’s request to access network resources is a “*service request*” received and acted upon by a data communications device. EX1004, 3:60-65. EX1002, ¶110.

9. [1h]

Gleichauf discloses that communications from a quarantined device are restricted or rejected. §IV(F)(7) (limitation 1f); EX1004, 11:58-67. Limited communications to certain destinations, such as a remediation server, may be permitted. EX1004, 14:62-66. For example, Gleichauf provides that “a redirect access instruction” is applied to “HTTP traffic.” EX1004, 15:55-67. A POSITA would have known that HTTP traffic includes a *web server request* because HTTP is the protocol used for communications between web browsers and web servers. This is evidenced by the fact that Gleichauf refers to the redirection as a “URL Redirect.” EX1004, 15:45. 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. EX1017, ¶3. For example, this technique is commonly used to redirect a user's web browser to a login page at a hotel or airport. EX1010, 4:56-57; EX1011, ¶44. EX1002, ¶122.

Gleichauf describes providing "one or more notification messages" for display to a user "indicating that the device has been quarantined." EX1004, 4:56-63, 21:1-12. The message may include "a link to a remediation server or the IP address of the remediation server." EX1004, 21:1-12. EX1002, ¶113.

Similar to Gleichauf, Lewis also discloses the use of redirection with a client device that is quarantined. EX1006, 4:24-31. 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. EX1006, 13:7-12. 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*"). EX1006, 13:12-15. The webpage informs the user that the device "is in quarantine and corrective action must be taken." EX1006, 10:61-66. EX1002, ¶114.

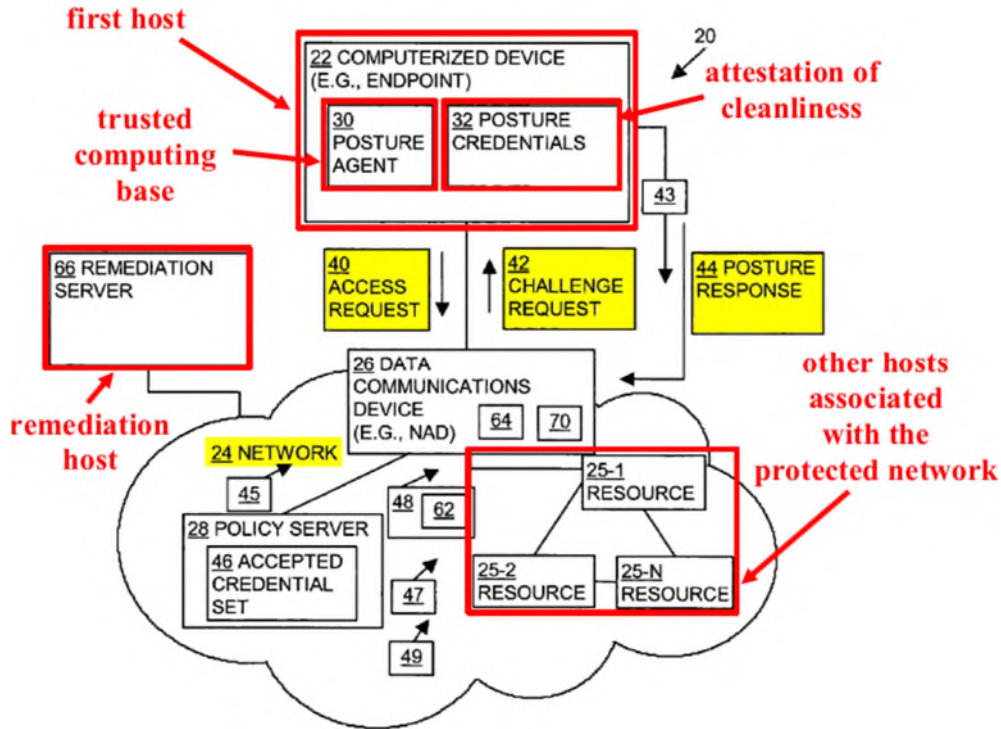
A POSITA would have been motivated to combine the teachings of Gleichauf, Ovadia, and Lewis and would have had the skill to do so. §IV(E) (motivation to combine). A POSITA would have appreciated the security benefits of redirecting HTTP traffic from Gleichauf's device to a quarantine server (per

Lewis), rather than a remediation server. The user of the quarantined device would receive information about the quarantine reasons via a webpage served by the quarantine server. The user would then have the options to proceed with remediation or terminate the connection attempt. A user could elect to terminate the attempt, for example, if the user did not desire to change their device configuration or if the user did not have time to remediate identified issues. In the combination, HTTP traffic from a quarantined device would be redirected to a quarantine server which would then serve a webpage to the user, as taught by Lewis. The webpage would inform the user that the device is in quarantine (a “*quarantine notification page*”). EX1002, ¶115.

10.[1i]

Gleichauf discloses that a device that is out of compliance with a network admission policy is quarantined and that communications from a quarantined device to protected resources in the network are restricted or rejected. §IV(F)(7) (limitation 1f); EX1004, 4:50-56, 11:58-67. EX1002, ¶117.

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



EX1004, FIG. 1. 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 usable to remedy the insecure condition*”). EX1004, 5:4-11. HTTP traffic from the device directed to the remediation server is permitted while HTTP traffic directed to other destinations is redirected. EX1004, 14:62-66, 15:55-67. EX1002, ¶118.

It would have been well-known to a POSITA that HTTP traffic, such as that from Gleichauf’s device, is usually preceded by a DNS query (i.e., “*and in the event the service request comprises a DNS query*”). 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. The device accesses a webpage by sending an HTTP request to the DNS-resolved IP address. These steps were commonly known and ubiquitously employed by web browsers in the prior art. EX1010, 5:49-63. EX1002, ¶119.

Lewis describes a “hijack DNS component 421 for intercepting DNS queries from quarantined clients.” EX1006, 11:15-17. The operation of the hijack DNS component is such that “[w]henver the client performs name resolution on any address, it will receive the IP [address] of QS [quarantine server].” EX1006, 14:22-23. As discussed at [1h], Lewis’s quarantine server (QS) provides a default webpage including a *quarantine notification*. The default webpage informs the user that the device is in quarantine and describes what corrective action must be taken for the device to be released from quarantine. EX1006, 10:61-66, 13:12-15. Thus, the IP address of Lewis’s quarantine server is *an IP address of a quarantine server configured to serve the quarantine notification*. EX1002, ¶120.

Both Gleichauf and Lewis describe allowing a device placed in quarantine to communicate with one or more servers to fix issues that led to it being quarantined. EX1004, 5:4-13; EX1007, 13:12-17, 14:17-31. It would have been obvious to a POSITA that communications with such servers would be preceded by a DNS request to resolve the remediation server’s name into its IP address. This is because servers are frequently known and referred to by their domain names; however,

domain names cannot directly be used to address an IP packet to servers. A domain name must first be resolved, using DNS, into a server's IP address. The IP address is then specified as the destination for the IP packet. A POSITA would have been familiar with the use and operation of DNS. EX1002, ¶121.

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.” EX1004, 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. 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. For all other queries, the hijack DNS component would function as described in Lewis by answering DNS queries with the IP address of a quarantine server. EX1002, ¶122.

A POSITA would have been motivated to combine the teachings of Gleichauf, Ovadia, and Lewis and would have had the skills to do so. §IV(E) (motivation to combine). A POSITA would have appreciated the need to ensure that a device placed into quarantine would be able to access the server necessary to remediate its security problem and thereafter gain full access to the network. Gleichauf contemplates identifying the remediation server by name since it

provides “a link to a remediation server,” which Gleichauf distinguishes from providing “the IP address of the remediation server.” EX1004, 21:8-12. Gleichauf further explains that the “link” to the remediation server would be provided as a URL. EX1004, 21:3-4. It was well known that a “URL typically includes the domain name of the provider of the identified resource.” EX1013, 9:12-14. Thus, a POSITA would have found it obvious from Gleichauf’s statement—that the remediation server is identified via URL—for the server to be identified to a quarantined device by its domain name. It would have been understood by a POSITA that the device needing to communicate with the remediation server would send a DNS query to obtain the IP address corresponding to the domain name of the server. In the combination, the DNS response provided to the device would include the IP address of the remediation server so that the device could resolve its security problems (e.g., obtain updated virus definitions) and exit quarantine. EX1002, ¶123.

11.[1j]

Gleichauf discloses that a device out of compliance with network admission policy is quarantined and communications from the device to network resources are restricted or rejected. §IV(F)(7) (limitation 1f); EX1004, 4:50-56, 11:58-67. However, HTTP traffic from the device to a remediation server is permitted (i.e.,

“*permitting the first host to communicate with the remediation host*”). EX1004, 14:62-66. EX1002, ¶125.

G. Claim 2

§IV(F)(claim 1). EX1002, ¶¶126-128.

Gleichauf teaches that a policy server *detects an insecure condition* in the device by reviewing the device’s posture credentials. §IV(F)(2) (limitation 1a); EX1004, 3:36-45, 11:29-32, 11:58-12:3, 14:16-34, 22:39-59. EX1002, ¶129.

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” and is “vulnerable.” EX1004, 3:36-49, 22:54-56. The failure to show “timely execution of the anti-virus application” is another indication the device is “vulnerable” and in an *insecure condition*. EX1004, 22:38-59. 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*.” EX1002, ¶130.

The posture credentials also contain information about whether the device can “resist reception or transmission of... content (spam and/or data theft).” EX1004, 8:52-56. A POSITA would have understood that spam is a type of “*malicious data*” because it is unsolicited. A POSITA would also have understood that the device would be “*scanned*” to determine whether it can resist reception or

transmission of spam. Furthermore, malicious data is known in the art to generate anomalous network traffic – thus, a detection of anomalous network activity from the device is an identification of an insecure condition. EX1002, ¶131.

The posture credentials further include information about the presence or absence of anti-virus software, whether the anti-virus software is up-to-date, and the status of a firewall application. EX1004, 3:36-49, 8:62-65. A POSITA would have understood that anti-virus and intrusion prevention software (e.g., a firewall) are types of “*security software*” because they can detect and prevent virus and other security infestations. Reviewing the posture credentials for information about the presence or absence of anti-virus software or a firewall involves “*determining whether a security software is installed.*” EX1002, ¶132.

H. Claim 3

§IV(F)(claim 1). EX1002, ¶¶133-135.

Gleichauf teaches that a policy server *detects an insecure condition* in the device by reviewing posture credentials. §IV(F)(2) (limitation 1a); EX1004, 3:36-45, 11:29-32, 11:58-12:3, 14:16-34, 22:39-59. EX1002, ¶136.

The posture credentials contain information about “operating system patches [that] are installed within the computing system” (i.e., “*updates to an operating system*”). EX1004, 3:36-49. Gleichauf explains, and a POSITA would have known, that an outdated operating system may make the device “vulnerable” to network

threats. EX1004, 1:65-2:3, 4:50-56. 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 patches to the operating system.” EX1004, 21:8-12, 5:8-11. Once remediated, the device “should be admitted” into the network. EX1004, 5:11-13. 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.*” EX1002, ¶137.

