UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

NETSKOPE, INC.,

Petitioner,

v.

K. MIZRA LLC,

Patent Owner.

PTAB Case No. IPR2025-01115

Patent No. 8,234,705 B1

DECLARATION OF A.L. NARASIMHA REDDY, PH.D., IN SUPPORT OF PETITION FOR *INTER PARTES* REVIEW

TABLE OF CONTENTS

I. INTRODUCTION 1

A. QUALIFICATIONS AND PROFESSIONAL EXPERIENCE 3

B. LEVEL OF ORDINARY SKILL IN THE ART 6

C. RELEVANT LEGAL STANDARDS 7

D. SUMMARY OF OPINIONS 9

II. OVERVIEW OF THE TECHNOLOGY 9

A. Redirection 9

B. Domain Name System (DNS) 10

C. Device Quarantine 11

D. Trusted Platform Module 12

III. OVERVIEW OF THE '705 PATENT 15

IV. CLAIM CONSTRUCTION 16

A. "*trusted computing base*" 16

B. *"trusted platform module"* 19

C. Claim Construction in Related Matters 23

V. IDENTIFICATION OF HOW THE CLAIMS ARE UNPATENTABLE 24

VI. THE COMBINATION OF GLEICHAUF, OVADIA, AND LEWIS RENDERS OBVIOUS CLAIMS 1-3, 5-13, AND 15-19 25

A. Summary of Gleichauf 25

B. Summary of Ovadia 27

C. Summary of Lewis 28

D. Reasons to Combine Gleichauf with Ovadia 29

E. Reasons to Combine Gleichauf, Ovadia, and Lewis.....41

F. Detailed Analysis of Claims.....48

1. Claim 1 48

2. Claim 2 104

3. Claim 3 107

4. Claim 5 110

5. Claim 6 113

6. Claim 7 115

7. Claim 8 118

8. Claim 9 120

9. Claim 10 120

10. Claim 11 121

11. Claim 12 122

12. Claim 13 126

13. Claim 15 127

14. Claim 16 127

15. Claim 17 127

16. Claim 18 127

17. Claim 19 128

VII. CONCLUSION.....132

I, Narasimha Reddy, Ph.D., do hereby declare as follows:

I. INTRODUCTION

1. I have been retained by Petitioner Netskope, Inc. ("Petitioner") to investigate and opine on certain issues relating to United States Patent No. 8,234,705 ("the '705 patent") to Roskind *et al*.

2. I am being compensated for my work in this matter at my standard hourly rate. I am also being reimbursed for reasonable and customary expenses associated with my work and testimony in this proceeding. My compensation is not contingent on the outcome of this matter or the specifics of my testimony.

3. I have been asked to provide my opinions regarding whether the subject matter of claims 1-3, 5-13, and 15-19 ("the Challenged Claims") of the '705 patent would have been obvious to a person having ordinary skill in the art ("POSITA") at the time of the alleged invention, in light of the prior art. It is my opinion that the Challenged Claims would have been obvious to a POSITA.

4. In the preparation of this declaration, I have studied:

   a. the '705 patent, Ex. 1001;

   b. the prosecution history of the '705 patent ("'705 File History"), Ex. 1002;

   c. U.S. Patent No. 9,436,820 to Gleichauf *et al*. ("Gleichauf"), Ex. 1005;

   d. U.S. Patent No. 7,747,862 to Ovadia ("Ovadia"), Ex. 1006; and

e. U.S. Patent No. 7,533,407 to Lewis *et al*. ("Lewis"), Ex. 1007.

5. In forming the opinions expressed below, I have considered: the documents listed above; the relevant legal standards, including the standard for obviousness; and my own knowledge and experience based upon my work in the field of network communications and security as described below, as well as portions of the following additional materials:

a. U.S. Patent No. 7,568,107, Ex. 1008;

b. U.S. Patent Pub. No. 2004/0177276, Ex. 1009;

c. U.S. Patent Pub. No. 2006/0005009, Ex. 1010;

d. U.S. Patent No. 6,829,654, Ex. 1011;

e. TCG Specification Architecture Overview (Rev. 1.2 Apr. 28, 2004) (hereinafter "TCG Specification Architecture Overview"), Ex. 1012;

f. TPM Main – Part 1 Design Principles, Specification Version 1.2 (Rev. 62 Oct. 2, 2003) (hereinafter "TPM Main Specification"), Ex. 1013;

g. U.S. Patent Pub. No. 2003/0061494, Ex. 1014;

h. U.S. Patent Pub. No. 2005/0078668, Ex. 1015;

i. U.S. Patent No. 7,571,460, Ex. 1016;

j. U.S. Patent Pub. No. 2005/0097199, Ex. 1017;

k. TCPA Main Specification (Version 1.1b, 2003), Ex. 1018

l. U.S. Patent Pub. No. 2005/0033987, Ex. 1021; and

m. U.S. Patent No. 7,330,977, Ex. 1023.

6. I understand that the '705 patent has been assigned to K.Mizra LLC.

7. Unless otherwise noted, all emphasis in the text has been added. Claim terms are italicized.

A. QUALIFICATIONS AND PROFESSIONAL EXPERIENCE

8. My complete qualifications and professional experience are described in my *Curriculum Vitae*, a copy of which can be found in Exhibit 1004. The following is a brief summary of my relevant qualifications and professional experience.

9. I am currently the Truchard Foundation Chair Professor of Electrical and Computer Engineering at Texas A&M University in College Station, Texas. I have over 25 years of experience in a wide variety of technologies and industries relating to data communications, storage systems, and distributed systems, including packet- switched network communications.

10. My academic credentials include a Bachelor of Technology degree in Electronics and Electrical Communications Engineering from the Indian Institute of Technology, Kharagpur, India, in August 1985. I then received a Master of Science and a Ph.D. in Computer Engineering from the University of Illinois at Urbana-Champaign in May 1987 and August 1990, respectively.

11. I have worked for over 25 years in the field of Electrical Engineering. My primary focus and research interest has been on computer networks, storage systems, multimedia systems, and computer architecture. I have authored and co-authored over a hundred technical papers and several book chapters related to several of these interests, including on such topics as multipath routing, route control, high-speed networks, network congestion, packet management, quality of service regulation, network security, network modeling, differentiated services, storage system enhancements, caching strategies, and multimedia system enhancements to name a few examples. I am listed as an inventor on five U.S. patents in the field of multi-node communication networks.

12. My employment history following my graduation from the University of Illinois at Urbana-Champaign began at the IBM Almaden Research Center in San Jose, California in 1990. At IBM, I worked on projects related to disk arrays, multiprocessor communication, hierarchical storage systems and video servers.

13. In 1995, I joined the faculty of the department of Electrical Engineering at Texas A&M University initially as an Associate Professor and was later promoted to full, tenured Professor. At Texas A&M, during 2016 to 2021, I was the Associate Agency Director for Strategic Initiatives and Centers for the Texas A&M Engineering Experiment Station (TEES), which engages in engineering and

technology-oriented research and educational collaborations. Further, during 2016 to 2021, I served as Associate Dean for Research. Currently, I serve as the Head of the Department of Electrical and Computer Engineering.

14. At Texas A&M, I have taught dozens of courses related to computer networking and communications, as well as computer architecture, multimedia systems and networks, topics in networking security, multimedia storage and delivery, as well as networking for multimedia applications. I have also served on various committees for the benefit of the scientific community and the Texas A&M University community.

15. I am a member of a number of professional societies, including the Institute of Electrical and Electronic Engineers (IEEE), where I have been elevated to an IEEE Fellow, and the Association for Computing Machinery (ACM). I have been responsible for chairing or co-chairing numerous conferences and programs, as well as presenting research at major IEEE and ACM conferences. For example, I served as program co-chair for the 2008 5th International Conference on Broadband Communications, Networks and Systems, panels co-chair for the 2008 3rd International Conference on Communication Systems Software & Middleware, and panel chair of the IEEE Conference of High Performance Computer Architecture.

16. My presentations include a Keynote speech at International Conference on Information Technology-New Generations in 2013, a Keynote speech at IEEE International Symposium on Computers and Communications 2010, and several invited talks including at Georgia Tech (2013), COMSNETS Conference (2013), Int. Conf. on Networking and Communications (2012), Samsung (2011), Korea University (2011), Aijou University (2011), Catedra Series at University of Carlos III, Madrid (2009), Thomson Research, Paris (2009), Telefonica Research, Barcelona (2009), and a Distinguished Seminar at IBM Austin Research Lab (2008).

17. I have received multiple awards in the field of networks and computer architecture. I received the NSF Career Award from 1996-2000. I received an outstanding professor award by the IEEE student branch at Texas A&M during 1997- 1998, an outstanding faculty award by the department of Electrical and Computer Engineering during 2003-2004, a Distinguished Achievement award for teaching from the former students' association of Texas A&M University, and a citation "for one of the most influential papers from the 1st ACM Multimedia conference." I received the "Regents Professor" award at Texas A&M University in 2024.

B. LEVEL OF ORDINARY SKILL IN THE ART

18. I understand there are multiple factors relevant to determining the level of ordinary skill in the pertinent art, including (1) the levels of education and experience of persons working in the field at the time of the invention; (2) the sophistication of the technology; (3) the types of problems encountered in the field; and (4) the prior art solutions to those problems.

19. A POSITA in the field of the '705 patent, as of its priority date of September 27, 2004, would have been someone knowledgeable and familiar with network communications techniques and security methodologies available in the mid-2000s. Such a POSITA would have a bachelor's degree in computer science, computer engineering, electrical engineering, or equivalent training, and approximately two years of experience working in the field of network communications, or more particularly, network security. Additional work experience can substitute for specific educational background, and vice versa.

20. For purposes of this Declaration, in general, and unless otherwise noted, my statements and opinions, such as those regarding my own experience and what a POSITA would have understood or known generally (and specifically related to the references I consulted herein), reflect the knowledge that existed in the relevant field as of the priority date of the '705 patent.

C. RELEVANT LEGAL STANDARDS

21. I am not an attorney. In preparing and expressing my opinions and considering the subject matter of the '705 patent, I am relying on certain basic legal principles that Netskope's counsel has explained to me.

22. I understand that prior art to the '705 patent includes patents and printed publications in the relevant art that predate the priority date of the '705 patent. For purposes of this Declaration, I am applying September 27, 2004, as the priority date of the '705 patent.

23. I have been informed by Netskope's counsel that a claimed invention is unpatentable under 35 U.S.C. § 103 if the differences between the claimed invention and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a POSITA. I have also been informed by Netskope's counsel that the obviousness analysis considers factual inquiries, including the level of ordinary skill in the art, the scope and content of the prior art, and the differences between the prior art and the claimed subject matter.

24. I have been further informed by Netskope's counsel that there are several recognized rationales for combining references or modifying a reference to show obviousness. These rationales include: (a) combining prior art elements according to known methods to yield predictable results; (b) simple substitution of one known element for another to obtain predictable results; (c) use of a known

technique to improve a similar device (method, or product) in the same way; (d) applying a known technique to a known device (method, or product) ready for improvement to yield predictable results; (e) choosing from a finite number of identified, predictable solutions, with a reasonable expectation of success; and (f) some teaching, suggestion, or motivation in the prior art that would have led a POSITA to modify the prior art or to combine prior art teachings to arrive at the claimed invention.

D. SUMMARY OF OPINIONS

25. I have reviewed and analyzed the ’705 patent.

26. Based on my review and analysis, it is my opinion that claims 1-3, 5-13, 15-19 of the ’705 patent are invalid based on the following ground:

Ground	Basis	References	Claims
1	§ 103	Gleichauf and Ovadia and Lewis	1-3, 5-13, 15-19

II. OVERVIEW OF THE TECHNOLOGY

A. Redirection

27. Uniform resource locator (URL) redirection is a well-known concept in HTTP traffic. Typically, a client device sends a packet to a network requesting access to an Internet web page. If the network determines that the packet must be redirected, the network returns a redirect address (e.g., URL) to the client. The

browser on the client then accesses the web page indicated by the redirect address. *See, e.g.*, U.S. Patent Pub. No. 2005/0078668 at [0003]. URL redirection is commonly used to direct a client's browser to a login page like those commonly found when accessing the Internet from a public space such as a hotel or airport. *See, e.g.*, U.S. Patent No. 7,568,107 at 4:56-57 ("redirect the users to a Uniform Resource Locator (URL) address of a network login page"); U.S. Patent Pub. No. 2004/0177276 at [0044] ("control device 20 can redirect the HTTP request to an authentication web page hosted by the control device using any HTTP redirection technique known in the art").

B. Domain Name System (DNS)

28. The Domain Name System ("DNS") is the distributed "phone book" of the Internet, translating human-readable hostnames into the numerical IP addresses used by routers to deliver network traffic. The core DNS architecture was designed in the 1980s and was standardized in the seminal IETF specifications RFC 1034 and RFC 1035 (both published in November 1987). *See* RFC 1034. By the early 2000s, DNS was regarded as "an integral part of the Internet," allowing users to access information on computers by typing easily remembered text names (e.g., "coolsite.com") instead of numerical addresses (e.g., "199.227.249.232"). U.S. Patent No. 6,332,158 at 1:40-52.

29. A DNS query is typically the first step of any web request: when a user enters a URL, the networking stack cannot open the HTTP/TLS connection until the domain name is resolved. Because DNS sits in this critical path, designers quickly realized they could intercept or modify the reply for policy or traffic engineering purposes. Early commercial hotspots illustrate the point: for example, when a user in a hotel attempts to load any web page, the gateway redirects the browser to a login page until payment is completed. Similarly, DNS interception enabled network administrators to transparently direct unauthorized or non-compliant users to quarantine or remediation servers instead of allowing normal DNS resolution. *See, e.g.*, Lewis at 14:20-23 ("The DNS, contained in the DHCP OFFER message at option code 6, is set to the IP address of the hi-jack DNS of the QS. Whenever the client performs name resolution on any address, it will receive the IP of QS. This way any client will be redirected to fix-up page on QS.").

C. Device Quarantine

30. It was common practice at the time of the ''705 patent priority date in 2004 to quarantine a device that is attempting to access a network or network resources. Placing a device in quarantine simply means that the device may be able to access the network but will not be allowed to access specific network resources until it is out of quarantine. Gleichauf at 14:62-66. There are many reasons why a

device may be placed in quarantine. For example, device may be put into quarantine if its behavior is found to be anomalous. *See* U.S. Patent Pub. No. 2005/0097199 at [0039] ("For example, if the first network device 110 is a server that always services requests from other devices, **an anomaly occurs when the server begins making requests**. … The scanning action module 215 can then perform one or more of the actions described above or below (e.g., **quarantine** the first network device 110, …")) Another common reason to put a device into quarantine is if the device is infected with a virus or is found to have been hijacked by an unauthorized entity. U.S. Patent No. 7,571,460 at 4:8-14.

D. Trusted Platform Module

31. Network security includes the prevention of theft and/or unauthorized use of computing assets. There are multiple ways in which such malicious use can be prevented, one of which is the use of a Trusted Computing Group ("TCG")-defined Trusted Platform Module ("TPM"). The TCG was formed in the Spring of 2003 as a not-for-profit "industry-standards organization with the aim of enhancing the security of the computing environment in disparate computer platforms." Consistent with that goal, contemporaneous literature characterizes the TPM specification as an industry standard set of guidelines for implementing hardware-based trust across platforms. For example, U.S. Patent Pub. No. 2003/0138105 notes

that the Trusted Computing Platform Alliance "has established standards for embedding security functionality in computer systems," with the TPM serving as "the hardware instantiation" of those standards. U.S. Patent Pub. No. 2003/0138105 at [0006]–[0008], [0018]. Further illustrating its broad uptake, U.S. Patent Pub. No. 2004/0205353 describes a computer system "built on a trusted platform such as the TCPA industry standard platform," adding that such trusted platforms had already "emerged as a standard for raising the level of confidence" in network transactions. U.S. Patent Pub. No. 2004/0205353 at [0001]–[0002].

32. A TCG-defined TPM can be used, among other things, to provide asset tracking functionalities. For example, if an asset that includes a TPM is lost or stolen, the TPM can prevent the thief from gaining access to data stored on the asset, data outside of the asset, such as on a private network, or both. A TPM can also be used to collect and report, in a trusted manner, platform configuration information, such as the platform's hardware and/or software components. TCG Specification Architecture Overview at 2-4. Information provided by the TPM is "trusted" because this information cannot be altered, and it is commonly encrypted or otherwise protected from manipulation. *See* U.S. Patent Pub. No. 2003/0061494 at [0011] ("The TCPA architecture includes an isolated computing engine, or trusted platform module (TPM) (not shown), **whose processes can be trusted because they**

cannot be altered. … Additionally, the **TPM may encrypt or wrap data**, such as a key, and may bind or seal an internally generated asymmetric key pair to a particular TPM and/or a particular platform configuration."). This additional information can be used to further secure client-server communications by allowing communications to occur within the context of a previously established, secure relationship. TCG Specification Architecture Overview at 2-4.

33. The TCG had also published other documents in the early 2000's, such as the Trusted Computing Platform Alliance (TPCA) Main Specification, Version 1.1b (2003) (Ex. 1018), which provided an industry specification to enable trust in computing platforms. *See* Ex. 1018, 1. The document starts by highlighting benefits of using standardized security mechanism: "Standardization is necessary so that the security and cryptographic community can assess the mechanisms involved, and so that customers can understand and trust the effectiveness of new features." *Id*. The document further describes that to enable trust, "a Trusted Platform provides information to enable the entity to deduce the software environment in a Trusted Platform. That information is reliably measured and reported to the entity. At the same time, a Trusted Platform provides a means to encrypt cryptographic keys and to state the software environment that must be in place before the keys can be decrypted." *Id*., 2. Therefore, the TPCA document further corroborates that all of

the features and functionality described in this section, which also appear in the claims of the '705 patent, were well known by the mid-2000s, and a POSITA would have been familiar with them.

III. OVERVIEW OF THE '705 PATENT

34. The '705 patent attempts to defend a computer network against threats by analyzing the security state of a device requesting network access. '705 patent at Abstract. The '705 patent discloses that "a request is received" from a device, via a network interface, "to connect to a protected network. It is determined whether the [device] is required to be quarantined. If the [device] is to be quarantined, the [device] is provided only limited access to the protected network." '705 patent at 3:9-13. "[A] quarantined [device] is permitted to access the protected network only as required to remedy a condition that caused the quarantine to be imposed." '705 patent at 3:14-17. The quarantine server transmits a quarantine notification page (a web page) to the device. '705 patent at 15:66-16:2. An example of a quarantine notification page is a "web page that provides notification that the computer is quarantined, and/or provides links to remediation sites appropriate to the quarantine such as a link to a site that provides anti-contagion software for removing a virus that the quarantined computer is believed to contain." '705 patent at 16:3-7.