I. Claim 4

Gleichauf discloses a posture plugin that executes on the device and indicates a security posture. EX1004, 8:45-52. In one embodiment, the posture plugin is maintained by the operating system of the computerized device. EX1004, 8:65-67, 9:21-24. Gleichauf’s policy server receives security posture information from the posture plugin and uses these credentials to determine whether to quarantine the device. EX1004, 9:29-34. A POSITA would have understood and found obvious that because the operating system has access to the posture information generated by the posture plugin, the operating system can quarantine the device upon startup, such as in situations where the device has installed a new

version of the operating system, but has not verified the new version with the policy server. EX1002, ¶¶138-140.

J. Claim 5

1. [5pre]

§IV(F)(claim 1). EX1002, ¶¶141-142.

2. [5a]

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." EX1004, 21:5-12. The device responds to the message by activating the link and being directed to the remediation server. EX1004, 21:20-23. EX1002, ¶¶143-144.

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. Gleichauf explains that data communications device 26 "is configured to **detect attempted connections** to the network 24 by the user device." EX1004, 10:6-11, 12:57-67. 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*." 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 permitted level of network access. EX1004, 13:52-65, 14:3-7, 14:20-34, 22:38-47. EX1002, ¶145.

3. [5b]

Gleichauf discloses that communications from a quarantined device are restricted or rejected. §IV(J)(2) (limitation 5a); EX1004, 22:38-47. A quarantined device is generally permitted to engage in HTTP communication with a remediation server. EX1004, 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 host.*” EX1002, ¶¶146-147.

K. Claim 6

1. [6pre]

§IV(F)(claim 1). EX1002, ¶¶148-149.

2. [6a]

§IV(J)(2)(limitation 5a). EX1002, ¶150.

3. [6b]

Gleichauf restricts or rejects communications from a quarantined device. EX1004, 11:58-67. 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.” EX1004, 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*”). EX1002, ¶¶151-152.

Gleichauf discloses that HTTP traffic (an “*approved service*”) from the device to a remediation server is permitted. §IV(F)(10) (limitation 1i); EX1004, 14:62-66. 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.” EX1004, 15:55-16:6. It would have been obvious to a POSITA, in view of Lewis, to *redirect* unauthorized access requests to a *quarantine server* that would inform the user of their device’s quarantine status via a quarantine notification page. §§IV(F)(9)-(10) (limitations 1h-1i); EX1006, 11:15-17, 14:22-23. EX1002, ¶153.

L. Claim 7

§IV(F)(claim 1). EX1002, ¶¶154-155.

Gleichauf discloses that a device that is out of compliance with network admission policy is quarantined. §IV(F)(7) (limitation 1f); EX1004, 4:50-56. Outbound communications from a quarantined device are not permitted to destinations other than the remediation server. EX1004, 11:58-67, 14:62-66. It would have been obvious to a POSITA that inbound communications would be similarly subject to access restrictions. If outbound requests from the quarantined

device are not permitted to non-remediation destinations, the quarantined device is prevented from receiving information that it would have received in response to such requests. 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. EX1002, ¶156.

To the extent this claim is interpreted as requiring blocking of all non-remediation-related traffic from reaching the device, this restriction would have been obvious to a POSITA. Gleichauf is concerned with limiting or preventing the risk that network resources “receive or transmit malware” **from or to** “vulnerable computerized devices.” EX1004, Abstract, 8:52-55. That is why vulnerable devices are quarantined in an “isolated ... network segment” until the vulnerability has been remedied. EX1004, 4:25-26. 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. EX1004, 14:3-7, 14:20-25; EX1002, ¶157.

Further, to the extent that Gleichauf does not explicitly disclose restricting inbound communications to quarantined devices, Lewis discloses that the component responsible for quarantining the device (i.e., local enforcement component 314) also restricts inbound communication to the device. EX1006,

9:60-66. A POSITA would have found it obvious to modify Gleichauf's system to implement Lewis's limitations on inbound communications to provide a more robust security protocol so that compromised devices can protect themselves from further attacks. EX1002, ¶158.

M. Claim 8

§IV(F)(claim 1). EX1002, ¶¶159-160.

Gleichauf's 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." EX1005, 3:50-56, 9:57-64. Network resources 25 are "servers that contain content accessible by the . . . device." EX1005, 9:65-67. EX1002, ¶161.

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. The policy server determines whether to quarantine the device (by means of instructions to the data communications device) and protect the network. EX1004, 4:50-56, 10:30-33, 11:45-67. While in quarantine, the device is unable to communicate with network resources. EX1004, 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 access resources on the network, e.g., on the Internet. The data communications device and 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.” Moreover, it would have been obvious for the data communications device and policy server to be owned by a company that provides Internet service. EX1002, ¶162.

N. Claim 9

§IV(F)(claim 1), §IV(H)(claim 3). EX1002, ¶¶163-165.

O. Claim 10

§IV(F)(claim 1). EX1002, ¶¶166-167.

Gleichauf’s device’s posture response includes posture credentials. The posture credentials contain information about the “particular **virus software and virus definition versions**” installed on the device. EX1004, 3:36-49, 8:56-9:5. If updated software or current virus definitions are not present, the device is quarantined and communications from it to the network will be restricted or completely rejected, because of a determination that the response does not include “*a valid . . . attestation of cleanliness.*” EX1004, 11:58-67. EX1002, ¶168.

P. Claim 11

§IV(F)(claim 1). EX1002, ¶¶169-170.

Gleichauf's posture credentials received by the policy server contain information such as "what particular virus software and virus definition versions are installed," "when was the last time a local antivirus scan was executed," "what operating system patches are installed." §IV(F)(5)(limitation 1d). To the extent the patches inquiry is limited to the operating system, a POSITA would have understood and found obvious that the same inquiry related to patch level could be made with respect to the virus software installed on the user's device. EX1002, ¶171.

Q. Claim 12

1. [12pre]

Gleichauf teaches a "**robust network security architecture and system for controlling access to a network** depending on the configuration state or 'security posture' of a device when it attempts to connect to that network." EX1004, 3:4-9; 8:16-31. EX1002, ¶¶172-174.

2. [12a]

Gleichauf's policy server operates to protect the network. §IV(F)(1)-(2) (limitations 1pre-1a). The policy server "includes a controller 209 formed of a memory 210 and a **processor 212**." EX1004, 25:44-46. The processor 212 performs operations associated with protecting the network to perform each of the steps in the claim as described below. EX1004, 27:16-21; EX1002, ¶¶175-176.

3. [12b]-[12k]

See §§IV(F)(2)-(F)(11)(limitations 1a-1j). EX1002, ¶¶177-186.

4. [12l]

Gleichauf's policy server "includes a controller 209 formed of a **memory 210** and a processor 212." EX1004, 25:41-46. A POSITA would have understood that memory 210 is coupled to processor 212 because they are part of the same controller 209. EX1002, ¶¶187-188.

Memory 210 is "*configured to provide instructions to the processor*" of the policy server. EX1004, 26:66-27:8. EX1002, ¶189.

R. Claim 13

§IV(G)(claim 2). EX1002, ¶¶190-192.

S. Claim 14

§IV(I)(claim 4). EX1002, ¶¶193-195.

T. Claim 15

§IV(L)(claim 7). EX1002, ¶¶196-198.

U. Claim 16

§IV(N)(claim 9). EX1002, ¶¶199-201.

V. Claim 17

§IV(O)(claim 10). EX1002, ¶¶202-204.

W. Claim 18

§IV(P)(claim 11). EX1002, ¶¶205-207.

X. Claim 19

1. [19pre]

A POSITA would have understood that Gleichauf's computer program product *comprises computer instructions*. EX1004, 25:46-51, 3:4-9; EX1002, ¶¶208-210.

Gleichauf's policy server operates to protect the network and directs the operation of the data communications device 26. The policy server includes a “*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. EX1004, 26:66-27:8. Gleichauf's system protects the network by reducing the risk of the network becoming infected with malware. EX1004, 8:16-31. EX1002, ¶¶210-211.

2. [19a]-[19j]

§§IV(F)(2)-(F)(11)(limitations 1a-1j). EX1002, ¶¶212-221.

V. GROUND 2: CLAIMS 1-19 ARE OBVIOUS OVER FREUND IN VIEW OF BALL AND DANFORTH

A. Freund

U.S. 2003/0055962 (“Freund”) was filed 8/30/2001 and published 3/20/2003. EX1007, cover page. Freund is prior art under at least 35 U.S.C. §§102(a)-(b), (e). EX1002, ¶222.

Freund is analogous art in the same field of endeavor as the '705 patent, and is reasonably pertinent to at least one problem with which the '705 patent was concerned because both describe network-based systems for enforcing security compliance and isolation of non-compliant or potentially compromised devices. EX1001, 1:38-41, 3:9-13; EX1007, Abstract, ¶89. Freund was not considered during prosecution of the '705 patent. EX1002, ¶¶223-227.

B. Ball

U.S. 2006/0005009 ("Ball") was filed 6/30/2004 and published 1/5/2006. Ball is prior art under at least 35 U.S.C. 102(e) (pre-AIA). Ball incorporates by reference into its disclosure the entirety of Revision 1.2 of "TCG Specification Architecture Overview" (EX1014). EX1008, ¶6. EX1002, ¶228.

Ball is analogous art to the '705 patent in the same field of endeavor as the '705 patent, and is reasonably pertinent to at least one problem with which the '705 patent was concerned because both address verification and management of the security status of computing devices within a networked environment. EX1001, 1:38-41; EX1008, ¶5; 14:10-12; EX1008, ¶31. Ball was not considered during prosecution of the '705 patent. EX1002, ¶229-233.

C. Danforth

U.S. 7,571,460 (“Danforth”) was filed 8/6/2004 and published 8/4/2009. EX1009, cover page. Danforth is prior art under at least 35 U.S.C. §102(e) (pre-AIA). EX1002, ¶234.

Danforth is analogous art to the ’705 patent in the same field of endeavor as the ’705 patent, and is reasonably pertinent to at least one problem with which the ’705 patent was concerned because both disclose systems and methods for enforcing network access control based on the compliance of security state of devices attempting to connect to a network. EX1001, 1:38-41; EX1009, 2:11-14. Both Danforth and the ’705 patent address the problem of protecting a network by dynamically restricting or redirecting access of non-compliant or potentially harmful devices, using network infrastructure to enforce security policies. EX1001, Abstract; EX1009, 7:64-66. Danforth was not considered during prosecution of the ’705 patent. EX1002, ¶¶235-238.

D. Motivation to Combine

1. Freund and Ball

Freund is directed to protecting against compromised computers on a network, where those compromised computers provide false assurances they are not compromised by infection or vulnerable to such compromise (or vulnerable to being compromised without detection) (§V(A) (Freund)). Freund’s protection is

limited because it is software-based. Freund's client-side security module is a piece of software executing on the user's machine, which can be disabled, tampered with, or spoofed if the user's machine is compromised by malware or a sophisticated attacker, or disabled because of user error. EX1007, ¶68, ¶19 ("even if appropriate security software is deployed, security may be at risk when one or more users inadvertently or intentionally disables his or her security software"). A POSITA would have been motivated by Ball's teachings to modify Freund's system by moving from Freund's software-based solution to Ball's hardware-based TPM. A POSITA would have understood at the time of the purported invention that a TPM is advantageous when protecting against compromised computers in a network because it is a hardware-based solution that provides reliable, verifiable attestations from a remote computer before connecting to any such computer. EX1002, ¶239.