35. As I will explain below, these concepts were well known as of the priority date of the '705 patent.

IV. CLAIM CONSTRUCTION

36. I have been asked to provide my opinion on a number of claim terms.

37. It is my understanding that in order to properly evaluate the '705 patent, the terms of the claims must first be interpreted. It is my understanding that for the purposes of this *inter partes* review, the claims are to be construed under the so-called *Phillips* standard, under which claim terms are given their ordinary and customary meaning as would have been understood by a POSITA in light of the specification and prosecution history, unless the inventor has set forth a special meaning for a term. I have also been informed that claim terms only need to be construed to the extent necessary to resolve the obviousness inquiry. I have reviewed the entirety of the '705 patent, as well as its prosecution history. It is my opinion that for purposes of applying the prior art presented herein to evaluate the patentability of the Challenged Claims, the two terms analyzed below require requires express construction.

A. "*trusted computing base*"

38. Claims 1, 12, and 19 recite in part a "*trusted computing base."* For example, claim 1 recites *"an attestation that the* ***trusted computing base*** *has ascertained the presence of a patch or a patch level associated with a software component on the first host*." '705 patent at claim 1.

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

> A computer associated with an address identified in a list is queried for a cleanliness assertion (1302), for example by contacting a **trusted computing base** within a computer, and requesting an authenticated infestation scan by trusted software. An example of a **trusted computing base** within a computer is the Paladium security initiative under development by Microsoft and supported by Intel and American Micro Devices. Another example of a **trusted computing base** is described in various TCG specifications, such as the TCG Architecture Overview, published by the Trusted Computing Group. Trusted code bases may for example execute antivirus scans of the remainder of the computer, including untrusted portions of the disk and/or operating system. In some embodiments, trusted code bases may digitally sign assertions about the cleanliness (e.g. infestation status) and/or state of their computers. In some embodiments, the query for cleanliness (1302) may be responded to by anti-

> contagion software, such as antivirus software, with assertions about the currency of a scan, such as the last time a scan was performed, or a version associated with a current anti- contagion software or definition file in use, wherein a sufficiently updated software and/or scan may act as a cleanliness assertion. In some embodiments, an operating system may respond with information associated with its patch level, wherein a sufficiently recent patch level may be interpreted as an assertion of cleanliness.

'705 patent at 13:64-14:22.

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

> Applicant notes that "trusted computing base" and "trusted platform module["] are terms of art with specific meanings that are in no way discussed in Liang.…As for the term of art "trusted computing base," the final paragraph of the "Definition and characterization" section of the Wikipedia entry on "Trusted Computing Base" (http://en.wikipedia.org/wiki/Trusted_computing_base;

> retrieved January 10, 2010) states that "A given piece of hardware or software is a part of the TCB if and only if it has been designed to be a part of the mechanism that provides its security to the computer system."

'705 patent file history at pp. 183-184.

41. For the purposes of analyzing the prior art in this declaration, I will apply the express definition of "*trusted computing base*" provided by the applicants during prosecution. That is, I will interpret *"trusted computing base"* as *a piece of hardware or software that has been designed to be part of a mechanism that provides security to a computer system*.

B. *"trusted platform module"*

42. Claims 1, 12, and 19 recite in part a "*trusted platform module*." For example, claim 1 recites "*contacting a trusted computing base associated with a trusted platform module within the first host*." '705 patent at claim 1.

43. The specification does not use the term *trusted platform module*. During prosecution of the '705 patent, the applicants argued that *trusted platform module* is a term of art that specifically refers to a secure cryptoprocessor that, among other things, implements a standard of the same name promulgated by the TCG:

> Applicant notes that "trusted computing base" and "trusted

> platform module[”] are terms of art with specific meanings that are in no way discussed in Liang. The Wikipedia entry for “Trusted Platform Module (http://en.wikipedia.org/wiki/Trusted platform module; retrieved January 10, 2010) begins, “In computing, Trusted Platform Module (TPM) is both the name of a published specification detailing a secure cryptoprocessor that can store cryptographic keys that protect information, as well as the general name of implementations of that specification, often called the “TPM chip” or “TPM Security Device” (as designated in certain Dell BIOS settings[1]). The TPM specification is the work of the Trusted Computing Group. The current version of the TPM specification is 1.2 Revision 103, published on July 9, 2007.” Reinforcing the industry adoption of this technology and standardization around the term of art used in the claims, this specification has also been adopted as ISO/IEC standard 11889.

’705 patent file history at pp. 183-184.

44. Consistent with the applicants’ arguments, the trusted platform module specification promulgated by the TCG describes a TPM that includes various components directed toward performing cryptographic operations and storing

cryptographic information. Major components of a TPM are depicted in the figures reproduced below:

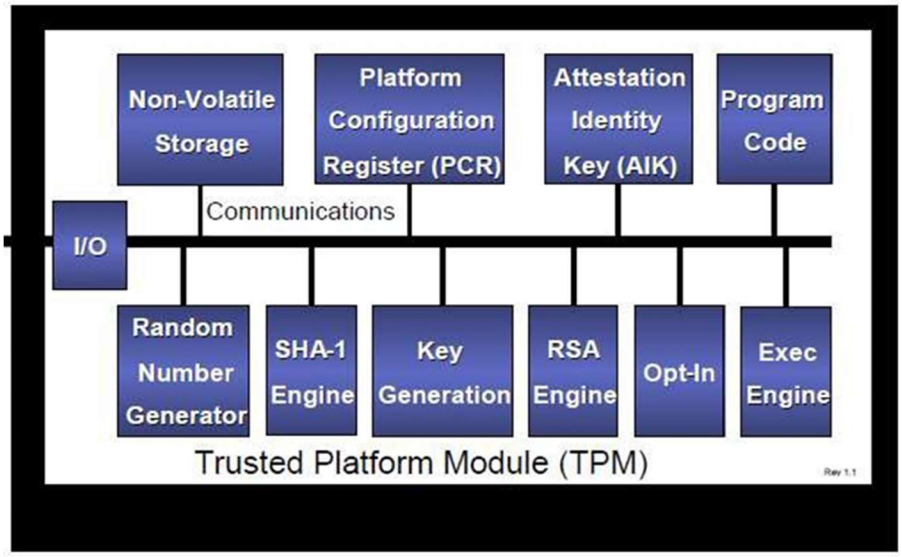
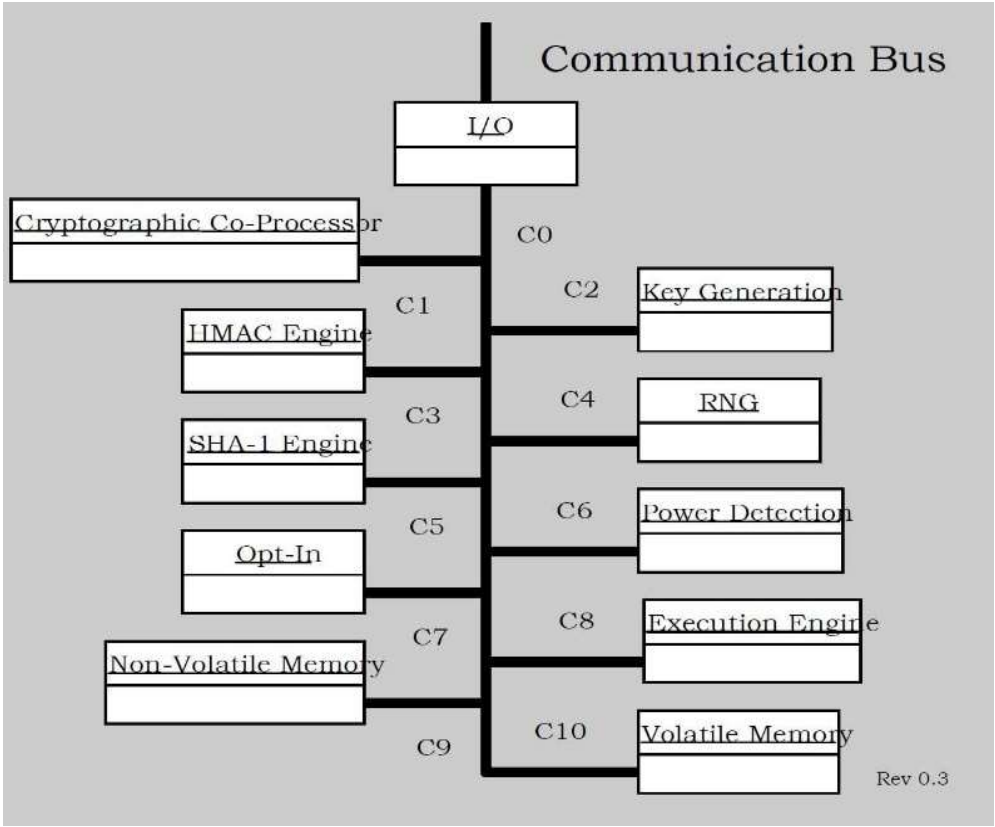


Figure 4:g – TPM Component Architecture

TCG Specification Architecture Overview at 19.



TPM Main Specification at 11.

45. Per the trusted platform module specification, the components of a TPM include an input/output (I/O) component, a key generation component, a random number generator (RNG), a SHA-1 engine, an opt-in component, an execution engine, and memory (volatile and non-volatile). TPM Main Specification at 11-22; TCG Specification Architecture Overview at 19-20. Consistent with the applicants' statement that a TPM "store[s] cryptographic keys that protect information," the trusted platform module specification provides that a compliant TPM stores cryptographic keys in memory. See TCG Specification Architecture Overview at 19 ("Non-volatile storage is used to store Endorsement Key (EK),

Storage Root Key (SRK), owner authorization data and persistent flags."); *see also* TPM Main Specification at 19 ("Non-volatile memory component … is used to store persistent identity and state associated with the TPM.")

46. For the purposes of analyzing the prior art in this declaration, I will apply the express definition of "*trusted platform module*" provided by the applicants during prosecution. That is, I will interpret "*trusted platform module"* as *a secure cryptoprocessor that implements the Trusted Platform Module specification from the Trusted Computing Group*.

C. Claim Construction in Related Matters

47. I understand that the terms "*trusted computing base*" and "*trusted platform module*" have already been construed in earlier proceedings involving the '705 patent. In *K.Mizra LLC v. Cisco Systems, Inc.*, No. 6:20-cv-01031 (W.D. Tex.), the parties stipulated that *"trusted computing base"* means "hardware or software that has been designed to be a part of the mechanism that provides security to a computer." *K.Mizra LLC v. Cisco Systems, Inc.*, No. 6:20-cv-01031, Dkt. No. 35, at 1 (W.D. Tex.). As to *"trusted platform module,"* although the parties initially proposed different constructions, the district court adopted the construction "a secure cryptoprocessor that can store cryptographic keys and that implements the Trusted

Platform Module specification from the Trusted Computing Group." *Id.*, Dkt. No. 46, at 1.

48. In *inter partes* review (IPR2021-00593), whichwas filed as a companion to the prior District Court litigation, the petitioner advocated that "trusted computing base" 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," relying on prosecution history statements. IPR2021-00593, Petition at 12-13. The patent owner did not dispute that construction. IPR2021-00593, Patent Owner Response at 5. The patent owner did not dispute that construction. I understand the parties did disagree over "trusted platform module," but the Board concluded that the dispute would not affect the outcome, observing that the parties did "not identify any issue that would turn on the construction of this term" and therefore determining that "no claim terms required express construction." IPR2021-00593, Final Written Decision at 14.

V. IDENTIFICATION OF HOW THE CLAIMS ARE UNPATENTABLE

49. The discussion in this Declaration provides a detailed analysis of how the asserted prior art references teach each limitation of the Challenged Claims.

50. As part of my analysis, I have considered, and discuss in detail, the scope and content of the prior art and any differences between the alleged invention and the prior art.

51. It is my opinion that the alleged invention recited in the Challenged Claims would have been obvious in view of the teachings of the asserted prior art and the knowledge of a POSITA.

Ground	Basis	References	Claims
1	§ 103	Gleichauf and Ovadia and Lewis	1-3, 5-13, 15-19

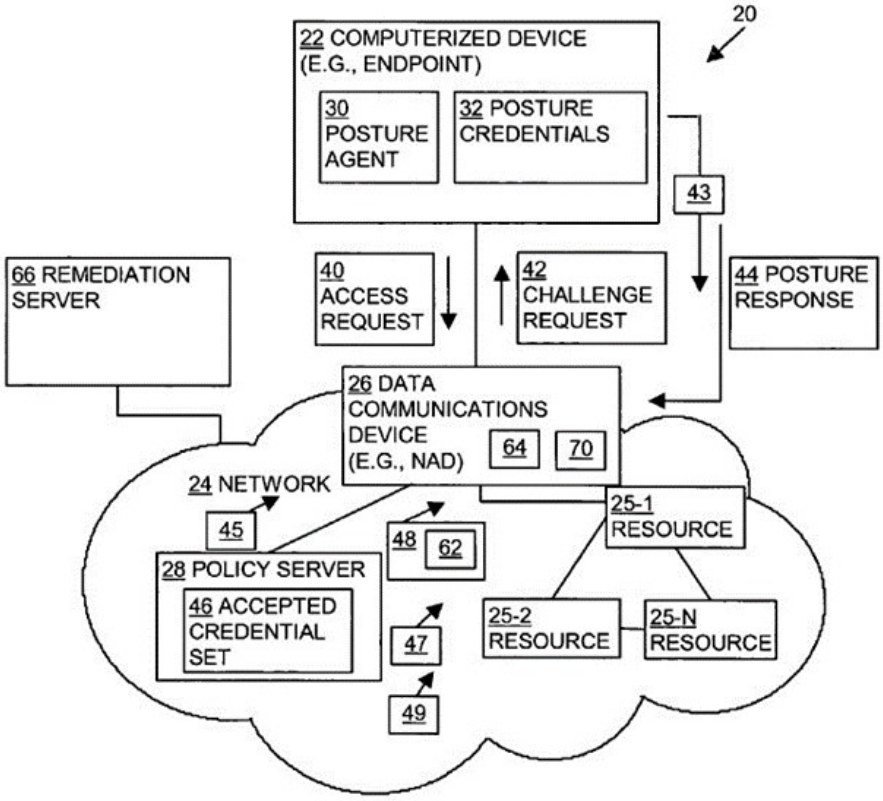
VI. THE COMBINATION OF GLEICHAUF, OVADIA, AND LEWIS RENDERS OBVIOUS CLAIMS 1-3, 5-13, AND 15-19

A. Summary of Gleichauf

52. U.S. Patent No. 9,436,820 to Gleichauf et al. (Ex. 1005, "Gleichauf") was filed August 2, 2004, and issued September 6, 2016. Gleichauf is titled "Controlling Access to Resources in a Network." Gleichauf was not before the Examiner during prosecution of the '705 patent.

53. Like the '705 patent, Gleichauf recognized that "conventional techniques for admitting computerized devices onto a network suffer from a variety of deficiencies that involve network security issues." Gleichauf at 2:46-48; see '705 patent at 1:38-40. Gleichauf describes a network security methodology that involves controlling access of a computerized device to a network based on the security posture of the device when it attempts to connect to that network. Gleichauf at 3:4-9. Gleichauf depicts, in Figure 1, a data communications system that includes a

computerized device 22 that is configured to attempt access to a network resource 25 on protected network 24:



Gleichauf at Fig. 1.

54. A policy server 28 controls access of the device to the network. Gleichauf at 10:20-23. The policy server "may determine the appropriate admission rules to be downloaded to the data communications device including allowing access to restricted resources, quarantining the computerized device, or denying access completely." Gleichauf at 7:4-8.

55. A computerized device is quarantined when the policy server detects a security issue with the device, such as an out-of-date anti-virus application.

Gleichauf at 15:55-61. While in quarantine, the device has limited network access but is still allowed to access a remediation server 66 that contains security updates that can bring the device into compliance with network access security protocols. Gleichauf at 16:13-21. After the device is updated, it is granted normal access to the protected network. Gleichauf at 16:40-43.

B. Summary of Ovadia

56. U.S. Patent No. 7,747,862 to Ovadia (Ex. 1006, "Ovadia") was filed on June 28, 2004, and issued on June 29, 2010. Ovadia is titled "Method and Apparatus to Authenticate Base and Subscriber Stations and Secure Sessions for Broadband Wireless Networks." Ovadia was not before the Examiner during prosecution of the '705 patent.

57. Ovadia discloses security measures for networks and describes a procedure that can ensure that a device seeking network access has an authorized configuration. Ovadia at 1:8-11, Abstract. The disclosed security scheme uses a trusted platform module (TPM) to generate security keys, including attestation identity keys ("AIKs"). Ovadia at Abstract. A subscriber station ("SS") seeking access to the network "sends authentication information including a manifest signed with the SS's private AIK key." Ovadia at Abstract. An authentication server receives the information and authenticates the SS via the use of the SS's public AIK

key. Ovadia at Abstract. The disclosed TPM is also used "to verify [that] a current configuration of a subscriber station platform is an authorized configuration." Ovadia at Abstract.

C. Summary of Lewis

58. U.S. Patent No. 7,533,407 to Lewis et al. (Ex. 1007, "Lewis"), was filed April 14, 2004, and issued May 12, 2009. Lewis is titled "System and Methods for Providing Network Quarantine."

59. Lewis discloses "a system for ensuring that machines having invalid or corrupt states are restricted from accessing network resources." Lewis at 4:7-9. Lewis describes how a device must provide a Bill of Health ("BoH") to a Quarantine Server ("QS") before gaining access to the protected network. Lewis at 4:13-15. The QS determines if the BoH is valid or invalid. Lewis at 4:16-23. A valid BoH indicates that the device has all necessary up-to-date security software, so the device is granted access to the network. Lewis at 4:18-21. An invalid BoH indicates that the device lacks the required security measures to access the network, so the device's network configuration is put in a lock-down configuration. Lewis at 13:44-52.

60. The lock-down configuration contains system settings that put a device in quarantine. Lewis at 13:63-65. A device in quarantine is permitted only limited communications, such as with the QS which provides a web page to the quarantined

device informing the "user that the client is in quarantine and corrective action must be taken." Lewis at 10:61-66. The user must perform the corrective action and submit an updated BoH to gain access to the network. Lewis at 13:12-15.

61. During the examination of the '705 patent, Lewis's corresponding application publication, U.S. Patent Pub. No. 2005/0131997, was cited by the Examiner for its teaching of a quarantine server providing a quarantine notification page. '705 patent file history (Ex.1002) at pp. 98-99. The applicants did not dispute this, arguing instead that the Examiner's cited art (including Lewis) failed to render obvious other features recited in then-pending claim 1. '705 patent file history (Ex.1002) at pp. 73-74.

D. Reasons to Combine Gleichauf with Ovadia

62. A POSITA would have combined Gleichauf with Ovadia because they both relate to providing secure access when an individual computerized device connects to a network. Gleichauf at 3:4-9 ("By contrast, embodiments of the present invention … provide mechanisms that include 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."); Ovadia at 1:8-11 ("The field of invention relates generally to security schemes for

wireless networks, and, more specifically but not exclusively relates to techniques for enhancing security for broad band wireless networks.")

63. A POSITA would have been motivated to combine Gleichauf with Ovadia and doing so would have been within the POSITA's skillset. The combination would have been obvious because it represents merely the use of a known technique (Ovadia's TPM security infrastructure and digital signatures, which were industry-standard security techniques) to improve a similar method (Gleichauf's authentication and authorization of computerized devices seeking network access) in the same way.

64. Notably, Ovadia's teachings of (1) using a TPM, and (2) digitally signing an attestation of cleanliness both strengthen the communication path between Gleichauf's electronic device 22 and data-communications device 26, thereby reducing exposure to malicious attacks, viruses, worms, and other malware. These safeguards advance the '705 patent's stated objective of "limiting vulnerable computerized devices from accessing the network resources and minimizing the risk that the network resources receive or transmit malware." '705 patent at Abstract.

65. A POSITA would have been motivated to take advantage of Ovadia's use of a TPM—as described in the TCG "**platform-independent industry specification**"—for multiple reasons, including because it would standardize

Gleichauf's posture credentials. Ovadia at 4:32-35. The TCG was formed in the Spring of 2003 as an "industry- standards organization with the aim of enhancing the security of the computing environment in disparate computer platforms." TCG Specification Architecture Overview at 2. A POSITA would have known the benefits of using a standardized security platform in implementing or augmenting the protocols of Gleichauf for several reasons.

66. First, a standardized security platform undergoes extensive expert testing before achieving industry-standard status, making it less susceptible to design flaws or hidden backdoors. For example, such standardized platforms are rigorously reviewed and re-reviewed by experts in standards organizations, are subjected to threat modeling analysis and are typically tested against a wide-ranging suite of known attacks. *See* Ex. 1018 at 1. Moreover, their effectiveness is routinely monitored, and patches or updates are issued if new security concerns are discovered. *See* Common Criteria for Information Technology Security Evaluation, Version 2.2 (ISO/IEC 15408), Part 3. A POSITA would have found that integrating such an industry standardized, vetted TPM into Gleichauf therefore reduces the security vulnerabilities inherent in non-standardized platform solutions.

67. Second, Ovadia, citing the TCG specification, describes the TPM as "a trusted platform subsystem that employs cryptographic methods when establishing

trust." Ovadia at 4:36-38. A POSITA would recognize that such a standardized TPM enhances the trustworthiness of information stored on—and messages exchanged between—Gleichauf's electronic device and its data communications device. As a specific example, the TPM's tamper-detection safeguards, both physical markers and software-implemented features, illustrate this point: "If the TPM discovers tampering with the tick count (through timing changes etc.) the TPM MUST treat this as an attack and shut down further TPM processing as if a self-test had failed," and "[a]ny tamper-resistance markers… are under programmable control." TPM Main at 97, 45. These features allow identification of tampered or compromised devices and thereby advance Gleichauf's objective of "limiting vulnerable computerized devices from accessing the network resources and minimizing the risk that the network resources receive or transmit malware." Gleichauf at Abstract, 12:7-12, 12:27-31.

68. Third, a TCG-defined TPM can be used to provide additional features and functionalities that include asset tracking functionalities, collect and report (in a trusted manner) platform configuration information, and secure client-server communications. TCG Specification Architecture Overview at 2-4 (Rev. 1.2 Apr. 28, 2004). Information provided by the TPM is "trusted" because this information cannot be altered, and it is commonly encrypted or otherwise protected from

manipulation by others. *See* U.S. Patent Pub. No. 2003/0061494 at [0011] ("The TCPA architecture includes an isolated computing engine, or trusted platform module (TPM) (not shown), **whose processes can be trusted because they cannot be altered**. … Additionally, the **TPM may encrypt or wrap data**, such as a key, and may bind or seal an internally generated asymmetric key pair to a particular TPM and/or a particular platform configuration."); Cromer at 2:8-11 ("Trusted Computing Platform Alliance (TCPA)" is now "currently referred to as the Trusted Computing Group (TCG)".); Yan at [0003]-[0004] (same). In this way, a device's posture credentials would be formatted at the device consistently and contain substantially the same information, regardless of the device manufacturer or components operating on the device. It would also allow Gleichauf's policy server to process posture credentials that were received, in a trusted manner from a device, more efficiently and in substantially the same way, again regardless of the device returning the credentials.

69. Fourth, a POSITA would have appreciated the value of following an industry specification. It would promote interoperability among devices from different device manufacturers, so long as the devices are all specification compliant. Indeed, device interoperability is a common and important goal in computer networking. For example, Gleichauf provides examples of its computerized devices

"as personal computers or laptop computers connect to computer networks such as the Internet to transmit and receive content or data with other computerized devices in communication with the network." Ex.1005, 1:6-10. Because such personal computers or laptop computers were made by a variety of manufactures and companies, implementation of security features using a standardized trusted computing module would have provided interoperability across myriad devices.

70. Moreover, Ovadia describes applying technology from the "Trusted Computing Group (TCG) security scheme (promulgated by the Trusted Computing Group…)," which addresses "network security" and "covers trust in computing platforms in general." Ovadia at 4:16-35. A POSITA would thus have understood that the trusted computing and authentication technologies described in Ovadia are not limited to the IEEE P802.16-based networks that Ovadia describes, but instead are generally applicable to networks—like that in Gleichauf—that require authentication and authorization for access.

71. Fifth, the combination also would have been obvious because it is merely the application of a known technique (digitally signing an attestation about the state of a device to make its authentication more secure, per Ovadia) to a known method (network authorization process that analyzes a device's posture credentials that contain critical security information about the device, including its

authentication, per Gleichauf) ready for improvement to yield predictable results. Here, the improvement provided to Gleichauf's process by Ovadia's digital signature technique is resistance to forged or tampered-with posture credentials. A POSITA would have been familiar with the use of digital signatures to prove the integrity and authenticity of messages or other information communicated over a network. *See, e.g.*, U.S. Patent Pub. No. 2006/0005009 at [0033] ("**To ensure that the quoted value is not corrupted during communication**, TPM 50 also can generate an attestation identity key (AIK) that is used to encrypt some or all of the quoted value. Attribute system 40 can communicate the encrypted quoted value and AIK to assurance system 42."). In fact, at the time of the invention, it was well-known by POSITAs that digital signatures were routinely used to verify the authenticity of messages or documents. *See, e.g.*, U.S. Patent No. 6,745,327 at 1:22-37 ("The dominant technology under prior art for individuals to sign electronic documents and transaction data is based upon client-side digital signatures. The signatures are created by software that uses an encryption algorithm, called a private key, of the user to electronically encode the electronic document or transaction data. A mathematically related algorithm, called the public key corresponds to the private signature key. The public key is used by the recipient to verify the authenticity of the electronic document or data and the integrity of the data since signing occurred,

including the fact that it has not been changed or altered since the signature was affixed."). Gleichauf's process is also improved because the use of a digital signature allows the policy server to authenticate that the provided posture credentials originated with the device attempting to gain access to the network. *See* Ovadia at 16:30-32 ("AS authenticates the SS by performing a signature check using the SS's AIK public key…").

72. Sixth, a POSITA also would have been motivated to combine the teachings of Gleichauf and Ovadia because Ovadia provides additional details regarding implementation of Gleichauf's disclosed techniques. For example, Gleichauf explains that "policy server 28 authenticates an identity of the computerized device 22 to generate an authentication result." Gleichauf at 22:3-5. Ovadia explains that one way to authenticate a device is to check a digital signature produced by the device. Ovadia at 16:30-32. Thus, it would have been obvious to a POSITA to employ Ovadia's digital signature-checking technique as a way of performing Gleichauf's device-authentication step.

73. A POSITA would have also had a reasonable expectation of success in augmenting and improving Gleichauf's system based on Ovadia's teachings relating to the use of the TPM and digital signatures. Notably, Ovadia already contemplated the inclusion of the TPM into an existing platform, such as Gleichauf's system, by

explaining "[t]he trusted platform … may be integrated into some existing platform component or components." Ovadia, 4:38-40.

74. Furthermore, components of Ovadia's trusted computing architecture have corresponding components in Gleichauf's architecture, so a POSITA would have recognized how details provided for components in Ovadia would be applied to the corresponding components in Gleichauf.

75. Like the posture agent in Gleichauf, which is used to gather information about the device's "configuration state," Ovadia describes using "an authentication agent to determine the state of a platform environment" and generate an "integrity measurement." Gleichauf at 3:8, 12:21-31; Ovadia at 4:40-43, 14:1-6, Fig. 8. Ovadia's integrity measurement is stored in the TPM and then provided to an authentication server as part of authentication data in response to an authentication challenge. Ovadia at 4:43-45, 13:61-63, Fig. 8. This is analogous to Gleichauf's description of sending the configuration state of a device (or "posture credentials") to a policy server in response to "one or more requests 49 to authenticate the computerized device" seeking network access. Gleichauf at 10:63-11:3. Ovadia's integrity measurement is "digitally signed with appropriate keys" before it is sent to the authentication server. Ovadia at 13:63-64, Fig. 8. This process is depicted in Ovadia's Figure 8:

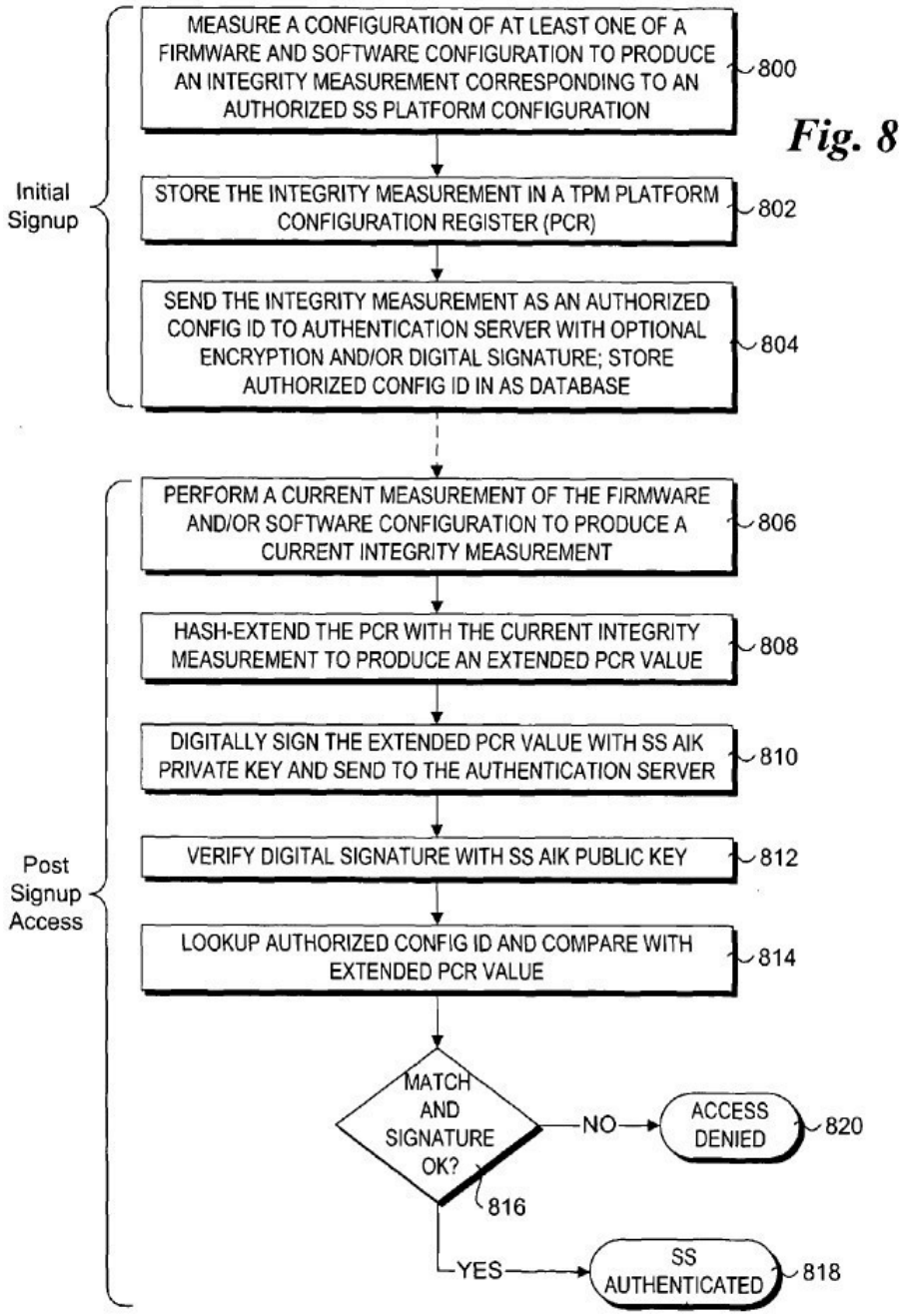


Fig. 8

Ovadia at Fig. 8.

76. A POSITA would have understood that Ovadia's TPM generates and sends the integrity measurement to the authentication server because the integrity measurement is digitally signed with an Attestation Integrity Key (AIK) and "AIKs

are only permitted to sign data generated by a TPM." Ovadia at 14:9, 5:35-36. It would thus have been obvious to a POSITA to implement Gleichauf's posture agent with TPM functionality. This functionality would include the ability to store, send, and digitally sign integrity measurements, as taught in Ovadia, to provide the greater security assurances available with digital signatures and limited-use integrity keys.

77. Gleichauf and Ovadia also both describe providing an explanatory message if the posture credentials or integrity measurement, respectively, are not accepted. Gleichauf at 4:50-67 ("notifications may include… an information message to be displayed to the user"); Ovadia at 16:62-65 ("rejection message… includes an error code and a display string containing an error message"). Therefore, Ovadia not only contemplated using the TPM in existing systems or components (Ovadia at 4:38-40), the analogous architectures of Gleichauf's and Ovadia's systems, together with the fact that they both solve similar problems within the same field of network access authentication, would have provided a POSITA with a reasonable expectation of success in making the combination. I also note that both Gleichauf and Ovadia acknowledge that a POSITA would have had the necessary skill set to make "various changes in form and details" of their described embodiments. Gleichauf at 27:24-29; see Ovadia at 18:64-19:4.

78. The combination permits but does not require physical incorporation of elements from Ovadia into Gleichauf. The TPM Main specification itself confirms that it offers design guidelines and generic information that a POSITA can use to tailor a given platform. TPM Main at 12 ("The design principles give the basic concepts of the TPM and generic information relative to TPM functionality. A TPM designer MUST review and implement the information in the TPM Main specification (parts 1-4) and review the platform specific document for the intended platform. The platform specific document will contain normative statements that affect the design and implementation of a TPM."). Thus, it is my opinion that a POSITA would have been able to readily use the TPM standard to effectuate a specific implementation that augmented and improved Gleichauf's system.

79. Notably, Ovadia teaches that a TPM can be implemented as a "hardware device (most common) or via software." Ovadia at 4:24-26. Gleichauf's posture agent can similarly be embodied "as a software or hardware application." Gleichauf at 9:29-39. Since Ovadia's TPM and Gleichauf's posture agent are both involved in the same process, it would have been obvious to pair Ovadia's TPM with Gleichauf's posture agent.

80. Furthermore, Ovadia and Gleichauf described systems that were compatible, and a POSITA would have understood that Gleichauf's system can

benefit from implementation of the trusted platform module and the use of digital signatures that Ovadia described. In other words, the two systems were not incompatible or conflicting; Ovadia described "methods and apparatus to authenticate base and subscriber stations and secure sessions for broadband wireless networks … [that] employs a trusted platform module (TPM) to generate security keys." Ovadia, Abstract. Ovadia described establishing a broadband connection, using an authentication protocol that involved sending subscriber authentication information and authenticating it via a digital signature, and once authentication was complete, a secure communication channel was established. See e.g., Ovadia at claim 1, Figures 3a-4b. Ovadia used a trusted platform module, and used a digital signature to ensure authenticity of the exchanged information. Gleichauf also described related operations, such as explaining that its "policy server 28 authenticate[d] an identity of the computerized device 22 to generate an authentication result." Gleichauf at 22:3-5. Therefore, a POSITA would have understood that above features of Ovadia were complementary to and enhanced the operations of Gleichauf's system that also relied on exchange of information between the policy server and the computing device that requested services.

E. Reasons to Combine Gleichauf, Ovadia, and Lewis

81. A POSITA would have combined Gleichauf, Ovadia, and Lewis because they all relate to providing secure access for an individual computing device connecting to a network. Gleichauf at 3:4-9 ("By contrast, embodiments of the present invention … provide mechanisms that include 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."); Ovadia at 1:8-11 ("The field of invention relates generally to security schemes for wireless networks, and, more specifically but not exclusively relates to techniques for enhancing security for broad band wireless networks."); Lewis at 4:7- 9 ("In view of the foregoing, the present invention provides a system for ensuring that machines having invalid or corrupt states are restricted from accessing network resources."). A POSITA would have been motivated to combine Gleichauf and Ovadia with Lewis and doing so would have been within the POSITA's skillset. The combination would have been obvious because it represents merely the use of a known technique (Lewis's redirection of device traffic comprising a DNS query to a quarantine server that serves a web page to the device) to improve a similar method (Gleichauf's traffic redirection and separate notification message to the device) in the same way.

82. Notably, while DNS queries were ubiquitous by 2004 (*see, e.g.*, U.S. Patent No. 6,332,158 at 1:40-52 ("(DNS) is an integral part of the Internet.")), Gleichauf does not specify the specific steps to take when a request comprising a DNS query is received from a non-compliant electronic device. Lewis supplies the missing detail: the query is resolved to the IP address of the quarantine server. Because DNS resolution translates a domain name into an IP address, a POSITA would recognize that—when the queried name from a non-compliant electronic device is not a remediation host—the obvious response is to return the quarantine server's IP address, as Lewis explains. The combined teachings of Gleichauf, Ovadia, and Lewis were obvious for several specific reasons.