A POSITA, therefore, would have combined Freund with Ball to create an improved process for connecting to trusted computers, understanding that a hardware-based solution offers advantages a software-based solution cannot, such as preventing malicious actors from overwriting or disabling a software-based solution. A POSITA would have combined Freund with Ball to take advantage of a standards-based solution (as opposed to a proprietary solution) that is open, supported by industry leaders, and increasingly adopted. A POSITA would have

had a reasonable expectation of success in modifying Freund's router to include Ball's hardware-based solution because TCG's TPM, at the time of alleged invention, was a well-known hardware component that could be installed on computing devices for performing secure functions, and such a modification would have been within the technical capacity of a POSITA. EX1002, ¶240.

2. Freund and Danforth

Freund discloses a sandbox server 360 to which the router 310 redirects a non-compliant computer. EX1007, ¶74, ¶78. Sandbox server 360 displays an error message based on the type of compliance problem detected by the router. EX1007, ¶115. Sandbox server 360 makes this determination based on the particular port sandbox server 360 received from the communication packet. EX1007, ¶115. If, for example, the compliance problem is a software update or a software download, sandbox server 360 presents a webpage that includes a selectable link for downloading the software update or software. EX1007, ¶116, FIGS. 7-8. EX1002, ¶241.

Danforth similarly discloses a quarantine server (e.g., enforcement server) to which a quarantined client is redirected when it attempts to access a website, where the enforcement server provides a web page to the user notifying the user of the reason for the quarantine and what the user must do to lift the quarantine. EX1009, 3:5-7, 3:22-32, 3:57-55, 9:56-58. However, rather than Danforth's enforcement

server providing links to download upgraded software, Danforth discloses a separate remediation server in the form of a “service upgrade server” that provides upgraded software to a client device that is quarantined. EX1009, 2:49-51, 9:23-28. EX1002, ¶242.

A POSITA would have been motivated to modify Freund’s system to include separate remediation and quarantine servers to improve system performance and scalability. For example, a POSITA would have understood that software upgrades often involve the transfer of significant data. By modifying Freund’s system to offload these tasks to a dedicated server, such as the remediation server, Freund’s sandbox server is no longer prone to becoming a bottleneck, particularly during mass upgrade events (e.g., a LAN with several computers requiring simultaneous security software upgrades). A POSITA would have had a reasonable expectation of success in modifying Freund’s Internet-based system to offload remediation functionality from the sandbox server to a dedicated remediation server, as distributing tasks across servers or computing devices at the time of the alleged invention was well-known in the art, and such a modification would have been within the technical capacity of a POSITA. EX1002, ¶¶243-244.

E. Claim 1

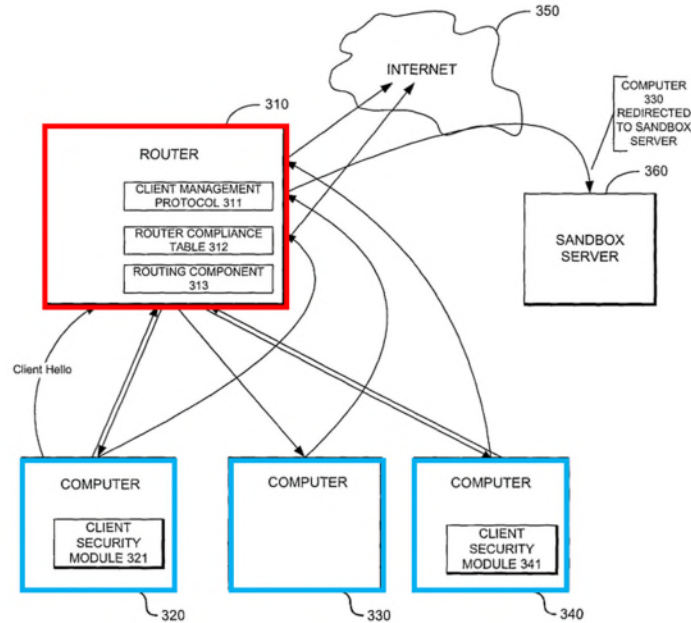
1. [1pre]

Freund discloses “methods for regulating access and maintaining security of individual computer systems and...LANs...connected to larger open networks (Wide Area Networks or WANs), including the Internet.” EX1007, ¶4, ¶68, ¶¶147-151, FIG. 9. EX1002, ¶245.

2. [1a]

Freund discloses *detecting an insecure condition* (non-compliant response) *on a first host* (first computer) *that has connected or is attempting to connect to a protected network* (LAN). EX1007, ¶71, ¶¶74-75, ¶78, ¶81, ¶85, ¶88, ¶¶126-132, ¶145, ¶147, FIG. 3. EX1002, ¶246.

Freund’s system includes a router (red) and a plurality of personal computers (blue) that connect to the Internet through the router:



EX1007, FIG. 3, ¶74. Freund’s router enforces security rules and procedures such as verifying that a computer (e.g., any of computers 320-340) “has installed and is running appropriate security software, and is in compliance with other established security policies” prior to allowing that computer to connect to the Internet.

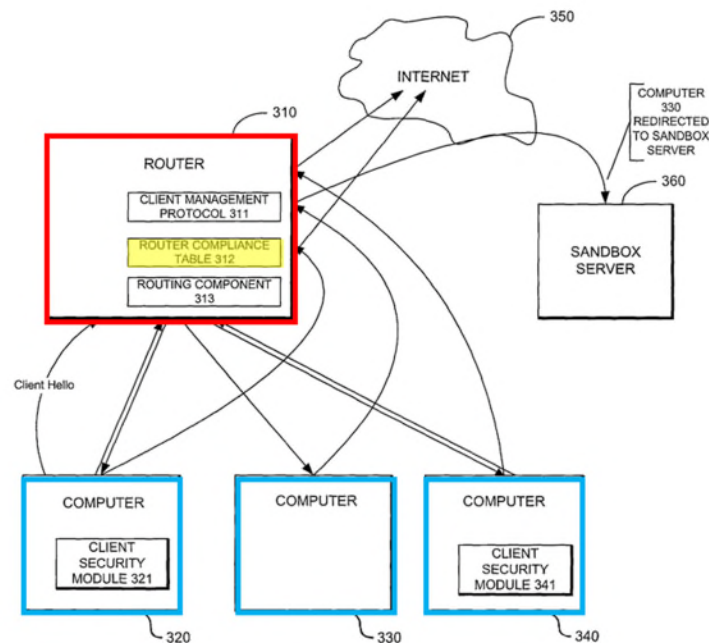
EX1007, ¶71. To determine whether a computer is compliant with security rules and procedures, the router periodically sends a “router challenge” to the computer that requests a response within a predefined time period. EX1007, ¶75 (“Every few seconds the CMP component 311 sends out a communication...that is described as a ‘router challenge,’” which “requests a response from computers...within a few seconds”), ¶¶126-132. EX1002, ¶274.

Examples of the router detecting an insecure condition include the computer:

(1) not responding to the router challenge, which may indicate no client-side

security module installed on the computer, which could be the result of a Trojan horse problem. EX1007, ¶75, ¶88 (“This failure to respond could result from the deactivation of the client-side security module....caused by a Trojan horse problem on the computer”); (2) not responding to the router challenge within the prescribed amount of time. EX1007, ¶78; and (3) indicating non-compliance with a security requirement (e.g., not having the most updated version of the security software). EX1007, ¶78, ¶85, ¶132. A POSITA would have understood that all these examples indicate the presence or detection of an insecure condition as defined by the Client Management Protocol (CMP) of the router. EX1002, ¶248.

The router maintains each computer’s responses in a router compliance table (yellow):



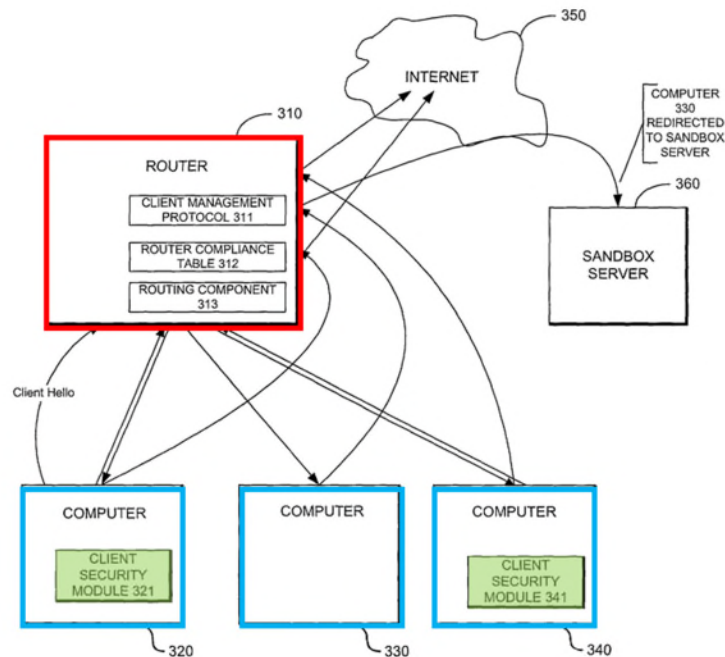
EX1007, FIG. 3, ¶86. Freund detects an insecure condition when a computer attempts to connect to the Internet by cross-referencing the computer's previous responses in the router compliance table. EX1007, ¶147 ("a connection attempt from one of the local computers to the Internet is received by the router...the CMP looks up the router compliance table entry under the client computer source IP address....the CMP evaluates the response in the router compliance table"), ¶145. EX1002, ¶249.

3. [1b]

Freund-Ball combination discloses *wherein detecting the insecure condition (non-compliant response) includes contacting a trusted computing base (TPM) associated with a trusted platform module within the first host (computer).*

EX1007, ¶¶74-75, ¶¶91-93, FIG. 3; EX1008, ¶6, ¶27, ¶30, FIG. 2. EX1002, ¶250.

One or more of Freund's computers include a client security module (green):

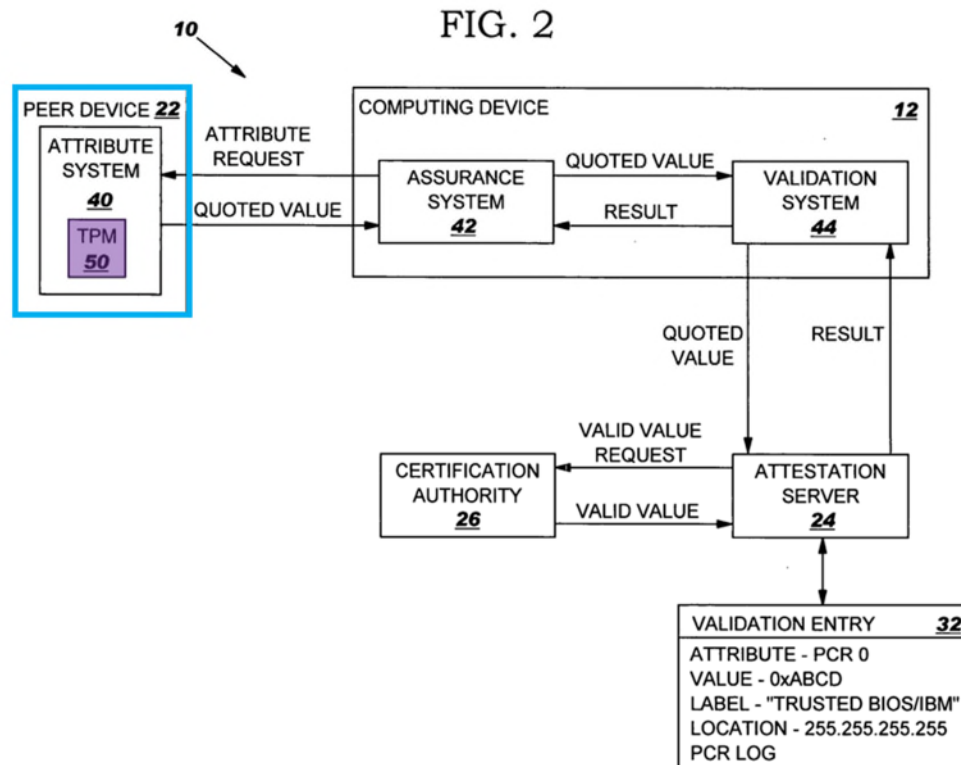


EX1007, FIG. 3, ¶74. Freund’s router contacts the computer’s client security module when the router sends the “router challenge” message to the computer.

EX1007, ¶75 (“Computers 320 and 340 have the client-side security module installed and... respond[] to the router challenge,” “computer 330 does not respond to the router challenge, as there is no client-side security module installed”), ¶¶91-93. Because the client security module is software utilized to provide security in Freund’s computing environment by receiving and responding to router challenges, the client security module constitutes the claimed “trusted computing base.”

§III(C)(1)(“trusted computing base” construction). EX1002, ¶251.

To the extent Freund does not expressly disclose that the client security module is *associated with a trusted platform module in the first host*, Ball discloses this functionality. Ball discloses a TPM (purple) on a computing device (blue):



EX1008, FIG. 2, ¶27, ¶30. TPM is a hardware component that “comprises a passive device that is installed in a computing device, and which can accurately measure, securely store, and securely communicate information on one or more attributes of the computing device.” EX1008, ¶6. TPM is compliant with the TCG Specification Architecture overview. EX1008, ¶6 (“The [TCG] specification...is described in detail in a document entitled “TCG Specification Architecture Overview,” Rev. 1.2, 28 Apr. 2004, which...calls for a computing device to be

built with an integrated...TPM”). Ball’s TPM stores cryptographic keys for encrypting information sent to another computing device. EX1008, ¶6. The TPM specification describes a TPM that includes components for performing cryptographic operations. Accordingly, Ball’s TPM constitutes the claimed *trusted platform module*. §III(C)(2) (“trusted platform module” construction). EX1002, ¶252.

A POSITA would have been motivated by the teachings of Ball to modify Freund’s teachings regarding computer devices in its protected network by including Ball’s hardware-based TPM. §V(D) (motivation to combine). EX1002, ¶253.

4. [1c]

Freund discloses *receiving a response*. EX1007, ¶75, ¶93. EX1002, ¶254.

Freund’s router receives a response from the client-security modules responsive to issuing the router challenge. EX1007, ¶75 (“Computers 320 and 340...respond[] to the router challenge”), ¶93, Abstract. EX1002, ¶255.

5. [1d]

Freund discloses *determining whether the response* (response to router challenge) *includes a valid digitally signed* (encrypted) *attestation of cleanliness* (indication that the computer complies with security policy requirements in the router challenge). EX1007, ¶78, ¶122, ¶127, ¶134, ¶144. EX1002, ¶256.

Freund's router receives encrypted responses from a computer's client security module responsive to sending the computer a router challenge. EX1007, ¶122, ¶127 ("router challenge packet is encrypted by the router's public key" and "is decrypted by the client computer using the client's private key"), ¶134. The response from the computer includes an indication of whether the computer complies with security policy requirements in the router challenge. EX1007, ¶78 ("A computer may also be precluded from connecting to the Internet if it answered the router challenge with a response that indicated it was not in compliance..."). When the router receives the encrypted response, the router checks whether the response is valid. EX1007, ¶144. EX1002, ¶257.

To the extent that encrypting the response to the router challenge does not amount to the claimed *valid digitally signed attestation of cleanliness*, Ball discloses this functionality. Ball's TPM stores cryptographic keys digitally signed using an AIK for encrypting information sent to another computing device. EX1008, ¶6. A POSITA would have understood and found it obvious to modify Freund's computing devices to digitally sign router challenge responses using the AIK created by Ball's TPM because modifying Freund's computing devices to include Ball's TPM would yield predictable results, i.e., digitally signing the router challenge responses using the AIK. EX1002, ¶258.

6. [1e]

Freund discloses this limitation. EX1007, ¶110, ¶¶121-122, ¶127, ¶132, ¶134, ¶144; §V(E)(3)(limitation 1b), §V(E)(5)(limitation 1d). EX1002, ¶259.

Freund discloses *wherein the valid digitally signed attestation of cleanliness* (indication that the computer complies with security policy requirements in the router challenge) *includes at least one of an attestation that the trusted computing based has ascertained that the first host is not infested* (response to anti-virus challenge). EX1007, ¶78, ¶102, ¶110, ¶¶121-122, ¶127, ¶132, ¶134, ¶144; §§V(E)(3),(5) (limitations 1b, 1d). EX1002, ¶260.

One of the router challenges sent by Freund's router to the trusted computer base is an "anti-virus challenge" that "allows the administrator to use the router for anti-virus enforcement and distribution." EX1007, ¶132. Examples of the fields in the "anti-virus challenge" include an indication of the antivirus product ID, antivirus version, whether antivirus real-time monitoring enforcement is selected, etc.:

Element	Size	Comment
Option ID	WORD	ANTI_VIRUS_CHALLENGE (= = 4)
Option size	WORD	32
Antivirus Product ID	DWORD	Code to identify AntiVirus name (TRENDMICRO = = 1)
Antivirus version	4xWORD	Optional: CMPVERS "7.0.0"
Antivirus Auto-Update Enforcement	BOOL	Optional: Boolean to identify if the Auto-Update enforcement option is selected
Antivirus Real-Time Monitoring Enforcement	BOOL	Optional: Boolean to identify if the Real-Time Monitor enforcement option is selected
Antivirus Reserved1	DWORD	Reserved
Antivirus Reserved2	DWORD	Reserved

EX1007, ¶132. Freund's trusted computer base sends a response to the router's anti-virus challenge that is encrypted by the trusted computer base. EX1007, ¶144; §V(E)(5) (limitation 1d). The response includes fields such as an indication of the antivirus product ID, antivirus version, whether antivirus real-time monitoring enforcement is selected, etc.:

Response	Value	Comment
Client time out	1	No recent valid client response.
No client response	2	No client response ever received.
Reserved	3-32	Reserved.
ZAP version outdated	33	Client application incorrect or old.
Invalid license	34	Invalid license key, contact administrator.
License exceeded	35	Max users exceeded for license - contact administrator.
No Antivirus program installed	36	AV not installed.
Antivirus wrong version	37	AVold version.
Antivirus Auto-Update not configured	38	AV Auto-Update is not configured.
Antivirus Real-Time Monitor not running	39	AV Real-Time Monitor not running on client
Packet time stamp	>256	Current time stamp in seconds + 256

EX1007, ¶144. A POSITA would have understood and found obvious that real-time monitoring for viruses includes scanning the first host for computer virus

infection in real-time. Therefore, if Freund's router determines that real-time monitoring is running on the computer, the client security module has ascertained that the first host is not infested. EX1002, ¶261.

To the extent that Freund does not explicitly disclose an attestation that the trusted computing base has ascertained the first host is not infested, a POSITA would have had good reason to and been motivated, based on the disclosure of Freund, to include an indication of the results of the most recent virus scan in Freund's client response, where a result of no virus infection is an attestation that the first host is not infected. Freund's anti-virus software is part of the client security module within the computer. §V(E)(3)(limitation 1b). Freund explicitly provides an extensible message format for its challenge and response to enforce security policies, including the ability to modify the format (via versioning of the protocol) and to include additional options to ascertain additional anti-virus information. EX1007, ¶¶121-122, ¶127, ¶132, ¶134, ¶110. A POSITA would have had a good reason to make this improvement because it is simply the use of a known technique (adding fields, such as the date and result of the most recent virus scan) to improve a known product ready for improvement (Freund's extensible message client response) to yield predictable results (a message with additional data fields). A POSITA would have been motivated to make such a modification to ensure that computers that were not subject to real-time monitoring for viruses still

had an attestation that the computer was not infested. For the same reasons, a POSITA would have further found it obvious to modify Freund to include a determination on the router of whether the client response included an attestation that the computer was not infested, where that attestation was an acceptably recent clean virus scan result. EX1002, ¶262.

Freund also discloses *wherein the valid digitally signed attestation of cleanliness includes at least one of...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* (response to anti-virus challenge). EX1007, ¶78, ¶102, ¶132, ¶144. EX1002, ¶263.

One of the fields in response to the anti-virus challenge is related to a version of the antivirus program installed on the computer. EX1007, ¶78 (“a computer running an older version of the security software may respond in the negative to a router challenge requesting confirmation that the computer is running a current version of the software”), ¶132, ¶144, ¶102. A POSITA would have understood and found obvious that versions of a software product, such as antivirus software, would include patches to address specific issues from the previous version of the antivirus software. To the extent this is not explicitly disclosed in Freund, a POSITA would have found this feature to be nothing more than the

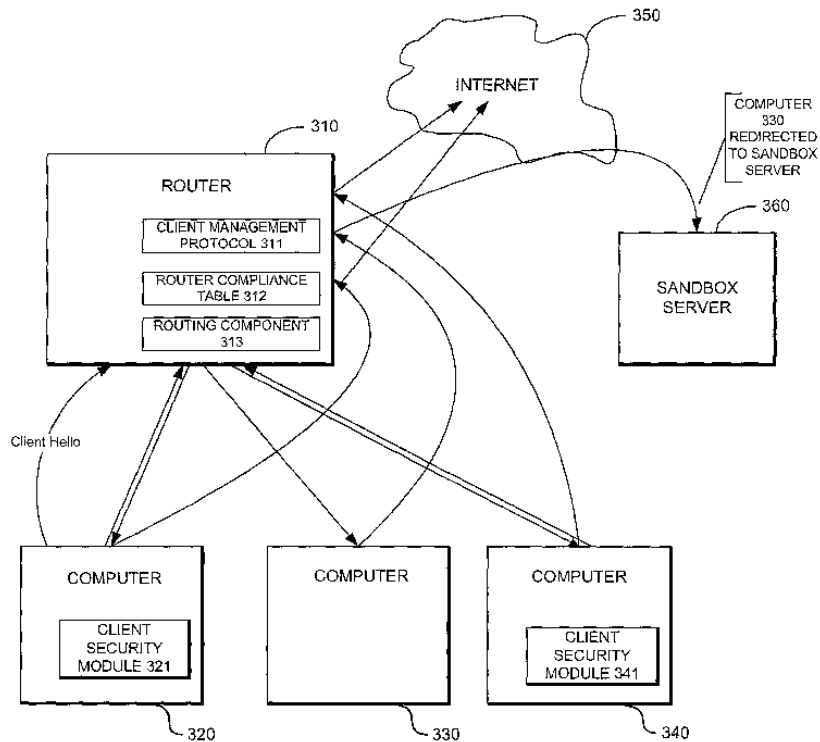
application of a well-known technique of using patches to improve the anti-virus software product. EX1002, ¶264.