83. First, DNS queries were routine; a POSITA would expect to encounter them and would benefit from Lewis's explicit guidance on handling such queries without compromising the security of the Gleichauf-Ovadia system.

84. Second, redirecting the DNS query to the quarantine server (part of the policy server in the Gleichauf-Ovadia-Lewis combination, as further explained below) is one of only a few logical options when receiving a request from a compromised electronic device. For instance, while as an alternative option, the query could instead resolve the DNS query to the IP address of the remediation server, redirecting it first to the quarantine server allows the quarantine server

(policy server) to reassess the electronic device's trustworthiness before deciding on the appropriate course of remediation. For example, a redirection to the quarantine server (policy server) would allow the policy server to determine if the electronic device were infected with additional viruses before providing instructions to access a particular remediation server appropriate for remediating the detected vulnerabilities. In fact, Gleichauf describes that its policy server is able "to re-assess the posture credentials and apply a new access policy on the data communications device," specifying that "the device will be again identified as being at some level of non-compliance and the appropriate response will be invoked by the policy server." Gleichauf at 5:60-67, 5:13-16.

85. Third, a POSITA would have found the combination with Lewis advantageous because Lewis describes serving a *webpage* with remediation instructions to a quarantined device.

86. Redirection to the quarantine server is triggered when the user opens a browser and attempts to access network resources. Lewis specifies, for example, "[w]hen client 202 is in quarantine and a user opens a web browser on client 202, QS 201 provides a web page to inform the user that the client 202 is in quarantine and corrective action must be taken." Lewis at 10:63-66. The webpage served by the quarantine server is displayed in the same browser the user has already opened,

thereby avoiding any need for additional software components or extra operations to receive and display the notification message to the user. The redirection to Lewis's quarantine server is triggered by the user opening a browser and attempting to access network resources. *See* Lewis at 10:63-66 ("When client 202 is in quarantine and a **user opens a web browser on client 202**, QS 201 **provides a web page** to inform the user that the client 202 is in quarantine and corrective action must be taken."). Redirecting the device to a quarantine server and then serving a web page from the quarantine server would merely require displaying the web page in the browser that the user is already using. This would be advantageous to sending a notification message, as in Gleichauf or Ovadia, that would potentially require additional software components running on the device to receive and display the message to the user. *See* Gleichauf at 4:56-60 ("At the same time the policy server determines access, the **policy server can transmit one or more notification messages to the posture agent and/or posture plug-ins running on the computer device** attempting the gain access to the network."); Ovadia at 16:62-65 ("…a display string containing an error message").

87. A POSITA would have had a reasonable expectation of success in combining the teachings of Gleichauf and Ovadia with Lewis. Lewis explains that the quarantine server may be implemented "on a trusted machine in a network"—

specifically, on the "network server 203." Lewis at 4:9-11 ("The invention provides a quarantine server located on a trusted machine in a network, and a quarantine agent located on a client computer that wishes to gain access to network resources administered by an organization."), 11:16-19 ("In an alternative embodiment of the invention, the QS 201 is installed on the same physical machine as the network server 203."). In the Gleichauf-Ovadia-Lewis combination, a POSITA would incorporate Lewis's quarantine server functionality into Gleichauf's policy server, because both Gleichauf's policy server and Lewis's network server were responsible for verifying the posture credentials (or bill of health) of a device seeking access to network resources. Gleichauf at 10:30-33 ("The policy server 28 is a computerized device configured to determine the particular posture credentials to request from the posture agent, and validate the posture credentials 32 associated with the user device 22."); *see* Lewis at 12:1-15. Claim 1 does not require any particular physical location for the quarantine server, and incorporating it with the policy server is consistent with the '705 patent's own disclosure that the quarantine server may reside on another system component, such as router 1026 and/or gateway 1028. '705 patent at 12:11-13 ("In various alternative embodiments the quarantine server 1034 is omitted or resides on the router 1026 and/or gateway 1028.").

88. Further, as explained above, Gleichauf's policy server sends a notification message to a quarantined device with information and remediation instructions. Gleichauf at 5:4-11. In other words, the policy server determines whether a device must be quarantined and what actions are necessary to remediate the security issue to allow the device to access the network. Gleichauf at 7:3-8, 4:60-63. It would have been straight-forward for a POSITA to provide the contents of Gleichauf's notification message in a web page as doing so would display the contents to the user within the browser the user is already using.

89. The combination with Lewis would have entailed only nominal changes to Gleichauf's system because Lewis's quarantine server functionalities would be incorporated into Gleichauf's existing policy server, which already had the capability to verify a device's posture credentials and issue access restriction or redirection instructions. Gleichauf at 15:50-54 ("In step 122, when (e.g., in addition to or instead of) receiving an access instruction 48, the data communications device 26 receives a redirect access instruction 62 generated by the policy server 28 in the analysis of the posture response 44 relative to the accepted credential set 46."). The '705 patent itself describes the functionalities of its quarantine server primarily as "provid[ing] a notice and/or other information and/or instructions to a user of the quarantined host." '705 patent at 12:9-11, *see also id*., 15:1 ("provides information

and links to assist in remediation"). Adding functionalities to provide DNS redirect and to serve a webpage would have thus been well within a POSITA's capabilities. The combination would have also incorporated Ovadia's TPM, as explained below.

F. Detailed Analysis of Claims

1. Claim 1

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

90. To the extent that the preamble is limiting, Gleichauf discloses it.

91. Gleichauf teaches a "computer-implemented method to control access to resources in a network." Gleichauf at 27:61-62. Figure 1, reproduced below in element 1.1, illustrates a protected network 24. Gleichauf at Fig. 1, 8:32-36 ("FIG. 1 illustrates a data communications system 20, according to one embodiment of the invention. The data communications system 20 includes a user device (e.g., computerized device) 22 and a network 24 having network resources 25, a data communications device 26, and a policy server 28."). Gleichauf's method protects the network by reducing the risk of the network becoming infected with malware or suffering from a virus outbreak and ensuring that the device attempting to access the network is authorized and is in fact the device that it claims to be:

> The policy server receives the challenge responses and, based upon an analysis of the posture credentials of the computerized device, determines a security state of the computerized device

> and either provides some level of or denies the computerized device access to the network resources based upon the analysis. The data communications system, therefore, detects 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 malware**. In effect, this invention helps "dampen" the propagation of **worms and viruses**, and related side effects, by **preventing systems capable of participating in such an outbreak from gaining access to the network**.

Gleichauf at 8:16-31.

> The status query mechanism is a simple authenticated challenge-response mechanism used to determine whether the posture on the computer device has changed, and also to make sure that the computer device that responds to the request is the same one that was previously authorized to access the network.

Gleichauf at 5:55-60; *see also* Gleichauf at 7:56-61, 12:57-67.

92. Therefore, Gleichauf discloses *"[a] method for protecting a network*," as recited in the preamble.

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

93. Gleichauf renders this limitation obvious.

94. **First**, Gleichauf describes a "computerized device, such as a personal computer" that attempts to communicate with restricted resources:

> During an example operation of one embodiment, **the computerized device, such as a personal computer attempting to communicate with restricted resources in the network**, transmits an access request to the data communications device (e.g., a router configured with embodiments of the invention) in an attempt to access the network resources within the network.

Gleichauf at 3:50-56.

> During operation, the computerized device such as a personal computer transmits packets or a signaling request to the data communications device in an attempt to access the network resources within the network.

Gleichauf at 8:1-4.

> The invention thus **allows the network to be protected** not only from computer devices that may be vulnerable to infection based on their security posture, but also makes sure that networks are used only by legitimate, known users and known machines.

Gleichauf at 6:39-43.

> In one arrangement, application of the access instruction is based on both an address associated with the computerized device 22, such as a source address (i.e., the network address of the user device 22), and a destination address (e.g., a network address of a resource 25 in the network 24 such as a remediation server). The data communications device 26 uses the source address and the destination address associated with the data traffic (e.g., access request 40, challenge response 43, and posture response 44) from the user device 22, to **restrict user device 22 access to particular resources 25 within the network 24 (e.g., the data communications device 26 provides a layer 3 restriction).** This can result, for example, in **providing a limited path to certain network resources** 25-1, while disallowing access to

other resources 25-2 through 25-N.

Gleichauf at 15:5-19.

95. Gleichauf's computerized device 22 is a *first host* that is attempting to access network resources 25, which are part of a *protected network*. The network resources 25 are protected by access control measures. The access requests, packet transmissions, and attempts to access the network resources show that Gleichauf's computerized device 22 *has connected or is attempting to connect to a protected network*.

96. **Second**, Gleichauf teaches that a policy server *detects an insecure condition* in the computerized device 22.

97. Specifically, the computerized device's communication attempt triggers a "posture based challenge-response" interaction during which the device transmits its posture credentials to the network. *See* Gleichauf at 11:29-32 ("The user device 22 **transmits the posture response 44 to the network 24 where the posture response 44 includes the posture credentials 32 of the user device 22**."). The posture credentials can identify, for example, "**critical security information such as what particular virus software and virus definition versions are installed within the computer system**." Gleichauf at 3:36-45. The policy server 28 of the network analyzes these posture credentials and detects, for example, the

absence of an anti-virus definition, of an anti-virus software, or recent execution of the anti-virus software, any of which indicate that the computerized device 22 can be considered to be "vulnerable." This "vulnerability" is identified because network resources may become "infected" with malware carried by "vulnerable" devices. A device would also be "vulnerable" to any malware that might be transmitted to it by various network resources. A POSITA would thus have understood Gleichauf gathering and analyzing posture credentials to render obvious "*detecting an insecure condition*":

> While connected to the network, the conventional computerized devices can be subject to receive malware, such as computer viruses, worms, spy-ware, spam, or other types of unauthorized applications or content (here after referred to as "malware").

Gleichauf at 1:10-14.

> The **security posture** of the user device 22 relates, for example, to the user device's 22 **ability to resist reception or transmission of malware** (e.g., viruses), **content** (spam and/or data theft), or **access by unauthorized users**.

Gleichauf at 8:52-56.

> Assume in another example, however, that based upon the

> comparison the policy server 28 **fails to detect the presence of, or a listing of, the anti-virus definition version** from the particular manufacturer within the accepted credential set 46. In such an example, in one arrangement, the policy server 28 transmits an access instruction 48 to the data communications device 26 that directs the data communications device 26 to restrict or completely reject communications from the user device 22 to the resources 25 within the network 24. This may include providing limited network access, such as only providing a path out of a corporate network to the Internet, but fully restricting access to any internal sub-networks.

Gleichauf at 11:58-12:13.

> In another example, assume that during the analysis of the posture credentials 32 of the computerized device 22, the policy server 28 **fails to detect the presence of an anti- virus application (e.g., or an up-to-date anti-virus application or virus definition file**) associated with the user device 22. The access instructions 48 generated by the policy server 28 and then transmitted to the data communications device 26 directs the data

> communications device 26, in turn, to restrict (e.g., prevent) communications traffic between the user device 22 and the network resources 25. In such an example, **because the user device 22 fails to include an associated anti-virus application, the user device 22 has a relatively high risk for transmitting a virus** to the network resources 25 of the network 24. As such, the data communications device 26, in one arrangement, denies the user device 22 access to the network resources 25, based upon the access instruction 48 because the user device has a relatively high probability or risk of passing a virus into the network 24.

Gleichauf at 14:16-34.

> Assume, however, that based upon the analysis, the policy server 28 **fails to detect the presence of, proper configuration, or timely execution of the anti-virus application** from the particular manufacturer within the accepted credential set 46. In such an example, in one arrangement, the policy server 28 transmits a network admission instruction 48 to the data communications device 26 that directs the data communications device 26 to restrict or reject communications from the user

> device 22 to the resources 25 within the network 24.
>
> In the above-described arrangement, the policy server 28 detects the real-time security posture or security state of a user device 22 (e.g., as indicated by the posture credentials 32) prior to providing the user device 22 with access to the network resources 25. By detecting the real-time security state of the user device 22, the policy server 28 aids in **minimizing "vulnerable" user devices 22 (e.g., user devices 22 that are not configured with anti-virus or up- to-date anti-virus applications) from accessing the network 24**. As such, the policy server 28 aids in minimizing the risk that the network 24 or network resources 25 become "infected" with malware carried by the "vulnerable" user device 22.

Gleichauf at 22:38-59.

98. Gleichauf describes detecting other potential *insecure conditions* based on, for example, the version of an operating system on the device, the presence of updated "patches" associated with the operating system, and when was the last time a local virus scan was performed because vulnerabilities may exist as a result of these device characteristics:

> Also assume that a scanning application periodically scans computerized devices connected to LAN (e.g., once a day) for the presence of **known vulnerabilities** in the device **operating system, resident applications, associated security programs, and their respective configurations**.

Gleichauf at 1:65-2:3.

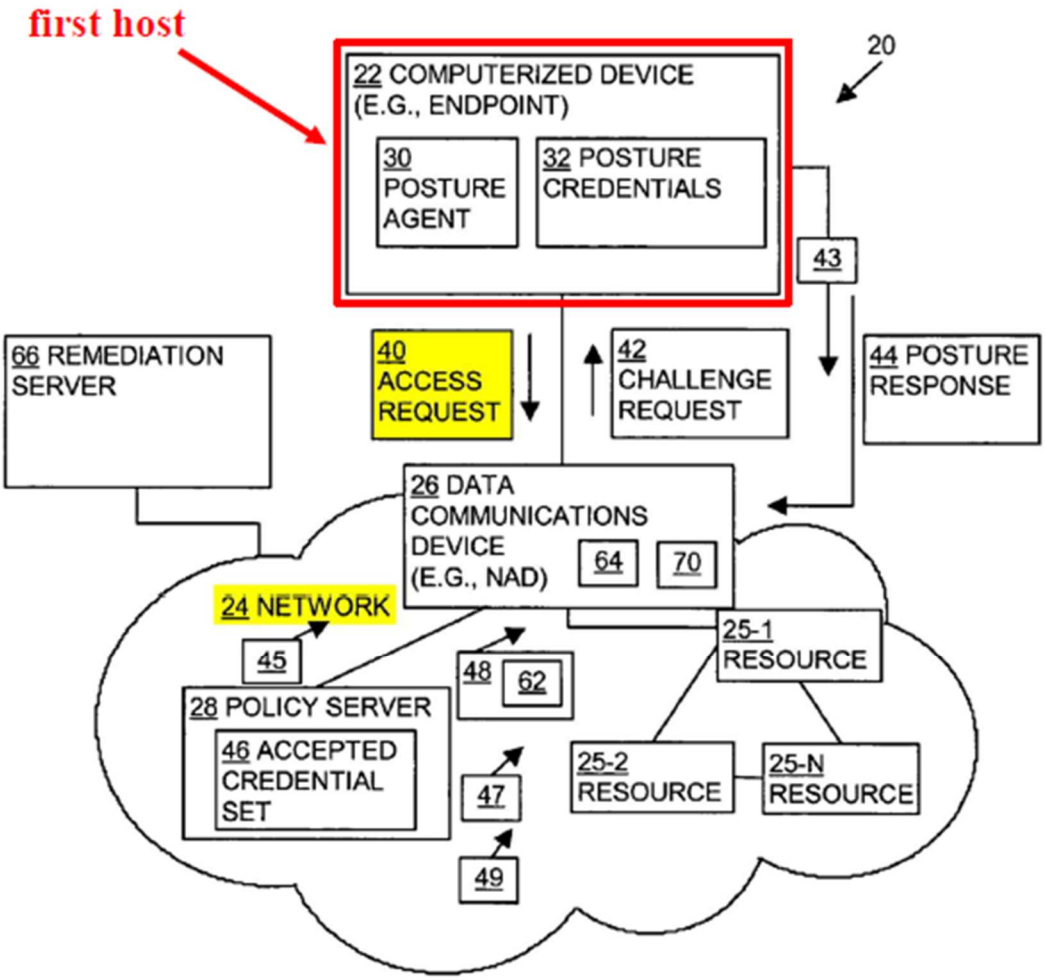
> In another arrangement, the user device 22 includes a posture plug-in 32 that indicates the **version of operating systems** used by the user device 22 or the presence of updated "patches" associated with the operating systems.

Gleichauf at 8:65-9:1.

> Posture credentials collected in real time from the posture plug-ins by the posture agent can identify critical security information Such as what particular virus software and virus definition versions are installed within the computer system, **when was the last time a local antivirus scan was executed, what operating system patches are installed within the computer system**, what particular types and versions of software applications are configured and active within the computer system, and so forth.

Gleichauf at 3:36-45; *see also* Gleichauf at 31:1-7.

99. Gleichauf depicts in Figure 1, below, an embodiment that includes the computerized device 22 sending an access request 40 to the network 24 ("*attempting to connect to a protected network*"):



Gleichauf at Fig. 1 (annotated).

100. Therefore, Gleichauf discloses "*detecting an insecure condition on a first host that has connected or is attempting to connect to a protected network, wherein detecting the insecure condition includes*."

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

101. The combination of Gleichauf and Ovadia renders this limitation obvious.

102. **First**, Gleichauf's computerized device 22 includes a "posture agent" 30 and one or more "posture plug-ins" 32 installed on the device, which collect information about the device's security posture:

> According to some embodiments of the invention, computer systems such as personal computers can be configured with security agent software referred to herein as a security, or "posture" agent that can be remotely activated, and one or more software plug-ins, hereafter referred to "posture plug-ins", that the posture agent uses to collect information associated with the security posture of the computer device.

Gleichauf at 3:9-16.

103. The posture agent accesses the posture information collected by each posture plug-in using a "posture plug-in API" that is implemented by each posture plug-in manufacturer:

> There is a well-defined application-programming interface (API) here-in-after named "**posture plug-in API**" that each posture

> plug-in manufacturer must implement to **allow the posture agent to access posture information**.

Gleichauf at 9:24-28.

104. The posture agent uses the posture information to prepare posture credentials that it transmits to a policy server for analysis and access control. The posture agent can be implemented in either software or hardware:

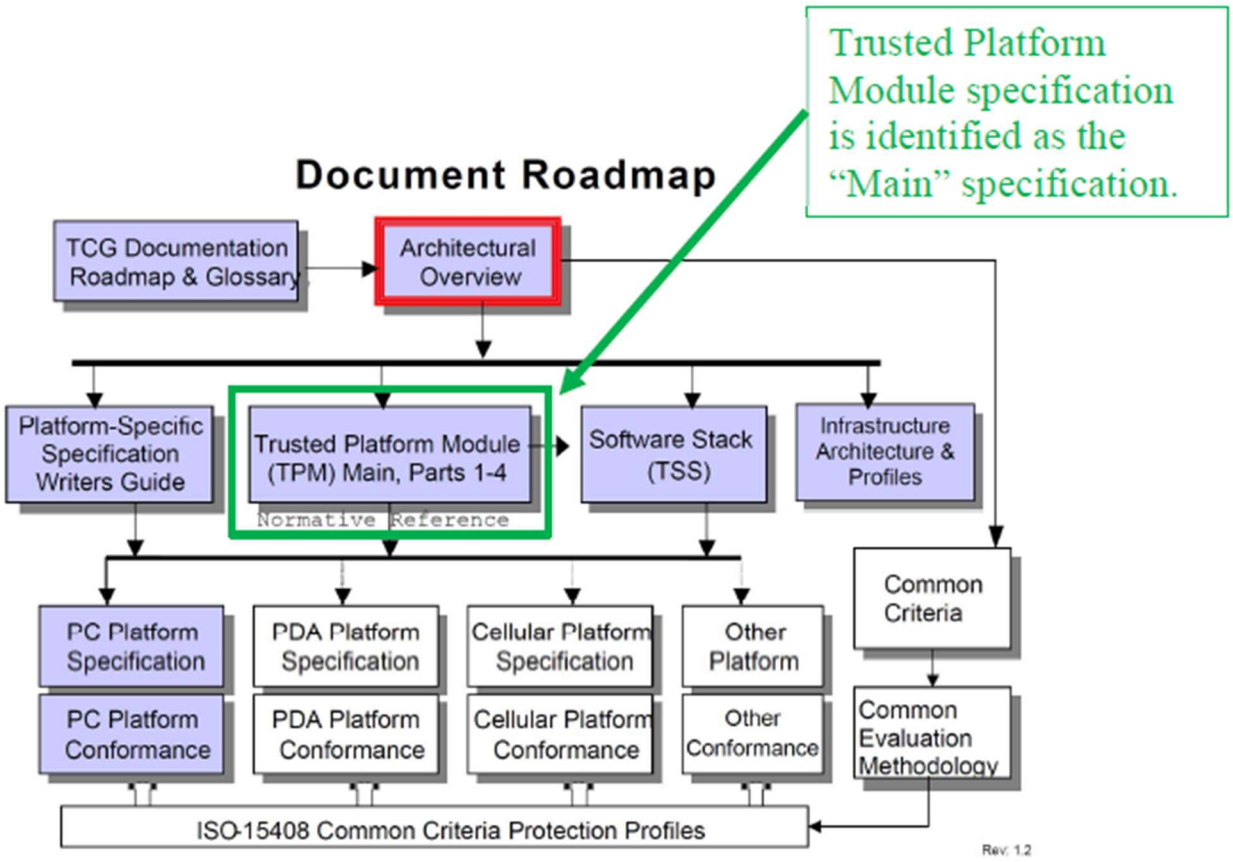
> **The user device 22 also includes a posture agent 30** configured to receive a sequence of challenge requests 42, and in response, **collects posture credentials from the installed posture plug-ins 32 and transmits posture credentials 32, associated with the user device 22, to the policy server 28. The posture agent 30, embodied as a software or hardware application**, works in conjunction with the data communications device and policy server 28 **to verify a security status** of the user device 22 prior to the user device 22 accessing the network 24 (e.g., the network resources 25 within the network 24).

Gleichauf at 9:29-39.

105. In other words, Gleichauf's posture credentials are used to secure access to restricted network resources. A POSITA would have understood that the

combined functionality of the posture agent and posture plug-ins on the computerized device is part of the mechanism that provides such security and, as a result, the combined functionality of the posture agent and posture plug-ins is a "*trusted computing base*" as claimed. *See* § IV.A (construing "*trusted computing base*").

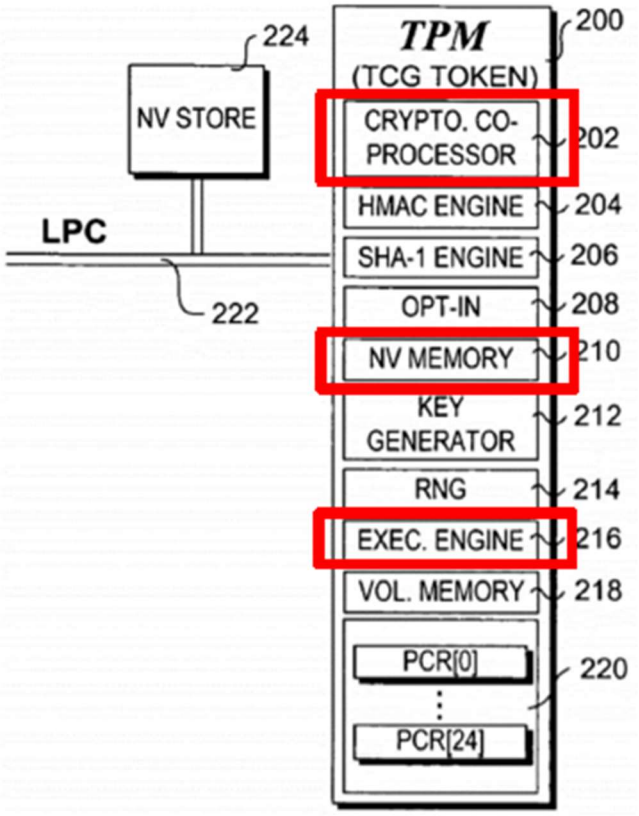
106. **Second**, Ovadia discloses a security apparatus that employs a trusted platform module ("TPM") implementing the TCG "main specification (Version 1.2, October 2003…)." Ovadia at 4:16-47. When Ovadia identifies the TCG main specification, Ovadia at 4:32-33, it is referring to the "main specification" from TCG, which is the Trusted Platform Module (TPM) Main Specification, Version 1.2, dated October 2003:



TCG Specification Architecture Overview at iv (annotated).

See also TPM Main Specification.

107. Ovadia's TPM, depicted in Figure 2 below, implements the TPM main specification from the TCG and further includes a "cryptographic co-processor," "non-volatile (NV) memory," and an "execution engine." Ovadia at 4:46-55. Since Ovadia's TPM implements the TPM main specification from the TCG, it would have been obvious to a POSITA that the non-volatile memory is used to store cryptographic keys because that function is described in the TPM main specification. TPM Main Specification at 13, 19; TCG Specification Architecture Overview at 19.



Ovadia at Fig. 2 (annotated).

108. Therefore, Ovadia's TPM is a *trusted platform module* as claimed. *See supra* § IV.B (construing "*trusted platform module*"). The TPM "may be embodied as a hardware device (most common) or via software." Ovadia at 4:24-26. The TPM, like Gleichauf's posture agent, is used "to verify [that] a current configuration of a subscriber station platform is an authorized configuration." Ovadia at Abstract. The configuration, referred to as the integrity measurement, includes information about the subscriber station:

> The initial signup phase begins in a block 800, wherein **a**

measurement of at least one of the SS platforms authorized firmware and software configuration(s) is performed to produce an integrity measurement corresponding to the authorized configuration(s). The integrity measurement is then stored in an appropriate PCR in a block 802, and sent to the AS as an authorized Config_ID in a block 804. As above, the integrity measurement may be encrypted and/or digitally signed with appropriate keys. The AS extracts the Config_ID and stores it in its database, which completes the initial signup process.

In response to initiation of a post-signup access, the SS will receive an authentication request message or send the following in an authentication request. First, in a block 806, a **current measurement of the firmware and/or software configuration (corresponding to the earlier measurement) is made to produce a current integrity measurement**. The appropriate PCR is then hash-extended with the current integrity measurement in a block 808. The hash-extended PCR value is then extracted, digitally signed with the SS's AIK private key, and sent to the AS in a block 810. Upon receipt, the AS

> authenticates the digital signature using the SS's AIK public key that it has previously received.

Ovadia at 13:57-14:12.

109. **Third**, the policy server "*contacts*" the computerized device's posture agent, prompting the posture agent to collect the computerized device's posture credentials from one or more posture plug-ins:

> The challenge request sequence happens in real time and causes the **posture agent within the computerized device to collect information associated with the security posture of the computer device, hereafter referred to as "posture credentials", from one or more posture plug- ins**, and to transmit these posture credentials securely to a policy server operating within the network to determine what type of admission policy and level of network access should be assigned to that device.

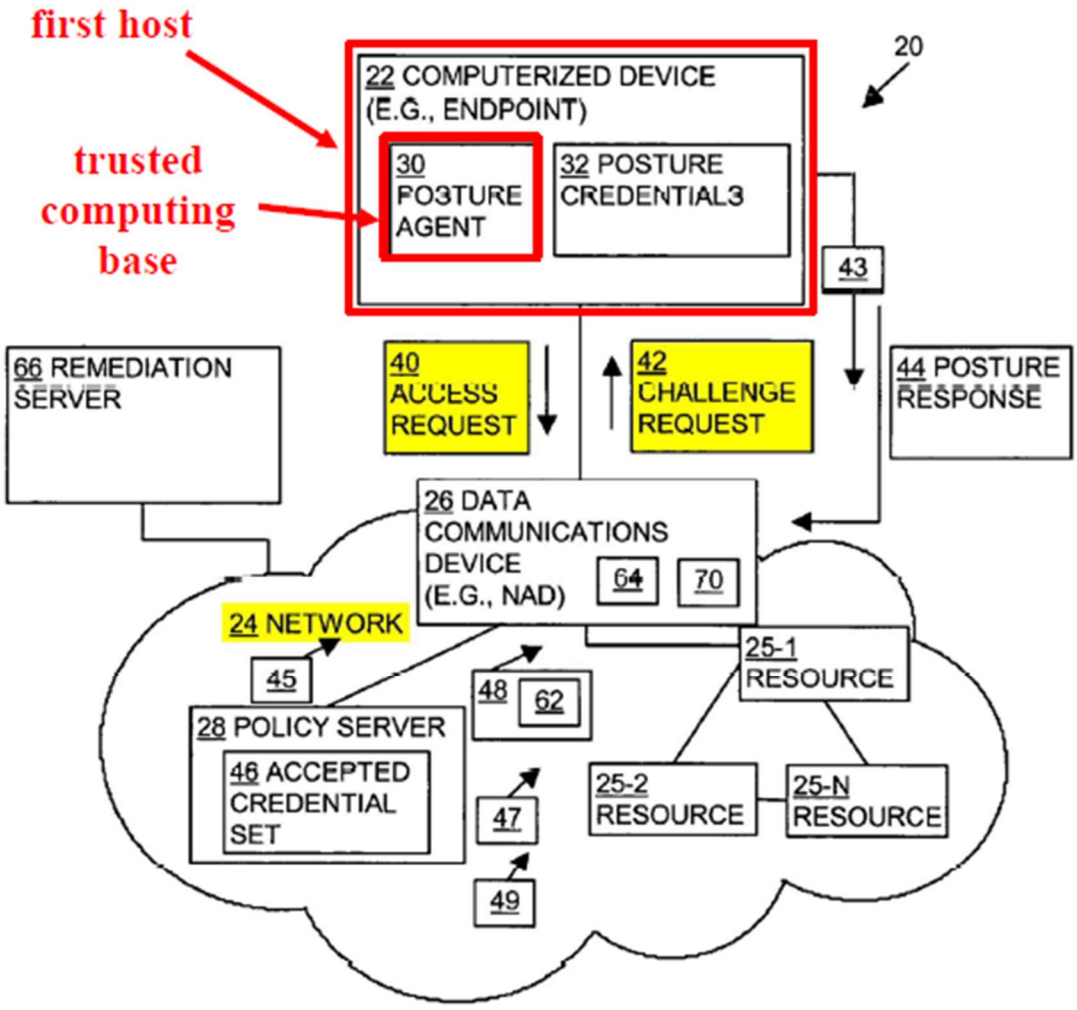
Gleichauf at 3:28-36.

> Once the secure channel has been established, **the policy server submits a request to the posture agent via the data communications device for all or a selected set of posture**

credentials.

Gleichauf at 4:7-11.

110. Gleichauf illustrates in Figure 1 (below) that the posture agent is *within* the computerized device 22 ("*the first host*"):



Gleichauf at Fig. 1 (annotated).

111. It would have been obvious to a POSITA that the "t*rusted platform module*" would also be "*within*" the computerized device.

112. As described in more detail in Section VI.D, 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. A POSITA would have appreciated the value of implementing Gleichauf's posture agent with TPM functionality. The TCG, which prepares the TCG specification discussed above, is "an **industry consortium** concerned with platform and network security." Ovadia at 4:31-32. The TCG specification, in turn, is a "**platform-independent industry specification** that covers trust in computing platforms" as well as how to implement and use TPMs. Ovadia at 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 various computerized devices.

113. From the device side (e.g., device 22 of Gleichauf), a posture agent implemented with TPM functionality would operate consistent with the TCG specification, regardless of the device it is running on (given that TPM is "platform independent"). Ovadia at 4:32-35. And operation of such TPMs would be consistent throughout the adopting industry. Furthermore, because the TPM specification (as an industry standard) had undergone extensive vetting and security testing, it was particularly well-suited for security-critical implementations like Gleichauf's. Implementing a TPM in Gleichauf's device therefore would have ensured (or greatly

increased) the likelihood of a stable, secure platform free from security backdoors or other vulnerabilities.

114. From the network side, a policy server, receiving Gleichauf's posture credentials, would process them more quickly and with a higher assurance that the necessary information is provided. Indeed, the policy server would be able to process all received posture credentials in substantially the same way simply because such credentials would have been gathered in substantially the same way. In the combination, Gleichauf's posture agent would be augmented to implement the TPM functionality, as taught by Ovadia (*i.e.*, "*trusted platform module*").

115. A POSITA would have appreciated that one obvious way to 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. *See, e.g.*, Ovadia at 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" to collect posture information. Gleichauf at 9:24-28. With Gleichauf's posture agent executing in Ovadia's 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 from the TCG. Ovadia at 5:36. A POSITA would have appreciated that another obvious way to

implement Gleichauf's posture agent with TPM functionality would be to simply integrate the posture agent software with a software-implemented TPM, as taught by Ovadia. See Ovadia at 4:24-26 ("TPM functionality may be embodied… via software."). With the posture agent (part of the "*trusted computing base*") either executing on TPM hardware or integrated with TPM software, the functionality of the posture agent and posture plug-ins would be "*associated with a trusted platform module*."

116. Therefore, the combination of Gleichauf and Ovadia discloses "*contacting a trusted computing base associated with a trusted platform module within the first host*."

d. [1.3] *receiving a response, and*

117. Gleichauf discloses and/or renders this limitation obvious.

118. The computerized device'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 posture response 44 to the policy server:

> The challenge request sequence happens in real time and causes the posture agent within the computerized device to collect information associated with the security posture of the computer device, hereafter referred to as "posture credentials", from one or

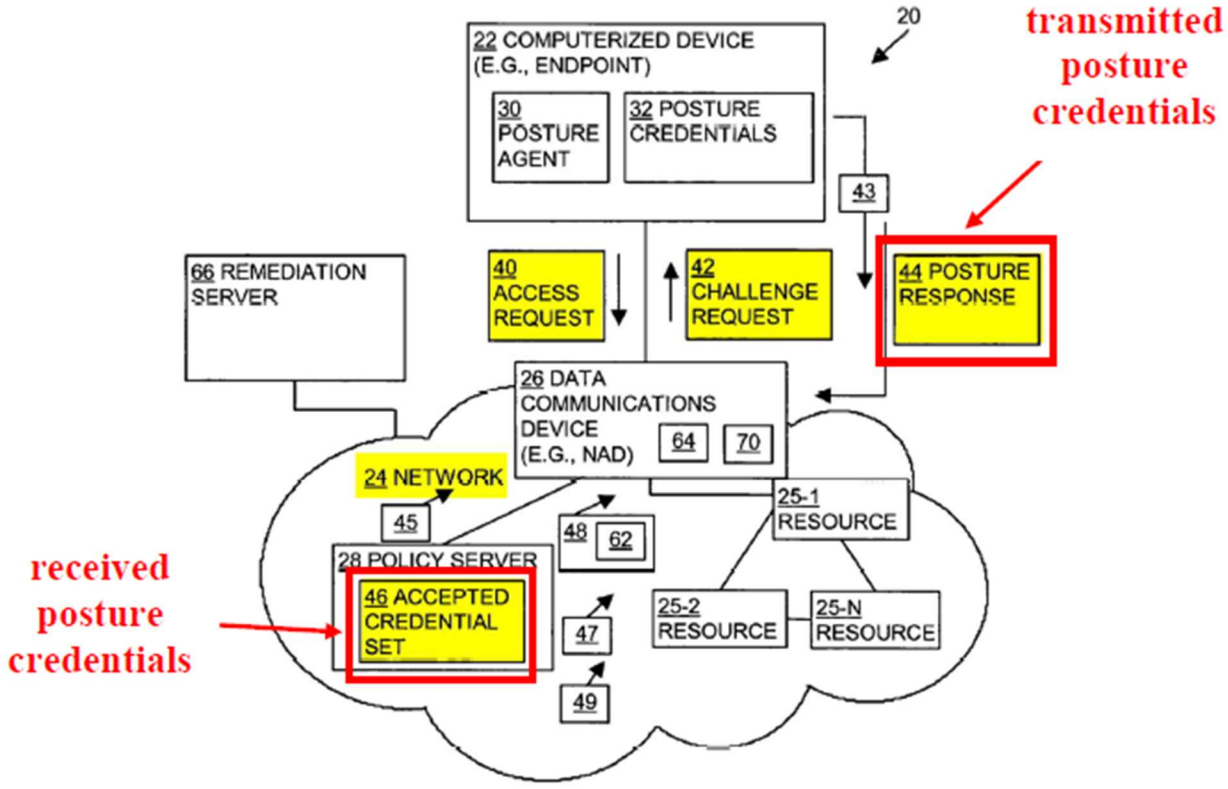
> more posture plug-ins, and to **transmit these posture credentials securely to a policy server** operating within the network to determine what type of admission policy and level of network access should be assigned to that device.

Gleichauf at 3:28-36.

> The posture agent requests the specified posture credentials from the associated posture plug-ins installed on the device and **transmits the requested posture credentials back to the policy server via the data communications device**.

Gleichauf at 4:11-15.