Therefore, a response that the antivirus software on the computer is the correct version of the security software is an attestation that the client security module ascertained the presence of a patch or patch level. EX1002, ¶265.

7. [1f]

Freund discloses *when it is determined that the response does not include a valid digitally signed attestation of cleanliness* (e.g., the antivirus software version is incorrect or the computer is infested), *quarantining the first host* (computer), *including by preventing the first host* (computer) *from sending data to one or more other hosts* (other computers) *associated with the protected network* (LAN). EX1007, ¶74, ¶78, FIG. 3. EX1002, ¶266.

After determining that the response from the computer is not valid, Freund's router prevents the first computer from sending data to any other computers associated with the LAN. EX1007, ¶¶68-69 ("The security component on the client premises equipment serves as the traffic cop that prevents the unauthorized transfer of code or data from the local machine to a local network"), ¶78. As shown in FIG. 3 below, the router is responsible for routing packets to and from each of the computers in the LAN:



EX1007, FIG. 3, ¶74. In other words, network traffic sent by the first computer 320 must first travel through the router 310 before reaching any other computer 330-340. EX1007, FIG. 3. Accordingly, if the router determines that the first computer is not in compliance, the router prevents the first computer from sending data to the other computers in the LAN thereby “prevent[ing] the unauthorized transfer of code or data from the local machine to a local network.” EX1007, ¶69. EX1002, ¶267.

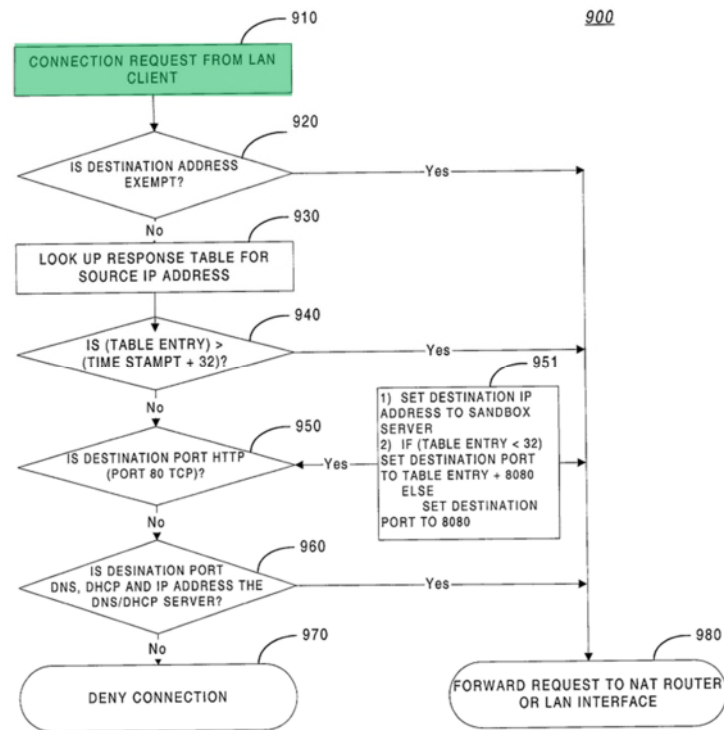
8. [1g]

Freund discloses *wherein preventing the first host from sending data to one or more other hosts associated with the protected network (limitation 1f) includes*

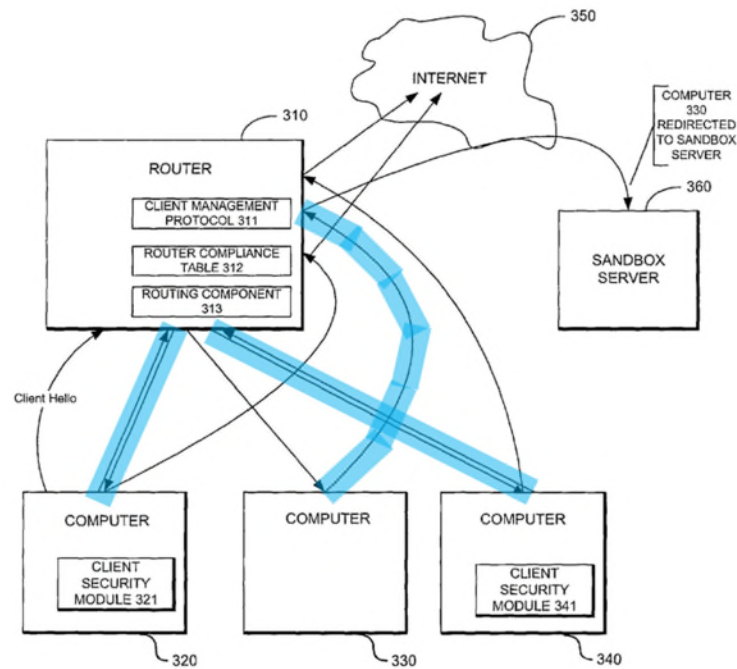
receiving a service request (connection request) sent by the first host (computer).

EX1007, ¶147, FIGS. 3, 9. EX1002, ¶268.

As part of the process for preventing the first computer from sending data to other computers in the LAN, Freund's router first receives a connection request from the first computer:



EX1007, FIG. 9 (block 910 (green)),

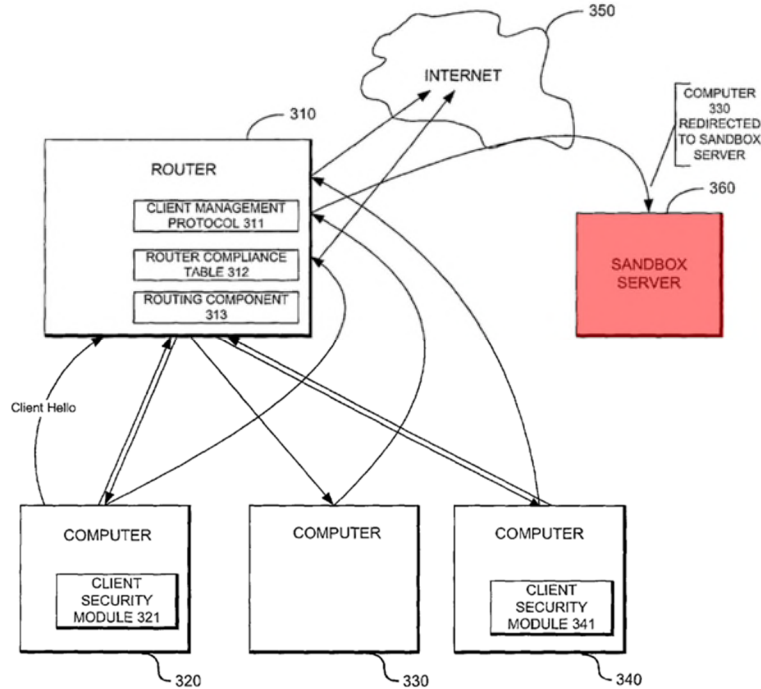


EX1007, FIG. 3 (service requests (blue)), ¶147. EX1002, ¶269.

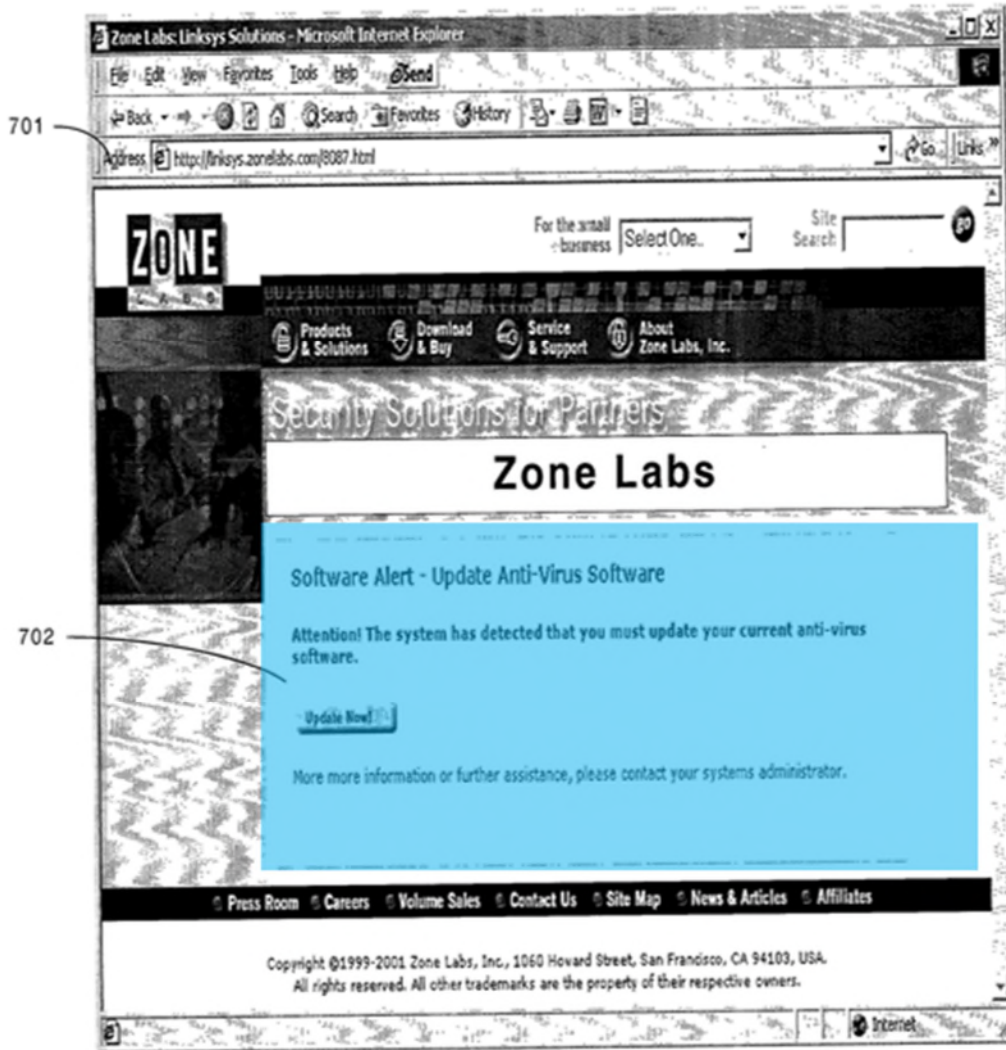
9. [1h]

Freund discloses *serving a quarantine notification page* (sandbox webpage) *to the first host* (first computer) *when the service request* (connection request) *comprises a web server request* (HTTP request). EX1007, ¶40, ¶71, ¶78, ¶81, ¶89 ¶¶94-95, ¶¶113-115, FIGS. 3, 7. EX1002, ¶270.

When Freund's router determines that the response is not compliant, the router redirects the user's request to a sandbox server (red):



EX1007, FIG. 3, ¶40, ¶89, ¶71 (“If a computer is not in compliance, then the computer’s access to the Internet is restricted to those activities necessary to get the computer back into compliance...by redirecting the attempted connection by a non-compliant computer to a designated ‘sandbox’ server”). When the user is redirected to the sandbox server, they are only able to perform “a defined set of tasks to address the non compliance” and “[a]ll other Internet access by [the non-compliant] computer...is disabled.” EX1007, ¶78, ¶81, ¶¶94-95. When the user is redirected to the sandbox server, they are served a notification (blue) that they have been quarantined:



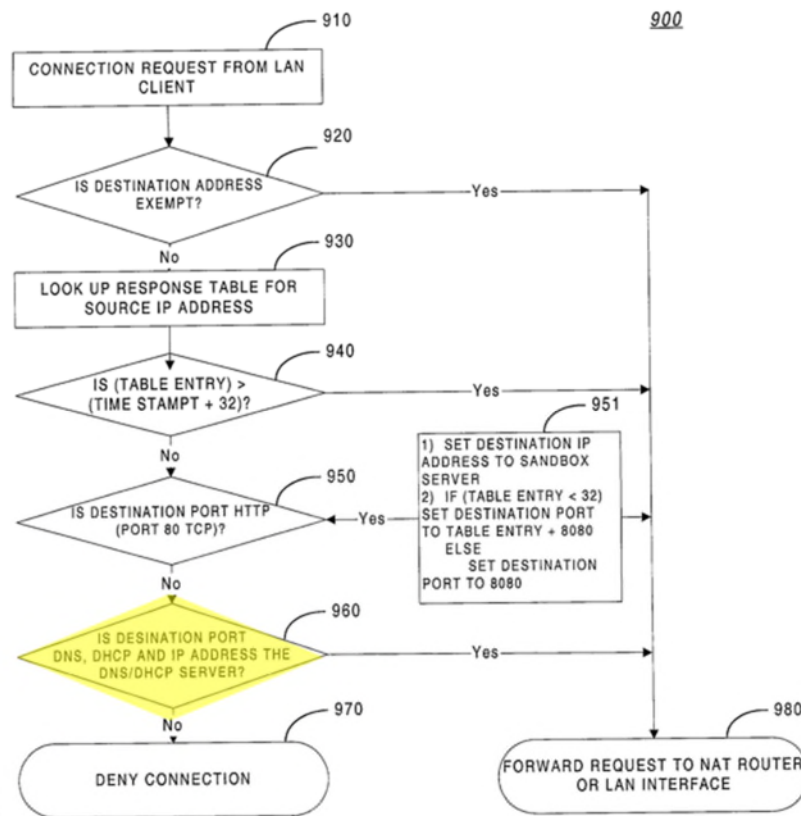
EX1007, FIG. 7, ¶¶95, ¶¶113-115. A POSITA would have understood and found obvious that redirection to the sandbox server is serving a quarantine notification page to the non-compliant computer. EX1002, ¶¶271-272.

10.[1i]

Freund-Danforth combination discloses *in the event the service request comprises a DNS query (DNS query), providing in response an IP address of a quarantine server (sandbox server) configured to serve the quarantine notification*

page (sandbox server webpage) if a host name that is the subject of the DNS query is not associated with a remediation host (antivirus software server) configured to provide data usable to remedy the insecure condition. EX1007, ¶95, ¶102, ¶114, ¶150, FIGS. 7-9; EX1009, 4:49-54, 8:23-27, 9:23-24, 9:26-28. EX1002, ¶273.

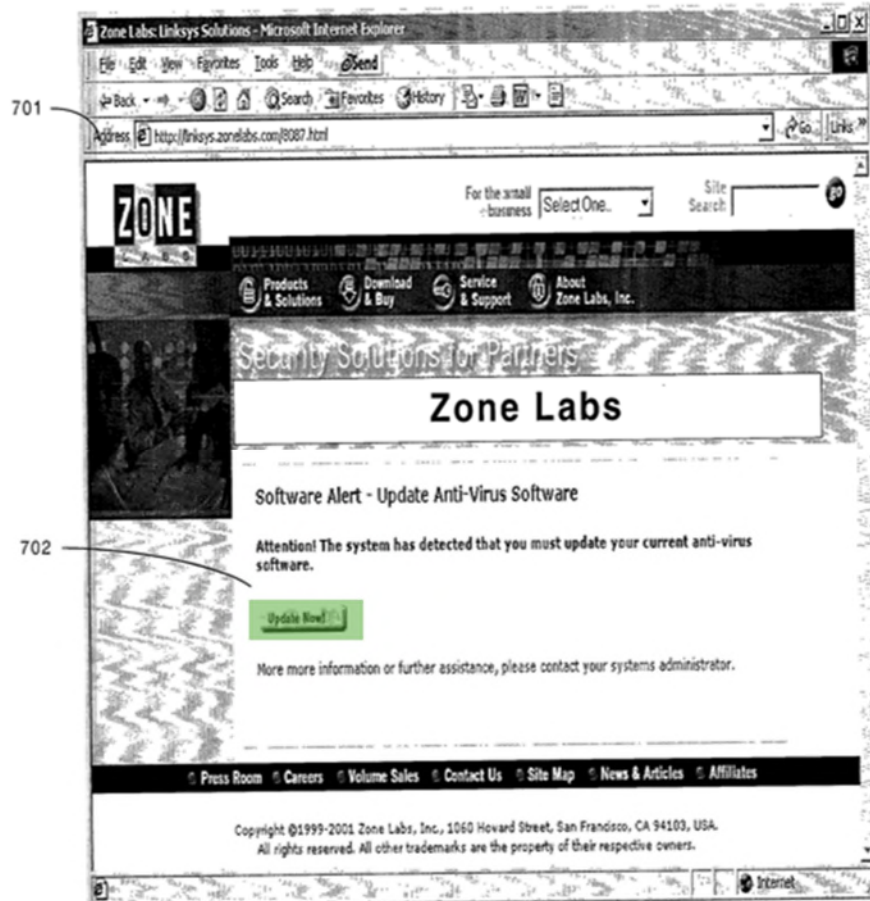
Freund's router can receive a service request in the form of a DNS query (yellow):



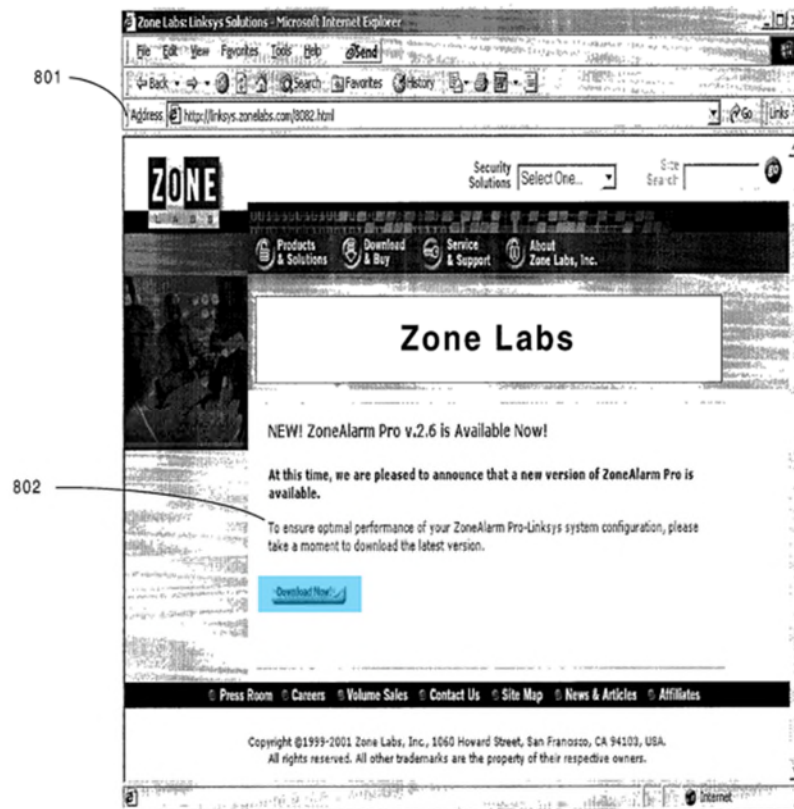
EX1007, FIG. 9 (block 960), ¶150 (“If in step 950 the destination port was not HTTP, then the re-routing manager proceeds to step 960 to evaluate whether or not the destination port was DNS or DHCP and the destination IP address that of the DNS/DHCP server”). If the service request is a DNS query, Freund's router allows

the computer “to communicate with the DNS server regardless of whether or not compliant” otherwise “the browser [on the computer] would fail prior to being redirected to the sandbox server because it could not lookup the IP address of the sandbox server.” EX1007, ¶150. It is only after a computer in Freund’s system has the IP address and initiates an HTTP request using the IP address that the router can redirect the HTTP request to the sandbox server. EX1007, ¶95, ¶114. EX1002, ¶274.

When the computer is served the sandbox server webpage, the quarantine notification webpage includes a link (green) that the user can select to download a software update:



EX1007, FIG. 7, ¶114. When the user does not have current security software installed on their computer, the quarantine notification webpage includes a link (blue) that the user can select to download a software update:



EX1007, FIG. 8, ¶114. A POSITA would have understood that the virus protection software update or the virus protection software itself can be hosted on a server separate from the sandbox server. EX1007, ¶102 (“using the router to enforce a requirement for a totally separate piece of software, in this case a third party anti-virus program”). EX1002, ¶275.

While Freund does not explicitly disclose *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*, Danforth discloses this limitation. Danforth discloses a system in which,

when a device is quarantined, DNS queries from that device are intercepted and responded to with the IP address of a special server. EX1009, 4:49-51 (“If the current subscriber state is unquarantined, the DNS request is redirected to an external DNS server”), 4:51-54 (“If the current subscriber state is quarantined, the DNS request is forwarded to the SACS DNS server” by “respond[ing] to the DNS request with the IP address of a SACS web server”), 8:24-27. SACS DNS can also be configured with a list of specific domain names that it is permitted to resolve, i.e., a list of specific domain names for which the SACS DNS server is permitted to return IP addresses. EX1009, 9:23-24. These domain names can include domain names associated with a remediation host, such as a host from which the device user can download software updates. EX1009, 9:26-28 (“the list may comprise the domain name of an enforcement server, a validation server, a virus scan server, and a service upgrade server”). A POSITA would have understood and found obvious to modify Freund’s router to prevent DNS queries from reaching the DNS server using the techniques taught by Danforth. §V(D) (motivation to combine). Such a modification would have involved the application of a known technique (interception of DNS requests from devices) to improve similar devices (Freund’s user devices) with a reasonable expectation of success. A POSITA would have further understood and found obvious to modify Freund’s router to include a “white list” of domain names that the SACS server is permitted to resolve, such as

a service upgrade server through which the user can download the anti-virus software update or the anti-virus software package linked in the sandbox server notification webpages, and if the DNS query is for a domain name not on the “white list,” to have the router respond with the IP address of the sandbox server. §V(D) (motivation to combine). Such a modification involves the application of a known technique (whitelisting access to permissible destinations in Danforth’s Internet-based system) to improve similar devices (Freund’s Internet-based system) with a reasonable expectation of success (preventing compromised devices from distributing unauthorized code to local or larger networks such as the Internet). EX1002, ¶276.

11.[1j]

Freund-Danforth combination discloses *permitting the first host* (computer) *to communicate with the remediation host* (server hosting anti-virus software update or the anti-virus software package). When Freund’s router determines the response is not compliant, the router permits users to communicate with a sandbox server that provides users with the ability to download a software update. EX1007, FIG. 3, ¶40, ¶89, ¶71. The Freund-Danforth combination discloses the ability of Freund’s router to whitelist other domains, in addition to the sandbox server, such as the remediation host. §V(E)(10) (limitation 1i). Accordingly, because the Freund-Danforth combination discloses whitelisting the remediation host, the

Freund-Danforth combination renders obvious permitting the non-compliant device to communicate with a server hosting an anti-virus software update or the antivirus software package. §V(E)(10)(limitation 1i); §V(D) (motivation to combine). EX1002, ¶277.