119. Thus, the policy server "*receiv[es]* a response" to its request for the posture credentials. Gleichauf illustrates in Figure 1 (below) the transmission and reception of the posture credentials:



Gleichauf at Fig. 1 (annotated).

120. Therefore, Gleichauf discloses "*receiving a response*."

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

121. The combination of Gleichauf and Ovadia renders this limitation obvious.

122. **First**, as discussed at [1.3], Gleichauf's posture response includes posture credentials. The computerized device's posture credentials (received by the policy server) contain information such as the status of anti-virus definitions, last time a local virus scan was performed, operating system patches, and types and

versions of installed software applications (each individually and collectively providing an "*attestation of cleanliness*"):

> Posture credentials collected in real time from the posture plug-ins by the posture agent can identify critical security information such as what particular **virus software and virus definition versions are installed** within the computer system, when was **the last time a local antivirus scan was executed, what operating system patches are installed** within the computer system, what particular **types and versions of software applications** are configured and active within the computer system, and so forth. By providing a security posture based challenge- response mechanism for network admission control, embodiments of this invention are able to dramatically reduce the opportunities for injection of malware into a network.

Gleichauf at 3:36-49. This is consistent with the examples of attestation of cleanliness described in the '705 patent:

> In some embodiments, the query for cleanliness (1302) may be responded to by anti-contagion software, such as antivirus software, with assertions about the currency of a scan, such as

> the last time a scan was performed, or a version associated with a current anti-contagion Software or definition file in use, wherein a sufficiently updated software and/or scan may act as a cleanliness assertion.

'705 patent at 14:15-19.

123. **Second**, the policy server reviews and validates the posture credentials received from the computerized device (*i.e.*, "*determining whether the response includes a valid . . . attestation of cleanliness*"):

> **The policy server 28** is a computerized device configured to determine the particular posture credentials to request from the posture agent, and [to] **validate the posture credentials 32 associated with the user device 22**.

Gleichauf at 10:30-33.

124. For example, the policy server may accept as valid a set of posture credentials that show an anti-virus application and a desired version of anti-virus definitions:

> The policy server 28 reviews the accepted credential set 46 to detect the presence (e.g., a listing) of the identity of the anti-virus application and definition version from the particular

> manufacturer within the accepted credential set 46. Assume, for example, that based upon the comparison the **policy server 28 detects the presence of, or a listing of the same anti-virus application and definition version from the particular manufacturer within the accepted credential set** 46. In such an example, in one arrangement, the policy server 28 transmits an access instruction 48 to the data communications device 26 that directs the data communications device 26 to allow communications between the user device 22 and the resources 25 within the network 24.

Gleichauf at 11:45-58.

125. The policy server reviews the returned posture credentials, for example, for the presence of an anti-virus application and definition (*i.e.*, "*determining*"). In this example, as described in more detail below, if the current anti-virus definition is present, the computerized device will be allowed to communicate with network resources, which demonstrates that the policy server deems the response to include "*a valid . . . attestation of cleanliness*." Gleichauf at 11:45-58. If, on the other hand, the current anti-virus definition is not present, communications from the computerized device to the network will be restricted or completely rejected,

showing that the response is deemed not to include "*a valid . . . attestation of cleanliness*." Gleichauf at 11:58-67.

126. **Third**, Gleichauf explains that the computerized device's posture agent transmits posture credentials (*i.e.*, "*attestation of cleanliness*") "securely" to the policy server:

> The data communications device 26, policy server 28, and posture agent 30 may **securely transmit posture** and/or identity using protocols such as the Extensible Authentication Protocol (EAP), Protected Extensible Authentication Protocol (PEAP) and Flexible EAP Authentication using Secure Tunnel Protocol (EAP-FAST) over various network transports, including a Layer 2 transport Such as wired LAN (Ethernet) and wireless LANs, and a Layer 3-based transport such as User Datagram Protocol (UDP).

Gleichauf at 11:3-12.

127. **Fourth**, Ovadia provides further details regarding the secure transmission of a communication device's configuration information for the purpose of gaining access to a network. Like Gleichauf, Ovadia's device seeking network access—a subscriber station, "SS"—uses its TPM to respond to an authentication

request message by making a "current measurement of the firmware and/or software configuration" and uses this as a "current integrity measurement." Ovadia at 14:1-6. The current integrity measurement is stored in the TPM and used to produce a value that is "**digitally signed** with the SS's AIK [Attestation Integrity Key] private key, and sent to the AS [Authentication Server]." Ovadia at 13:61-63, 14:6-10, Fig. 8. If the integrity measurement of the subscriber station is acceptable and "the digital signature is authentic," then the subscriber station "is authenticated" and allowed access to the network. Ovadia at 14:13-23; *see also* Ovadia at 16:66-17:6 ("…the requesting SS, whose identity was authenticated through the SS AIK private key signature, is authorized for basic unicast [communication] services…").

128. As described in more detail in Section VI.D, 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. A POSITA would have understood the role Ovadia's digital signature played in authenticating a user's device. Digitally signing Gleichauf's posture credentials, as taught by Ovadia, would provide a mechanism for Gleichauf's policy server (which evaluates those credentials) to be assured that the posture credentials have not been forged or tampered with by a malicious actor. Additionally, at the time of the alleged invention in 2004, digital signatures were already widely known and commonly used for security

authentication and validation, with their use dating back to the 1980s. As discussed in [1.2], in the combination, Gleichauf's posture credentials would be prepared by a posture agent implemented with TPM functionality, as taught by Ovadia. Ovadia explains that a device's integrity measurements (*e.g.*, posture credentials) are digitally signed with an Attestation Integrity Key (AIK), and that "AIKs are only permitted to sign data generated by a TPM." Ovadia at 14:9, 5:35-36. Since only the TPM (embodied in Gleichauf's posture agent implemented with TPM functionality as discussed at [1.2]) 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. In other words, the policy server would be assured that a virus or other malware on the computing device had not tampered with or forged the posture credentials to hide the presence of the virus or malware on the computing device. 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.

129. Additional analysis of the obviousness of combining Gleichauf and Ovadia is provided above in Section VI.D.

130. Therefore, the combination of Gleichauf and Ovadia discloses "*determining whether the response includes a valid digitally signed attestation of cleanliness*."

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

131. Gleichauf discloses this limitation.

132. The computerized device's posture credentials that are transmitted to the policy server (the "*attestation of cleanliness*" in [1.4]), contain information about "the last time a local antivirus scan was executed." Gleichauf at 3:41. Gleichauf expressly contemplates that the failure to show "timely execution of the anti-virus application" is treated as a failure to show cleanliness and results in restrictions on the computerized device's ability to access the network. Gleichauf at 22:38-47. It would have been obvious to a POSITA that the opposite showing would have the opposite effect. In other words, it would have been obvious for a posture credential that does show the timely execution of the anti-virus application to be treated as "*an*

attestation that the trusted computing base has ascertained that the first host is not infested."

133. Gleichauf's posture credentials also contain information about the presence (or absence) of an anti-virus software and currency of an anti-virus definition, operating system patches, and particular types and versions of software applications on the computerized device. This information is gathered by the posture plug-in ("*trusted computing base*"):

> **Posture credentials** collected in real time from the posture plug-ins by the posture agent can identify critical security information such as what **particular virus software and virus definition versions** are installed within the computer system, when was the last time a local antivirus scan was executed, what **operating system patches** are installed within the computer system, what particular **types and versions of software applications** are configured and active within the computer system, and so forth. By providing a security posture based challenge-response mechanism for network admission control, embodiments of this invention are able to dramatically reduce the opportunities for injection of malware into a network.

Gleichauf at 3:36-49.

134. Such information on software and patches are "*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*."

135. Gleichauf's posture agent prepares the posture credentials from information it collects from the posture plug-ins. Gleichauf at 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." Gleichauf at 9:50-53. The combined functionality of the posture agent and posture plug-ins, i.e., the "*trusted computing base*," thus "*ascertains*" the security posture of the device. Ex.1003, ¶ XX.

136. Gleichauf's disclosures regarding the claimed *attestation* are substantially the same as those provided in the '705 patent specification. The '705 patent notes that the attestation of cleanliness may include "assertions about the currency of a scan, such as the last time a scan was performed, or a version associated with a current anti-contagion software or definition file in use." '705 patent at 14:12-19. Specifically, the '705 patent states:

> In some embodiments, the query for cleanliness (1302) may be responded to by anti-contagion software, such as antivirus

> software, with **assertions about the currency of a scan, such as the last time a scan was performed, or a version associated with a current anti-contagion software or definition file in use, wherein a sufficiently updated software and/or scan may act as a cleanliness assertion**.

'705 patent at 14:12-19.

137. A POSITA would have understood Gleichauf's disclosure that posture credentials contain information about "the last time a local antivirus scan was executed," Gleichauf at 3:36-49, to equate to the '705 patent's assertions about the currency of a virus scan, e.g., how recently a virus scan was performed.

138. Therefore, Gleichauf discloses "*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*."

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

139. Gleichauf renders this limitation obvious.

140. **First**, as explained in [1.4], the policy server reviews and validates the posture credentials received from the computerized device. Gleichauf at 10:30-33, 11:45-58. The policy server reviews the returned posture credentials, for example, for the presence of an anti-virus application and definition (Gleichauf at 11:45-58) or the timeliness of the last virus scan (Gleichauf at 22:38-47). *See also* Gleichauf at 3:28-45, 4:21-27, 23:56-65.

141. **Second**, Gleichauf teaches using the posture credentials to determine whether to place a device in quarantine:

> Devices that are more significantly out of compliance may be placed on an isolated, or "**quarantine**" network segment where they can be brought into compliance.

Gleichauf at 4:24-27.

> In one arrangement, the data communications device assigns devices that are out of compliance with admission requirements to a quarantine, or remediation network.

Gleichauf at 4:50-53.

> Based on posture information gleaning from these other mechanisms, the policy server may determine the appropriate

> admission rules to be downloaded to the data communications device including allowing access to restricted resources, **quarantining the computer device** or denying access completely.

Gleichauf at 7:3-8.

> For example, in the case that a user device 22 is non-compliant, the message may be displayed to the user or written to a log file indicating that the device has been quarantined and needs to be remediated.

Gleichauf at 21:5-8.

142. As Gleichauf indicates, there are many reasons why a device might be placed in quarantine. For example, a device may be quarantined if it is found to be out of compliance with a network admission policy. Gleichauf at 4:50-56. A device might also be put into quarantine if its behavior is found to be anomalous, suggesting that it may have been compromised. *See* U.S. Patent Pub. No. 2005/0097199 at [0039] ("For example, if the first network device 110 is a **server that always services requests** from other devices, **an anomaly occurs when the server begins making requests**. … The scanning action module 215 can then perform one or more of the actions described above or below (e.g., **quarantine** the first network device

110,…"). Another reason a device might be quarantined is if it is found to be infected with a virus or is found to have been hijacked by an unauthorized entity. *See, e.g.*, U.S. Patent No. 7,571,460 at 4:8-14.

143. For example, if the policy server "fails to detect the presence of, or a listing of the anti-virus definition" in the computerized device's posture credentials, the device is out of compliance with network admission policy (*i.e., "when it is determined that the response does not include a valid digitally signed attestation of cleanliness*"). Devices that are out of compliance are quarantined such that communications from such devices to network resources are either restricted or rejected (*i.e., "quarantining the first host, including by preventing the first host from sending data to one or more other hosts associated with the protected network*"):

> In one arrangement, the **data communications device assigns devices** that are out of compliance with admission requirements **to a quarantine**, or remediation network. This remediation admission policy is issued by the policy server based on comparison of the posture credentials of the device relative to predefined network admission requirements, or security policy.

Gleichauf at 4:50-56.

Assume in another example, however, that based upon the

> comparison the **policy server 28 fails to detect the presence of, or a listing of the anti-virus definition version** from the particular manufacturer within the accepted credential set 46. In such an example, in one arrangement, the **policy server 28** transmits an access instruction 48 to the data communications device 26 that **directs the data communications device 26 to restrict or completely reject communications from the user device 22 to the resources 25 within the network 24**.

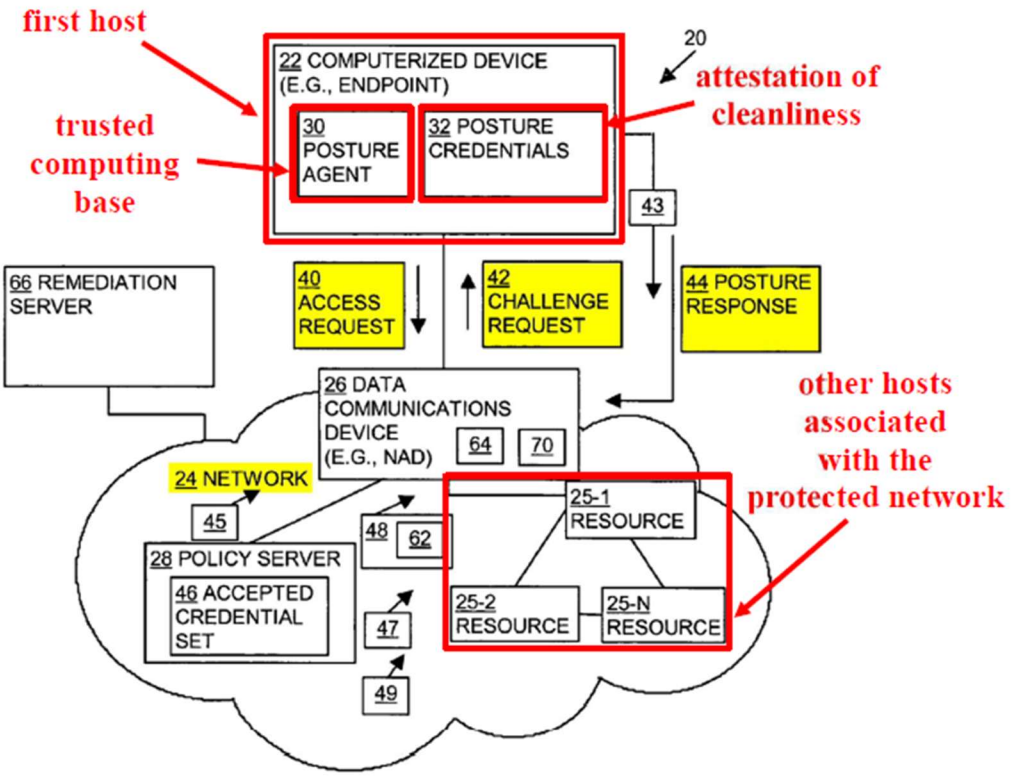
Gleichauf at 11:58-67.

> Assume, however, that based upon the analysis, the policy server 28 fails to detect the presence of, proper configuration, or timely execution of the anti-virus application from the particular manufacturer within the accepted credential set 46. In such an example, in one arrangement, the policy server 28 transmits a network admission instruction 48 to the data communications device 26 that directs the data communications device 26 to **restrict or reject communications from the user device 22 to the resources 25 within the network 24**.

Gleichauf at 22:38-47.

144. Gleichauf describes that resources 25 can be, for example, "servers that contain content accessible by the user device 22." Gleichauf at 9:65-67. This is consistent with the '705 patent's characterization that a host can be a server. '705 patent at 12:7-11 ("In some embodiments, **requests to connection to a host other than the remediation server 1032** are redirected to a quarantine server 1034 configured to provide a notice and/or other information and/or instructions to a user of the quarantined host."), *see also* 20:48-50 ("remediation host").

145. Gleichauf illustrates in Figure 1 that communications from the computerized device 22 pass through data communications device 26. Thus, the data communications device 26, at the direction of the policy server, can prevent the computerized device 22 from communicating with network resources (*e.g.,* 25-1 through 25-N, corresponding to *"one or more other hosts associated with the protected network*"):



Gleichauf at Fig. 1 (annotated).

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

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

147. Gleichauf discloses this limitation.

148. Gleichauf describes the computerized device ("*first host*") attempting to communicate with restricted resources, such as resources 25-1, 25-2, 25-N shown in Figure 1:

> During an example operation of one embodiment, **the computerized device, such as a personal computer attempting to communicate with restricted resources in the network**, transmits an **access request** to the data communications device (e.g., a router configured with embodiments of the invention) in an attempt to access the network resources within the network.

Gleichauf at 3:50-56.

> During operation, **the computerized device such as a personal computer transmits packets or a signaling request to the data communications device in an attempt to access the network resources within the network**.

Gleichauf at 8:1-4.

149. Gleichauf further describes network traffic from the computerized device seeking to access network resources after the device has been placed into quarantine. This would have been obvious based both on common sense and

understanding of a POSITA, and based on the explanations in Gleichauf. Notably, a POSITA would have understood that a device may be attempting to access the network resources multiple times legitimately to confirm that it truly is not able to access the network or mistakenly assuming that it has successfully been updated with the proper security features; or maliciously by attempting to test the limits of the system via repeated access attempts.

150. In addition to the POSITA's understanding, Gleichauf also explains that a device is repeatedly subjected to a posture query procedure. For example, to ensure a device requesting network access is the same device previously authorized, the posture query procedure shown in Figure 1 is repeated:

> In order to detect a posture change on a user device 22 and to make sure that the user device is still the same device 22 as was previously authorized, the data communications device 26 periodically polls the user device 22 using the posture update query. In the case where the data communications device 26 detects a change in the state of the user device 22, the data communications device 26 attempts to re-authorize the posture of user device 22 as explained in FIG. 1 (i.e. triggers the re-start of the full system process as described in FIG. 1).

Gleichauf at 16:46-55. Gleichauf further notes that when a device has been directed to a remediation server but has failed to be remediated, "the device **will be again identified as being at some level of non-compliance** and the appropriate response will be invoked by the policy server." Gleichauf at 5:13-16. Therefore, a POSITA would have found it obvious based on Gleichauf's description that a device would be sending a request even after is has already deemed not trustworthy

151. The request to access network resources is a "*service request*." Gleichauf explains that, in some arrangements, resources 25 can be "servers that contain content accessible by the user device." Gleichauf at 9:65-67. Thus, a POSITA would have found it obvious that a request to access such a server to obtain content would constitute a service request. This "*service request*" is received and acted upon by a data communications device:

> The **data communications device** that detects the presence of the new device **initiates a challenge-response sequence to the device** that directs a posture agent running on the computerized device to retrieve **posture credentials** describing the security posture of the computerized device.

Gleichauf at 3:60-65.