F. Claim 2

Freund discloses that a computer's response to a router challenge includes fields such as the antivirus product ID (determining whether a security software is installed) and whether antivirus real-time monitoring enforcement is selected (scanning for a vulnerability or for malicious data). EX1007, ¶132. A POSITA would have understood that the presence of malicious data can result in anomalous network traffic emanating from the infected device. A POSITA would have understood that the security software detecting anomalous network traffic outbound from the device is an indication of the presence of malware. EX1002, ¶¶278-280.

G. Claim 3

Freund renders obvious this limitation. As previously discussed, one example of the router detecting an insecure condition is if the computer responds to a router challenge with an indication it does not have the correct version of security software installed. EX1007, ¶71 ¶78, ¶85, ¶132. While Freund does not explicitly disclose that the security policy is related to the version of the operating system

executing on the computer, a POSITA would have understood and found obvious that Freund's technique for quarantining a computer responsive to determining that the computer is running an older version of the security software can be applied to determining that the computer's operating system version is not current with reasonable expectations of success. EX1002, ¶¶281-283.

H. Claim 4

Freund renders this claim obvious. Freund discloses that the functionality of the router can be installed "on a device that resides on the client's premises of LAN." EX1007, ¶69. Freund discloses that its router-side functionality can be installed on a host, such as computer 320. A POSITA would have understood that because the client management protocol 311, compliance table 312, and routing component 313 could exist on computer 320, computer 320's operating system can quarantine the device upon startup, such as in those situations where the computer has installed a new version of the operating system but has not verified the new version. EX1002, ¶¶284-286.

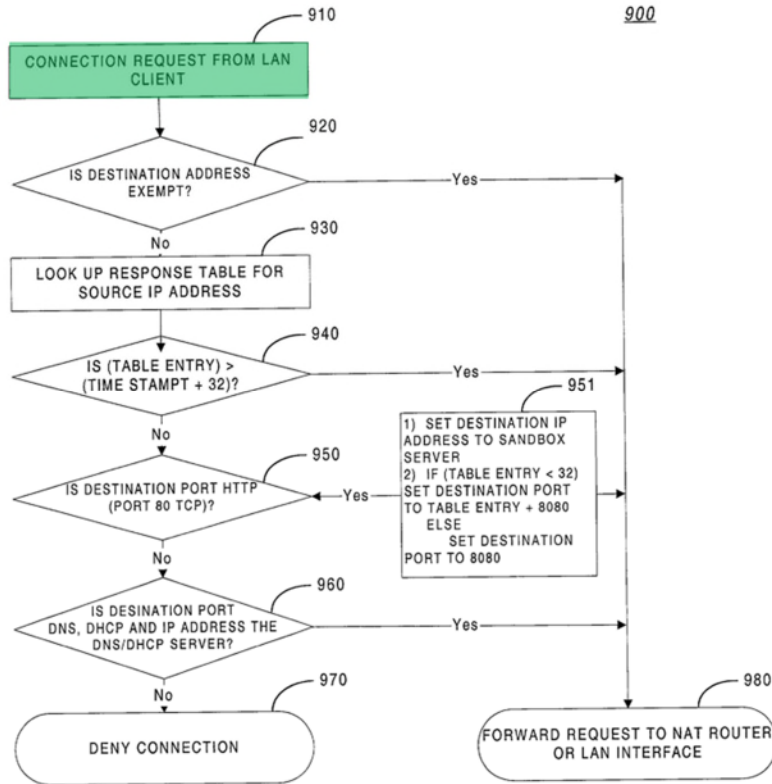
I. Claim 5

1. [5pre]

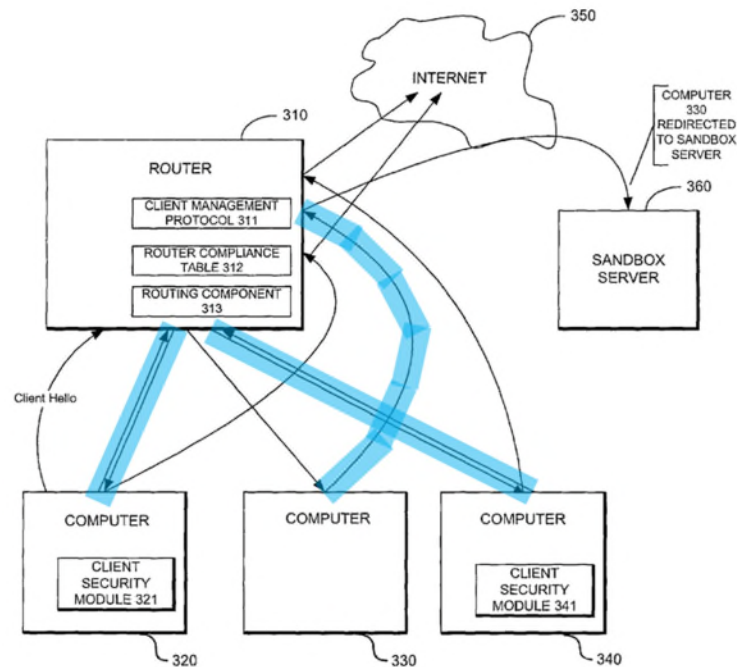
§V(E)(claim 1). EX1002, ¶¶287-288

2. [5a]

Freund discloses this limitation. Freund's router first receives an outbound communication in the form of a connection request from the first computer:



EX1007, FIG. 9 (block 910 (green)),



EX1007, FIG. 3 (illustrating service requests (blue)), ¶147 (“In step 910, a connection attempt from one of the local computers to the Internet is received by the router”), FIG. 9 (block 910). EX1002, ¶289.

3. [5b]

Freund-Danforth combination discloses this limitation. Danforth’s SACS DNS forwards outbound communications to the appropriate server if the outbound request is to one of a whitelisted set of specific domain names that it is permitted to resolve. EX1007, 9:23-24 (“SACS DNS server 120 comprises a list of domain names it is permitted to resolve”). This list can include the remediation host. EX1009, 9:26-28. EX1002, ¶290.

J. Claim 6

1. [6pre]

§V(E)(claim 1). EX1002, ¶¶291-292.

2. [6a]

§V(I)(2)(limitation 5a). EX1002, ¶293.

3. [6b]

Freund-Danforth combination discloses this limitation. Freund discloses redirecting requests from computers in the LAN to the sandbox server if the computer is not in compliance regardless of the type of request that is received. §V(E)(9) (limitation 1h). Freund-Danforth combination discloses exceptions to this redirect, such as if the outbound communication is addressed to a remediation host.

§V(E)(10)(limitation 1i); EX1007, 9:23-24, 9:26-28 (“the list may comprise the domain name of an enforcement server, a validation server, a virus scan server, and a service upgrade server”). Accordingly, in the case where the request is for an approved service that is not one of the whitelisted domains, such as the service upgrade server, Freund-Danforth combination discloses redirecting this communication to the sandbox server. EX1002, ¶294.

K. Claim 7

Freund-Danforth combination discloses this limitation. Freund discloses redirecting requests from computers in the LAN to the sandbox server if the computer is not in compliance. §V(E)(9)(limitation 1h). Freund-Danforth combination discloses this redirect if the outbound communication is not addressed to a remediation host, i.e., not addressed to one of the whitelisted domains. §V(E)(10)(limitation 1i); EX1007, 9:23-24, 9:26-28 (“the list may comprise the domain name of an enforcement server, a validation server, a virus scan server, and a service upgrade server”). Accordingly, because the Freund-Danforth combination discloses only being able to access the sandbox server or a remediation server if the computer is not determined to be in compliance, the Freund-Danforth combination necessarily discloses preventing the computer on the LAN from receiving data not related to remediation of the insecure condition. EX1002, ¶¶295-297.

L. Claim 8

Freund discloses this limitation. In particular, Freund discloses that its computers connect to the Internet through the router. EX1007, ¶74 (“The three personal computers 320, 330 and 340 connect to the Internet through router 310 via DSL or cable modem or another method of transport”), FIG. 3. A POSITA would have understood that the combination of the router and DSL or cable modem constitutes an Internet service provider because the combination of elements selectively provides access to the network, which Freund describes is the Internet. EX1002, ¶¶298-300.

M. Claim 9

§V(G)(claim 3). EX1002, ¶¶301-303.

As previously discussed, Freund’s router detects the presence or absence of security software installed on computers 320-340. EX1007, ¶75. Freund’s router also detects the current version or patch level associated with the security software. §V(F)(6)(limitation 1e). A POSITA would have understood that Freund’s techniques could easily be extended to software beyond security software, such as an operating system, with a reasonable expectation of success. Namely, such a modification of Freund employs a simple substitution of one application type (e.g., security software) having one set of properties (e.g., versions, patch level) with another application (e.g., operating system) having the same set of properties (e.g.,

versions, patch level). EX1020 (example security bulletin describing a security patch for Windows NT 4.0 operating system). EX1002, ¶304.

N. Claim 10

Freund discloses this limitation. As previously discussed, one example of the response not including a valid digitally signed attestation of cleanliness is determining that the computer does not have the correct version of security software installed. EX1007, ¶71 ¶78, ¶85, ¶132. EX1002, ¶¶305-307.

O. Claim 11

Freund discloses this limitation. As discussed, Freund's router issues an "anti-virus challenge" to the computer and receives a response to the challenge. EX1007, ¶132, ¶144. One of the fields in the challenge and response concerns whether the correct version of the antivirus program is installed. EX1007, ¶78 ("a computer running an older version of the security software may respond in the negative to a router challenge requesting confirmation that the computer is running a current version of the software"), ¶132, ¶144, ¶102. A POSITA would have understood that versions of a software product, such as an antivirus software, would include patches to address specific issues from previous version of the antivirus software. Therefore, determining whether to quarantine a computer based on a response that the antivirus software is not the correct version involves a

determination that the patch level associated with the software component on the first host is not sufficiently recent. EX1002, ¶¶308-310.

P. Claim 12

1. [12pre]

Freund discloses *a system for protecting a network*. EX1007, ¶4, ¶68, ¶¶147-151, FIG. 9. EX1002, ¶¶311-312.