152. Therefore, Gleichauf discloses "*receiving a service request sent by the first host*."

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

153. The combination of Gleichauf and Lewis renders this limitation obvious.

154. **First**, as explained in [1.6], Gleichauf discloses that a computerized device that is out of compliance with network admission policy is quarantined. Gleichauf at 4:50-56. Communications from a quarantined device to restricted resources in the network are restricted or rejected. Gleichauf at 11:58-67. Such communications may be limited only to certain destinations, such as a remediation server. *See* Gleichauf at 14:62-66 ("The access instruction 48 **may permit traffic from user device 22 only to certain destination addresses, e.g. IP address of remediation server**, and only to certain protocols or ports, e.g. HTTP traffic."). In such circumstances, Gleichauf explains:

> In another example, when the data communications device 26 applies the access instruction 48 to restrict a user device's 22 access to the network resources 25, the data communications device 26 **limits the user device's 22 access to only a portion of the network resources 25** (i.e., for example,

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

Gleichauf at 14:55-15:4.

155. Gleichauf further explains that "HTTP traffic"—which would include a *web server request* because hyper-text transfer protocol (HTTP) is the protocol used for communications between a web browser running on a device and web servers— is redirected:

> For example, assume that when comparing the posture credentials 32 of the user device 22 with the accepted credential

> set, the policy server 28 detects the presence of an anti-virus application associated with the user device 22 and further detects that the anti-virus application is out-of date (e.g., the user device does not have the most recent version of the anti-virus application). Based on such detection, the policy server 28 determines that a **redirect access** instruction 62 should be **applied to HTTP traffic from the user device 22**. When the data communications device 26 receives communications traffic from the computerized device 22, the data communications device 26 applies the redirect access instruction 62.

Gleichauf at 15:55-67.

156. Gleichauf refers to this redirection technique as a "URL Redirect." Gleichauf at 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 web page. *See, e.g.*, U.S. Patent Pub. No. 2005/0078668 at [0003]. For example, it was common to use URL redirection to direct a user's web browser to a login page like those that were commonly found when accessing the Internet from a public space such as a hotel or airport. *See, e.g.*, U.S. Patent No. 7,568,107 at 4:56-57 ("redirect the users to a Uniform Resource Locator (URL)

address of a network login page"); U.S. Patent Pub. No. 2004/0177276 at [0044] ("control device 20 can redirect the HTTP request to an authentication web page hosted by the control device using any HTTP redirection technique known in the art").

157. **Second**, Gleichauf describes providing a notification message to be displayed to a user when the user's computerized device has been placed in quarantine. The message may also include a link to a remediation server or the IP address of the remediation server:

> At the same time the policy server determines access, **the policy server can transmit one or more notification messages** to the posture agent and/or posture plug-ins running on the computer device attempting the gain access to the network. **The notifications may include the results of the posture check, remediation actions (if any) and an informational message to be displayed to the user**, or written to a local log file.

Gleichauf at 4:56-63.

> The **notification message** may contain a text message for display to a user or for logging to a log file (e.g., messages to display to the user or a log or URLs of the remediation server 66 and any

> remediation actions to be performed). **For example, in the case that a user device 22 is non- compliant, the message may be displayed to the user** or written to a log file **indicating that the device has been quarantined and needs to be remediated. The notification message may also include a link to a remediation server or the IP address of the remediation server 66 that a user may click on** or an application on the user device 22 may process to remediate the user device 22 automatically.

Gleichauf at 21:1-12.

158. This claim limitation is also rendered obvious by the combination of Gleichauf, Ovadia, and Lewis. Lewis also discloses the use of redirection with a quarantined client device:

> When a client requests the use of network resources from a network administrator, the network administrator requests the client's BoH. If the BoH is valid, the client is admitted to the network. **If the BoH is invalid, or if the client does not have a quarantine agent, the client is placed in quarantine, in which the only network resources accessible to the client are those**

> **necessary to install the quarantine agent and requisite software/patches to achieve a valid state**.

Lewis at 4:24-31. Thus, in the Gleichauf-Ovadia-Lewis combination, the functionality of providing a quarantine notification can additionally or alternatively be provided by Lewis's quarantine server, which is part of the policy server in the combination. Using Lewis's quarantine server to provide the quarantine notification would be beneficial for the same reasons described above in Section VI.E.

159. In this combination, when a user of the device opens a web browser (which would generate a "*service request compris[ing] a web server request*"), the initial domain name service (DNS) request from the device is redirected to a quarantine server (QS). Lewis at 13:7-12. The quarantine server (QS) receives the DNS request and supplies the device with a default web page on the quarantine server (*i.e.*, "*serving a quarantine notification page to the first host*"). The default web page, among other things, informs the user that the device is in quarantine and corrective action must be taken:

> With reference to FIG. 4, the QS 201 includes an Internet Information Service (IIS) 417 for providing a default web page. When client 202 is in quarantine and a user opens a web browser

> on client 202, QS 201 provides **a web page to inform the user that the client 202 is in quarantine** and corrective action must be taken.

Lewis at 10:61-66.

> At step 1125, **QS 201 receives the DNS request and supplies the client 202 with a default web page on QS 201** that requests the user to trust the web site and instructs the user of the client 202 to install a QA by clicking on a link to the QA.

Lewis at 13:12-15.

160. As described in more detail in Section VI.E, 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. A POSITA would have appreciated the security benefits of initially redirecting HTTP traffic from Gleichauf's computerized device to a quarantine server, rather than a remediation server, as taught by Lewis. For example, as explained in Section VI.E, the combination would have allowed the quarantine server (part of policy server) to re-assess the electronic device's posture credentials before re-directing the request to the appropriate remediation server. The user of the quarantined device would receive information about the quarantine reasons via a web page served by the

quarantine server. In the combination with Lewis, HTTP traffic from a quarantined computerized device, like those described by Gleichauf, would be redirected a quarantine server (QS), as taught by Lewis. The quarantine server (QS) (part of the policy server) would then serve a web page to the user of the computerized device, as further taught by Lewis. The web page would inform the user that their computerized device is in quarantine, and thus, be a "*quarantine notification page*."

161. Therefore, the combination of Gleichauf and Lewis discloses "*serving a quarantine notification page to the first host when the service request comprises a web server request*."

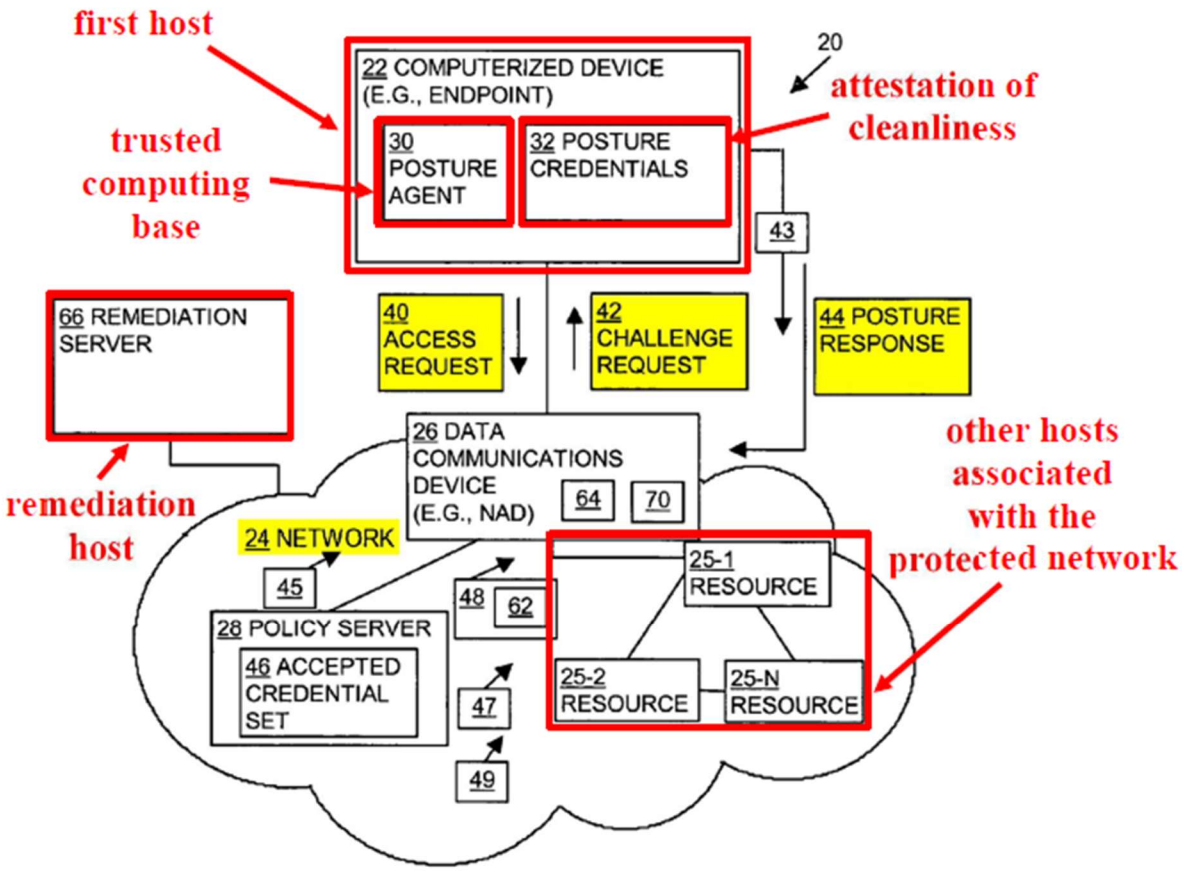
j. [1.9] ***and in the event the service request comprises a DNS query, providing in response an IP address of a quarantine server configured to serve the quarantine notification page if a host name that is the subject of the DNS query is not associated with a remediation host configured to provide data usable to remedy the insecure condition; and***

162. The combination of Gleichauf and Lewis renders this limitation obvious.

163. **First**, as discussed in [1.6], Gleichauf discloses that a computerized device that is out of compliance with network admission policy is quarantined. Gleichauf at 4:50-56. Communications from a quarantined device to protected resources in the network are restricted or rejected. Gleichauf at 11:58-67.

164. **Second**, Gleichauf discloses a remediation server 66 (*i.e.*, "*remediation host*"). 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" (*i.e.*, "*remediation host configured to provide data usable to remedy the insecure condition*"). Gleichauf at 5:4-11.

165. Gleichauf's Figure 1 (below) depicts the remediation server:



Gleichauf at Fig. 1 (annotated).

166. HTTP traffic from the computerized device directed to a remediation server is permitted:

> The access instruction 48 may permit traffic from user device 22 only to certain destination addresses, e.g. IP address of remediation server, and only to certain protocols or ports, e.g. HTTP traffic.

Gleichauf at 14:62-66.

167. HTTP traffic from the computerized device directed to a destination other than the remediation server is redirected:

> For example, assume that when comparing the posture credentials 32 of the user device 22 with the accepted credential set, the policy server 28 detects the presence of an anti-virus application associated with the user device 22 and further detects that the anti-virus application is out-of- date (e.g., the user device does not have the most recent version of the anti-virus application). **Based on such detection, the policy server 28 determines that a redirect access instruction 62 should be applied to HTTP traffic from the user device 22. When the data communications device 26 receives communications traffic from the computerized device 22, the data communications device 26 applies the redirect access**

instruction 62.

Gleichauf at 15:55-67.

168. It would have been obvious to a POSITA that HTTP traffic, such as that from Gleichauf's computerized 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) network address of a named server, which, in the case of HTTP traffic, would be a web server. The user's device then accesses the web page by sending an HTTP request to the DNS-resolved IP address. *See* U.S. Patent No. 7,568,107 at 5:49-63.

169. **Third**, Lewis describes a "hijack DNS component 421 for intercepting DNS queries from quarantined clients." Lewis at 11:15-17. The operation of the hijack DNS component is such that "[w]henever the client performs name resolution on any address, it will receive the IP [address] of QS [quarantine server]." Lewis at 14:22-23. Thus, the IP address of Lewis's quarantine server (QS) (part of the policy server in the combination) *is an IP address of a quarantine server configured to serve the quarantine notification*.

170. **Fourth**, both Gleichauf and Lewis describe allowing a device placed in quarantine to communicate with one or more servers to fix the problem that caused the device to be quarantined. *See, e.g., Gleichauf at 5:4-13; Lewis at 13:12-17,*

*14:17- 3*1. It would have been obvious to a POSITA that communications with such a remediation server would be preceded by a DNS request to resolve the remediation server's name into its IP address. This is because servers are frequently known and referred to by their domain name, but the domain name cannot be directly used to address an IP packet to the server. Instead, the domain name must be resolved, using DNS, into the server's corresponding IP address. The IP address can then be specified as the destination for an IP packet. A POSITA would have been familiar with the use and operation of DNS in networks. Gleichauf describes allowing a computerized device in quarantine to communicate with a remediation server "to upgrade or provide up-to-date patches to applications, such as anti-virus applications." Gleichauf at 16:14-16. Thus, it would have been obvious to a POSITA for a DNS server—responding to the computerized device's request to resolve the IP address of the remediation server—to provide the correct IP address of the remediation server. To achieve this, it would have been obvious to a POSITA to adapt the functionality of a hijack DNS component—like that described in Lewis—to provide correct IP address resolution answers to DNS queries relating to remediation servers. For all other queries, the hijack DNS component would function as described in Lewis, namely, by answering such other DNS queries with the IP address of a quarantine server. Thus, the combined teachings of Gleichauf

and Lewis render obvious responding to a DNS query with the IP address of a quarantine server *if a host name that is the subject of the DNS query is not associated with a remediation host configured to provide data usable to remedy the insecure condition*.

171. As described in more detail in Section VI.E, 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.

172. Therefore, the combination of Gleichauf and Lewis discloses "*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*."

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

173. Gleichauf discloses this limitation.

174. As explained in [1.6], Gleichauf discloses that a computerized device that is out of compliance with network admission policy is quarantined. Gleichauf at 4:50-56. Communications from a quarantined device to restricted resources in the network are restricted or rejected. Gleichauf at 11:58-67.

175. 153. HTTP traffic from the computerized device directed to a remediation server, however, may be permitted (*i.e.*, "*permitting the first host to communicate with the remediation host*"):

> The access instruction 48 **may permit traffic** from user device 22 only **to certain destination addresses**, e.g. IP address of **remediation** server, and only to certain protocols or ports, e.g. HTTP traffic.

Gleichauf at 14:62-66.

176. Therefore, Gleichauf discloses "*permitting the first host to communicate with the remediation host*."

2. Claim 2

a. [2.0] *A method as recited in claim 1,*

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

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

178. Gleichauf discloses this limitation.

179. **First**, as explained in [1.1], Gleichauf teaches that a policy server *detects an insecure condition* in the computerized device 22 by reviewing the

computerized device's posture credentials. Gleichauf at 3:36-45, 11:29-32, 11:58-12:3, 14:16-34, 22:39-59.

180. **Second**, the computerized device's posture response includes posture credentials. The posture credentials contain information about "the last time a local antivirus scan was executed." Gleichauf at 3:36-49. Gleichauf explains that a computerized device that is "not configured with anti-virus or up-to-date anti-virus applications" is "vulnerable." Gleichauf at 22:54-56. Gleichauf also contemplates that the failure to show "timely execution of the anti-virus application" indicates that a computerized device is "vulnerable" and in an insecure condition. Gleichauf at 22:38-59. As a result, a POSITA would have understood that checking for an installed anti-virus application and checking the recency of an anti-virus scan each constitute "*scanning for a vulnerability*."

181. The posture credentials also contain information about whether the device can "resist reception or transmission of… content (spam and/or data theft)." Gleichauf at 8:52-56. A POSITA would have understood that "*scanning*" the device would determine whether the device can, in fact, resist the transmission of content, such as spam. A POSITA would also have known that the term spam refers to a form of "*malicious data*" because spam is unsolicited by the user of the computer and can result in the computer becoming infected with a virus or malware.

182. The posture credentials further contain information about, for example, the presence (or absence) of an anti-virus software, currency of an anti-virus definition, and the status of a firewall application. Gleichauf at 3:36-49; *see also* Gleichauf at 8:62-65 (reporting on the "status of intrusion prevention (e.g., firewall) applications"). A POSITA would have understood that an anti-virus software or firewall application is a type of "*security software*" because it can detect and prevent virus and other security infestations on the device. Reviewing the posture credentials for information about the presence or absence of anti-virus software or firewall application involves "*determining whether a security software is installed*" on the computerized device.

183. Furthermore, Gleichauf notes that its "invention helps 'dampen' the propagation of worms and viruses, and related side effects, by **preventing systems capable of participating in such an outbreak from gaining access to the network**." Gleichauf at 8:28-31. Gleichauf also explains that the "security posture of user device 22 relates, for example, to the user device's 22 ability to resist reception or transmission of malware (e.g., viruses), content (spam and/or data theft), or access by unauthorized users." Gleichauf at 19:64-67. A POSITA would have found it obvious that preventing worm and virus outbreaks—and determining a device's ability to resist malware—involves detecting anomalous network traffic,

such as monitoring the user device's inbound and outbound communications for the reception/transmission of malware.

184. Therefore, Gleichauf discloses "*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. Claim 3

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

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

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

186. Gleichauf renders this limitation obvious.

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

188. **Second**, the computerized device's posture response includes posture credentials gathered by the posture plug-ins. The posture credentials contain information about "operating system patches [that] are installed within the computer system" (*i.e., "updates to an operating system*"):

> **Posture credentials** collected in real time from the posture plug-ins by the posture agent can **identify critical security information** such as what particular virus software and virus definition versions are installed within the computer system, when was the last time a local antivirus scan was executed, **what operating system patches are installed within the computer system**, what particular types and versions of software applications are configured and active within the computer system, and so forth. By providing a security posture based challenge-response mechanism for network admission control, embodiments of this invention are able to dramatically reduce the opportunities for injection of malware into a network.

Gleichauf at 3:36-49.

189. Gleichauf discloses that posture credentials can identify whether the device's operating system makes the device "vulnerable":

> Also assume that a scanning application periodically scans computerized devices connected to LAN (e.g., once a day) for the presence of **known vulnerabilities** in the device **operating system, resident applications, associated security programs,**

and their respective configurations.

Gleichauf at 1:65-2:3.

190. A computerized device will be quarantined if it is out of compliance with the network admission policy. Gleichauf at 4:50-56. A POSITA would have understood that missing operating system patches can make a device vulnerable to various network threats and, as a result, can render the device out of compliance with network admission policy, consistent with Gleichauf's teachings.

191. Once the device is quarantined, the policy server can transmit a notification message to the device "indicating that the device has been quarantined and needs to be remediated." Gleichauf at 21:5-8. The notification message may also "include a link to a remediation server or the IP address of the remediation server 66 that a user may click on or an application on the user device 22 may process to remediate the user device 22 automatically." Gleichauf at 21:8-12. 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." Gleichauf at 5:8-11. The computerized device "**should be admitted into the original network if the remediation process was successful**." Gleichauf at 5:11-13. A POSITA would have found it obvious from Gleichauf's discussion of a device with outdated operating systems being placed into

quarantine and of providing operating system updates, to quarantine a device "*until an update to an operating system has been installed*." This is consistent with Gleichauf's teaching that the computerized device will be admitted into the original network, out of its quarantine, if the remediation process was successful.

192. Therefore, Gleichauf discloses "*wherein detecting an insecure condition includes determining that the first host should be quarantined until an update to an operating system has been installed*."

4. Claim 5

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

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

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

194. As discussed in [1.10], Gleichauf discloses that communications from a quarantined computerized device are restricted or rejected. Gleichauf at 11:58-67. HTTP traffic from the device to a remediation server, however, is permitted. Gleichauf at 14:62-66. Thus, Gleichauf discloses this limitation for the same reasons discussed in [1.10].

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

195. Gleichauf discloses this limitation.

196. A quarantined computerized device can receive a notification message from the policy server "indicating that the device has been quarantined and needs to be remediated." Gleichauf at 21:5-8. The notification message may also "include a link to a remediation server or the IP address of the remediation server 66 that a user may click on or an application on the user device 22 may process to remediate the user device 22 automatically." Gleichauf at 21:8-12. The computerized device responds to the notification message by "**activat[ing] a link, associated with the notification message, to direct the computerized device 22 to the remediation server 66**." Gleichauf at 21:20-23.

197. A POSITA would have understood that activating the link to direct the computerized device to the remediation server includes an "*outbound communication*" to the remediation server. In fact, Gleichauf's data communications device 26 is already "configured to detect attempted connections to the network 24 by the user device 22." Gleichauf at 10:6-11, 12:57-67. Figure 2 of Gleichauf further illustrates that outbound access requests from the electronic device 22 are received at the data communications device 26. Gleichauf at Fig. 2. Since the remediation server is not a part of the computerized device (the "*first host*"), directing the computerized device to the remediation server would be an "*outbound communication from the first host*."

198. A POSITA would have further understood that the network "*detects*" the outbound communication from the first host. The specific access rights of the quarantined device to network resources is administered by the data communications device 26, based on admission instructions 48 provided by the policy server. Gleichauf at 22:38-47. The data communications device 26 receives (i.e., "*detects*") communications from the quarantined device and then either restricts or rejects them, depending on the instructions from the policy server. Gleichauf at 13:52-65, 14:3-7, 14:20-34, 22:38-47.

199. Therefore, Gleichauf discloses "detecting an outbound communication from the first host."

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

200. Gleichauf discloses this limitation.

201. As discussed in [5.2], Gleichauf discloses that the data communications device 26 receives communications from a quarantined device and then either restricts or rejects them, depending on instructions from the policy server. Gleichauf at 22:38-47. HTTP traffic from the quarantined device to a remediation server are generally permitted. Gleichauf at 14:62-66. A POSITA would have understood that permitting HTTP traffic from a device to a remediation server, within Gleichauf's

context, would entail the data communications device 26 "*forwarding the outbound communication*" from the device "*to the remediation host*."

202. Therefore, Gleichauf discloses "*forwarding the outbound communication if it is addressed to the remediation host*."

5. Claim 6

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

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

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

204. As discussed in [1.6], Gleichauf discloses that communications from a quarantined computerized device to network resources are either restricted or rejected. Gleichauf at 11:58-67. Thus, Gleichauf discloses this limitation for the same reasons discussed in [1.6].

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

205. As discussed in [5.2], Gleichauf discloses that the data communications device 26 detects outbound communications from the computerized device to the remediation server. Gleichauf at 21:20-23, 22:38-47. Thus, Gleichauf discloses this limitation for the same reasons discussed in [5.2].

d. [6.3] *redirecting the outbound communication to a quarantine server if it comprises a request for an*

approved service and is not addressed to a remediation host.

206. The combination of Gleichauf and Lewis renders this limitation obvious.

207. **First**, communications from a quarantined device to restricted resources in the network are restricted or rejected. Gleichauf at 11:58-67. Indeed, communications may only be permitted "only to **certain destination addresses**, e.g. IP address of **remediation server**, and only **to certain protocols or ports**, e.g. **HTTP traffic**." Gleichauf at 14:62-66. As such, 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*."

208. This interpretation is consistent with the '705 patent's description that an approved service request can be a request for a web page:

> If outbound traffic is not associated with a remediation request (1403), then, in this example, it is tested to see if it is an approved service request (1405), such as a request for a web page.

'705 patent at 14:62-65. A POSITA would have understood that an HTTP communication constitutes a web page request because HTTP was ubiquitously used for communications between web browsers and web servers, as explained in [1.8].

209. **Second**, as discussed in [1.9], Gleichauf discloses that HTTP traffic ("*approved service*") from the computerized device directed to a remediation server is permitted. Gleichauf at 14:62-66. HTTP traffic ("*approved service*") from the device to other destinations ("*not addressed to a remediation host*") are 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." Gleichauf at 15:55-16:6, *see also* 28:34-40 ("applying the redirect access instruction at the intercepting device by operation of one or more computer processors, to communications traffic originating from the computerized device and destined for resources within the network, in order to grant the computerized device with access to the remediation server without including access to the resources in the network."). It would have been obvious to a POSITA, in view of Lewis's teachings, to *redirect* unauthorized outbound access requests to a *quarantine server* that would inform the user of their device's quarantine status via a quarantine notification page. Lewis at 11:15-17, 14:22-23; *see supra* [1.8]-[1.9].

210. Therefore, the combination of Gleichauf and Lewis discloses "*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*."

6. Claim 7

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

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

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

212. Gleichauf renders this limitation obvious.

213. As explained in [1.6], Gleichauf discloses placing into quarantine a computerized device that is out of compliance with network admission policy. Gleichauf at 4:50-56. Communications from a quarantined device to restricted resources in the network are restricted or rejected, unless they are directed to a remediation server. Gleichauf at 11:58-67, 14:62-66, 28:34-40. In other words, Gleichauf discloses that outbound traffic is not permitted from the quarantined device to a destination that is not the remediation server:

> The access instruction 48 **may permit traffic** from user device 22 only **to certain destination addresses**, e.g. IP address of **remediation server**, and only to certain protocols or ports, e.g. HTTP traffic.

Gleichauf at 14:62-66.

214. It would have been obvious to a POSITA that because a quarantined device is not permitted to transmit a request for information to a non-remediation

server, the quarantined device is effectively prevented from receiving information that it would have received in response to the request. This is because such a request for information is blocked from ever reaching the non-remediation server, and hence the non-remediation server never sends any response. Thus, it would have been obvious to a POSITA that, by preventing a quarantined device from sending traffic (such as an HTTP GET request) to non-remediation servers, Gleichauf's system also prevents the quarantined device from receiving information not related to remediation.

215. To the extent that this claim is interpreted as requiring blocking all non-remediation-related traffic from reaching the quarantined device, such a restriction would have been obvious to a POSITA. Gleichauf describes an example of a quarantine location as an "isolated… network segment." Gleichauf at 4:25-26. Gleichauf also explains that its quarantine is used to "minimiz[e] the risk that the network resources receive or **transmit malware**" from or to "vulnerable computerized devices." Gleichauf at Abstract. Consistent with the concern that a device in quarantine would receive undesirable transmissions such as malware, Gleichauf's security posture (used to determine whether to quarantine, see supra [1.3]-[1.5]) includes information about a device's "**ability to resist reception** or transmission of malware (e.g., viruses), content (spam and/or data theft)…."

Gleichauf at 8:52-55. Because Gleichauf describes the concern that a vulnerable device might receive malware from the network, it would have been obvious to a POSITA that a device placed into 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. A POSITA would have had a reasonable expectation of success in restricting unauthorized communications to or from a device in quarantine because Gleichauf explains that data communications device 26 is responsible for allowing or preventing "communications traffic **between** the user device 22 and the network resources 25." Gleichauf at 14:3-7, 14:20-25.

216. Therefore, Gleichauf renders obvious "*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*."

7. Claim 8

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

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

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

218. Gleichauf discloses this limitation.

219. The computerized device attempts to communicate using the network. Gleichauf at 3:50-56. The network is "a computer network, such as an enterprise

network or the **Internet**, which includes the network resources 25, the data communications device 26, and the policy server 28." Gleichauf at 9:57-64. The network resources 25 are "configured as servers that contain content accessible by the . . . device 22 via network 24." Gleichauf at 9:65-67.

220. A POSITA would have understood that the combination of the data communications device 26 and the policy server 28 is an Internet service provider because they selectively provide access to network 24, which Gleichauf suggests being the Internet. The policy server reviews the computerized device's posture credentials. Gleichauf at 10:30-33, 11:45-58. If the device is found to be out of compliance, it is quarantined by the data communications device in response to instructions from the policy server. Gleichauf at 4:50-56, 11:58-67. While quarantined, the computerized device is unable to communicate with network resources 25. Gleichauf at 11:58-67, 14:62-66. A POSITA would have therefore appreciated that the data communications device and policy server determine whether the computerized device can access network resources on the network, which is the Internet. The data communications device and the policy server together operate to provide the computerized device Internet access, which a POSITA would have understood to be the function of an "*Internet service provider*."

221. Therefore, Gleichauf discloses "*performed at an Internet service provider.*"

8. Claim 9

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

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

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

223. As discussed in [3.1], the computerized device's posture response includes posture credentials gathered by the posture plug-ins. The posture credentials contain information about the presence of "operating system patches [that] are installed within the computer system." Gleichauf at 3:36-49, 8:65-9:1. Thus, Gleichauf discloses this limitation for the same reasons discussed in [3.1].

9. Claim 10

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

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

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

225. As discussed in [1.4], the computerized 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.

Gleichauf at 3:36-49, 8:56-9:5. If the current anti-virus definition is not present, communications from the computerized device to the network will be restricted or completely rejected, showing that the response is deemed not to include "*a valid . . . attestation of cleanliness*." Gleichauf at 11:58-67. Thus, the combination of Gleichauf and Ovadia renders this limitation obvious for the same reasons discussed in [1.4].

10. Claim 11

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

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

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

227. Gleichauf discloses this limitation.

228. The computerized 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. Gleichauf at 3:36-49, 8:56-9:5. If a review of the posture credentials "fails to detect the presence of an anti- virus application (e.g., or **an up-to-date anti-virus application or virus definition file**)," the device will be quarantined. Gleichauf at 14:16-34. A POSITA would have understood that failure to detect an up-to-date anti-virus application or

definition file indicates that the existing anti-virus application or definition file on the computerized device "*is not sufficiently recent*." A POSITA would have understood that updates to anti-virus applications or definition files were often done through patches that updated out-of-date applications or virus definition files. Consequently, if no current, up-to-date anti-virus application or definition file were detected, a POSITA would have found it obvious that the patch level associated with the software component was not sufficiently recent.

229. Therefore, Gleichauf discloses "*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*."

11. Claim 12

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

230. To the extent that the preamble is limiting, Gleichauf discloses it.

231. 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." Gleichauf at 3:4-9. Gleichauf's system, and specifically its policy server, protects the network by reducing the risk of the network becoming infected with malware or suffering from a virus outbreak:

> The policy server receives the challenge responses and, based upon an analysis of the posture credentials of the computerized device, determines a security state of the computerized device and either provides some level of or denies the computerized device access to the network resources based upon the analysis. The data communications system, therefore, detects 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 malware**. In effect, this invention helps "dampen" the propagation of **worms and viruses**, and related side effects, by **preventing systems capable of participating in such an outbreak from gaining access to the network**.

Gleichauf at 8:16-31.

232. Therefore, Gleichauf discloses *"[a] system for protecting a network*," as recited in the preamble.

b. [12.1] *a processor configured to:*

233. Gleichauf discloses this limitation.

234. As discussed in [1.0]-[1.10], Gleichauf's policy server protect the network. The policy server "includes a controller 209 formed of a memory 210 and **a processor 212**." Gleichauf at 25:44-46.

235. Therefore, Gleichauf discloses "*a processor configured to*."

c. [12.2] ***detect an insecure condition on a first host that has connected or is attempting to connect to a protected network, wherein detecting the insecure condition includes***

236. See analysis at [1.1].

d. [12.3] ***contacting a trusted computing base associated with a trusted platform module within the first host,***

237. See analysis at [1.2].

e. [12.4] ***receiving a response, and***

238. See analysis at [1.3].

f. [12.5] ***determining whether the response includes a valid digitally signed attestation of cleanliness,***

239. See analysis at [1.4].

g. [12.6] ***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;***

240. See analysis at [1.5].

h. **[12.7]** ***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, wherein preventing the first host from sending data to one or more other hosts associated with the protected network includes***

241. See analysis at [1.6].

i. **[12.8]** ***receiving a service request sent by the first host,***

242. See analysis at [1.7].

j. **[12.9]** ***serving a quarantine notification page to the first host when the service request comprises a web server request,***

243. See analysis at [1.8].

k. **[12.10]** ***and in the event the service request comprises a DNS query, providing in response an IP address of a quarantine server configured to serve the quarantine notification page if a host name that is the subject of the DNS query is not associated with a remediation host configured to provide data usable to remedy the insecure condition; and***

244. See analysis at [1.9].

l. **[12.11]** ***permit the first host to communicate with the remediation host; and***

245. See analysis at [1.10].

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

246. Gleichauf discloses this limitation.

247. As discussed in [12.1], Gleichauf's policy server operates to protect the network. Gleichauf's policy server "includes a controller 209 formed of a **memory 210** and a processor 212." Gleichauf at 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.

248. Memory 210 is "*configured to provide instructions to the processor*" of the policy server because it is "**encoded with logic instructions (e.g., software code) and/or data** that form a credential analysis application 246 configured according to embodiments of the invention. In other words, the credential analysis application 246 represents **software code, instructions and/or data . . . that reside within memory** or storage or within any computer readable medium accessible to the policy server 28." Gleichauf at 26:66-27:8.

249. Therefore, Gleichauf discloses "*a processor configured to*."

12. Claim 13

a. [13.0] ***A system as recited in claim 12,***

250. See analysis at [12.0]-[12.10].

b. [13.1] ***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.***

251. See analysis at [2.1].

13. Claim 15

a. [15.0] ***A system as recited in claim 12,***

252. See analysis at [12.0]-[12.10].

b. [15.1] ***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.***

253. See analysis at [7.1].

14. Claim 16

a. [16.0] ***A system as recited in claim 12,***

254. See analysis at [12.0]-[12.10]

b. [16.1] ***wherein the software component on the first host is an operating system.***

255. See analysis at [9.1].

15. Claim 17

a. [17.0] ***A system as recited in claim 12,***

256. See analysis at [12.0]-[12.10].

b. [17.1] ***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.***

257. See analysis at [10.1].

16. Claim 18

a. [18.0] ***A system as recited in claim 17,***

258. See analysis at [12.0]-[12.10] and [17.0]-[17.1].

b. [18.1] ***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.***

259. See analysis at [11.1].

17. Claim 19

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

260. To the extent that the preamble is limiting, Gleichauf discloses it.

261. Gleichauf teaches a "**computer program product 200** includes an application or logic instructions that are loaded into the computerized device 22, data communications device 26, and policy server 28 to configure the devices 22, 24, 26 to operate as part of the data communications system 20" described throughout the reference. Gleichauf at 25:46-51; *see also* Gleichauf at 3:4-9 (disclosing 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."). A POSITA would have understood that product 200 "*comprises computer instructions*."

262. As described in [12.12], Gleichauf's policy server operates to protect the network. The policy server includes "*non-transitory computer readable medium*" in the form of memory 210 that stores the "**logic instructions (e.g., software code) and/or data**" necessary to operate the disclosed security architecture. Gleichauf at 26:66-27:8.

263. Gleichauf's system protects the network by reducing the risk of the network becoming infected with malware or suffering from a virus outbreak:

> The policy server receives the challenge responses and, based upon an analysis of the posture credentials of the computerized device, determines a security state of the computerized device and either provides some level of or denies the computerized device access to the network resources based upon the analysis. The data communications system, therefore, detects 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 malware**. In effect, this invention helps "dampen" the propagation of **worms and**

> **viruses**, and related side effects, by **preventing systems capable of participating in such an outbreak from gaining access to the network**.

Gleichauf at 8:16-31.

264. Therefore, Gleichauf discloses *"[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*," as recited in the preamble.

a. [19.1] ***detecting an insecure condition on a first host that has connected or is attempting to connect to a protected network, wherein detecting the insecure condition includes***

265. See analysis at [1.1].

b. [19.2] ***contacting a trusted computing base associated with a trusted platform module within the first host,***

266. See analysis at [1.2].

c. [19.3] ***receiving a response, and***

267. See analysis at [1.3].

d. [19.4] ***determining whether the response includes a valid digitally signed attestation of cleanliness,***

268. See analysis at [1.4].

e. [19.5] ***wherein the valid digitally signed attestation of cleanliness includes at least one of an attestation that the trusted computing base has ascertained that the first host***

is not infested, and an attestation that the trusted computing base has ascertained the presence of a patch or a patch level associated with a software component on the first host;

269. See analysis at [1.5].

f. [19.6] ***when it is determined that the response does not include a valid digitally signed attestation of cleanliness, quarantining the first host, including by preventing the first host from sending data to one or more other hosts associated with the protected network, wherein preventing the first host from sending data to one or more other hosts associated with the protected network includes***

270. See analysis at [1.6].

g. [19.7] ***receiving a service request sent by the first host,***

271. See analysis at [1.7].

h. [19.8] ***serving a quarantine notification page to the first host when the service request comprises a web server request,***

272. See analysis at [1.8].

i. [19.9] ***and in the event the service request comprises a DNS query, providing in response an IP address of a quarantine server configured to serve the quarantine notification page if a host name that is the subject of the DNS query is not associated with a remediation host configured to provide data usable to remedy the insecure condition; and***

273. See analysis at [1.9].

j. [19.10] ***permitting the first host to communicate with the remediation host.***

274. See analysis at [1.10].

VII. CONCLUSION

275. I declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true, and that these statements were made with knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under section 1001 of Title 18 of the United States Code

A. C. Narasimha Reddy

Dated: 13th Aug. 2025 _____

Narasimha Reddy, Ph.D.