2. [12a]

Freund discloses *a processor* that is configured to perform the steps of this claim. EX1007, ¶55:

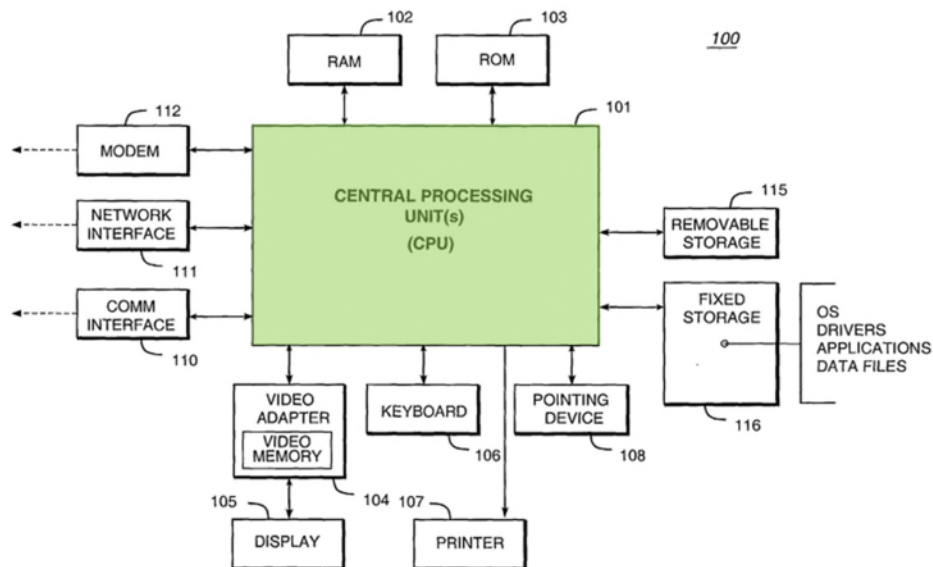


FIG. 1 (CPU 101 (green)), ¶¶55-56. The router's processor is configured to perform operations by executing program logic to implement the router-based policy enforcement techniques. EX1007, ¶58. To the extent it is argued that system 100 is not representative of router 310, a POSITA would have understood that

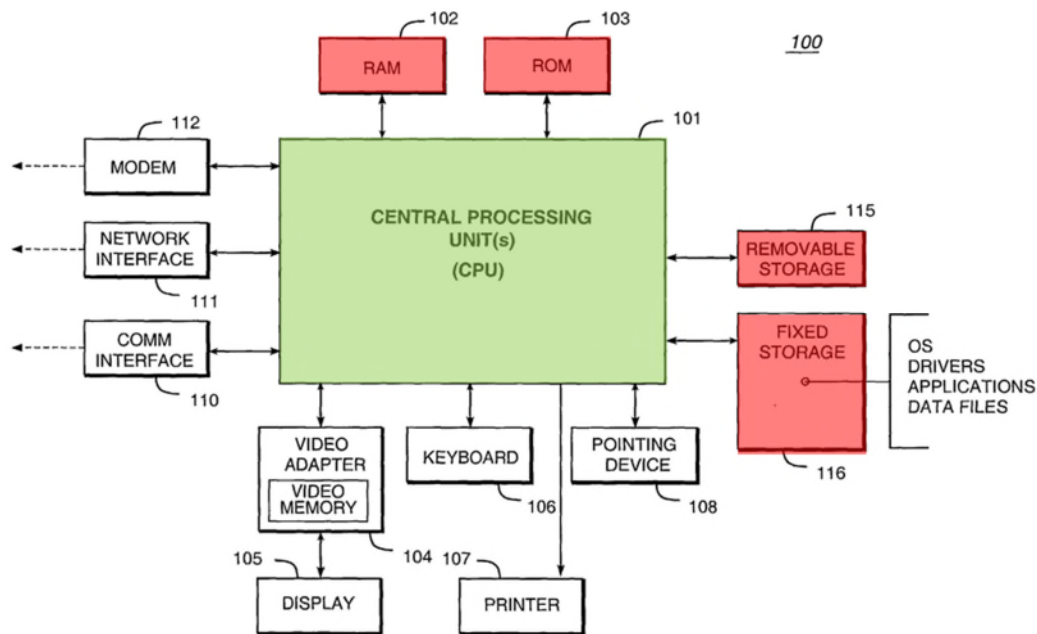
routers, such as router 310, include a processor configured to execute program logic loaded into main memory. *See, e.g.*, EX1021 (illustrating a generic router architecture that includes a routing processor). EX1002, ¶313.

3. [12b]-[12k]

§§V(E)(2)-V(E)(12)(limitation 1a-1j). EX1002, ¶¶314-323.

4. [12l]

Freund discloses *a memory* (fixed storage 115, 116/RAM 102/ROM (all red)) *coupled to the processor* (CPU 101 (green)) *and configured to provide instructions to the processor*:



EX1007, FIG. 1, ¶55, ¶58. To the extent it is argued that system 100 is not representative of router 310, a POSITA would have understood and found obvious

that routers, such as router 310, include a memory coupled to the processor and configured to provide instructions to the processor. EX1002, ¶324.

Q. Claim 13

§V(F)(claim 2). EX1002, ¶¶325-327.

R. Claim 14

§V(H)(claim 4). EX1002, ¶¶328-330.

S. Claim 15

§V(K)(claim 7). EX1002, ¶¶331-333.

T. Claim 16

§V(M)(claim 9). EX1002, ¶¶334-336.

U. Claim 17

§V(N)(claim 10). EX1002, ¶¶337-339.

V. Claim 18

§V(O)(claim 11). EX1002, ¶¶340-342.

W. Claim 19

Freund-Ball-Danforth combination renders this claim obvious. EX1002, ¶343.

1. [19pre]

Freund discloses this limitation. EX1007, ¶58 (“program logic...including that which implements methodology of the present invention...is loaded from the

storage device or mass storage 116 into...RAM...for execution by the CPU 101”).

EX1002, ¶344.

2. [19a]-[19j]

§§V(E)(2)-V(E)(11)(limitations 1a-1j). EX1002, ¶¶345-354.

For both Ground 1 and 2, there are no objective evidence of non-obviousness. EX1002, ¶355.

VI. MANDATORY NOTICES UNDER 37 C.F.R. §§ 42.8(B)(1)-(4)

A. Real Party-In-Interest

Petitioners identify Citrix Systems, Inc. and Cloud Software Group, Inc. as the real parties-in-interest.

B. Related Matters

The '705 patent is subject to the following actions:

- *Netskope, Inc. v. K.Mizra LLC*, IPR2025-01115 (PTAB, Aug. 13, 2025)
- *Netskope, Inc. v. K.Mizra LLC*, 3-25-cv-04833 (NDCA Jun. 6, 2025)
- *K.Mizra LLC v. Citrix Systems. Inc.*, 0-25-cv-60803 (SDFL Apr. 24, 2025)
- *K.Mizra LLC v. Google LLC*, 1-25-cv-00236 (WDTX Feb. 18, 2025)
- *K.Mizra LLC v. SonicWall Inc.*, 1-25-cv-00047 (DDE Jan. 10, 2025)

- *Cisco Systems, Inc. v. K.Mizra LLC*, 2022-2290, 2023-1183 (CAFC 2024)
- *Forescout Technologies, Inc. v. K.Mizra LLC*, IPR2022-00081 (PTAB Oct. 22, 2021)
- *Hewlett Packard Enterprise Co. v. K.Mizra LLC*, IPR2022-00084 (PTAB Oct. 22, 2021)
- *K.Mizra LLC v. Hewlett Packard Enterprise Co.*, 2-21-cv-00305 (EDTX Aug. 9, 2021)
- *K.Mizra LLC v. Forescout Technologies, Inc.*, 2-21-cv-00248 (EDTX Jul. 08, 2021)
- *K.Mizra LLC v. Fortinet, Inc.*, 2-21-cv-00249 (EDTX Jul. 08, 2021)
- *Cisco Systems, Inc. v K.Mizra LLC*, IPR2021-00593 (PTAB Mar. 15, 2021)
- *K.Mizra LLC v. Cisco Systems, Inc.*, 6-20-cv-01031 (WDTX Nov. 06, 2020)
- *Network Security Technologies, LLC v. Bradford Networks, Inc.*, 1-17-cv-01487 (DDE Oct. 24, 2017)
- *Network Security Technologies, LLC v. ForeScout Technologies, Inc.*, 1-17-cv-01488 (DDE Oct. 24, 2017)

- *Network Security Technologies, LLC v. McAfee, Inc.*, 1-17-cv-01489
(DDE Oct. 24, 2017)
- *Network Security Technologies, LLC v. Pulse Secure, LLC*, 1-17-cv-01490 (DDE Oct. 24, 2017)

C. Lead and Backup Counsel

Lead Counsel	Backup Counsel
James M. Heintz Reg. No. 41,828 Jim.Heintz@us.dlapiper.com DLA Piper LLP (US) One Fountain Square 11911 Freedom Drive Suite 300 Reston, Virginia 20190-5602 Phone: + 1 703 773 4000 Fax: + 1 703 773 5000	Safraz Ishmael (<i>pro hac vice</i> motion forthcoming) Safraz.Ishmael@us.dlapiper.com DLA Piper LLP (US) 33 Arch Street, 26th Floor Boston, Massachusetts 02110-1447 Phone: +1 617 406 6019 Fax: +1 617 406 6119 Michael G. Strapp (<i>pro hac vice</i> motion forthcoming) Michael.Strapp@us.dlapiper.com DLA Piper LLP (US) 33 Arch Street, 26th Floor Boston, Massachusetts 02110-1447 Phone: +1 617 406 6031 Fax: +1 617 406 6119 Joseph Wolfe Reg. No. 73,173 Joseph.Wolfe@us.dlapiper.com DLA Piper LLP (US) 1650 Market Street Suite 5000 Philadelphia, Pennsylvania 19103-7300 Phone: + 1 215 656 3359 Fax: + 1 215 606 2059

D. Service Information

Service information for lead and backup counsel is provided in the designation of lead and backup counsel above.

VII. CONCLUSION

All Challenged Claims of the '705 patent should be found unpatentable for the reasons discussed in this Petition.

Respectfully submitted,

/s/ James M. Heintz

James M. Heintz

Reg. No. 41,828

DLA Piper LLP US

One Fountain Square

11911 Freedom Drive, Suite 300

Reston, VA 20190-5602

Phone: + 1 703 773 4000

Fax: + 1 703 773 5000

Jim.heintz@us.dlapiper.com

Attorney for Petitioners

Attorney for Petitioners

CERTIFICATE OF WORD COUNT

Pursuant to 37 C.F.R. § 42.24(d), Petitioners certify that this petition includes 13,992 words, as measured by Microsoft Word, exclusive of the table of contents, mandatory notices under § 42.8, certificates of service, word count, claim listing, and exhibits.

Date: August 29, 2025

/s/ James M. Heintz

James M. Heintz

Reg. No. 41,828

DLA Piper LLP US

One Fountain Square

11911 Freedom Drive, Suite 300

Reston, VA 20190-5602

Phone: + 1 703 773 4000

Fax: + 1 703 773 5000

Jim.heintz@us.dlapiper.com

Attorney for Petitioners

CERTIFICATE OF SERVICE

The undersigned certifies pursuant to 37 C.F.R. §§ 42.6(e) and 42.105 that on August 29, 2025, a true and correct copy of the Petition for *Inter Partes* Review of U.S. Patent No. 8,234,705 was served by emailing a copy of same (by agreement in lieu of service under 37 C.F.R. § 42.105) to the following attorneys for the Patent Owner:

Robert R. Brunelli – rbrunelli@sheridanross.com

Bart Star – bstarr@sheridanross.com

Brian Boerman – bboerman@sheridanross.com

Lori Brown – lbrown@sheridanross.com

Respectfully submitted,

/s/ James M. Heintz

James M. Heintz

Reg. No. 41,828

DLA Piper LLP US

One Fountain Square

11911 Freedom Drive, Suite 300

Reston, VA 20190-5602

Phone: + 1 703 773 4000

Fax: + 1 703 773 5000

Jim.heintz@us.dlapiper.com