UNITED STATES PATENT AND TRADEMARK OFFICE

_____

BEFORE THE PATENT TRIAL AND APPEAL BOARD

_____

CISCO SYSTEMS, INC.,
Petitioner

U.S. Patent No. 10,601,780

_____

DECLARATION OF DR. PRASHANT SHENOY, UNDER 37 C.F.R. § 1.68 IN SUPPORT OF PETITION FOR *INTER PARTES* REVIEW

TABLE OF CONTENTS

I. Introduction....4

II. Qualifications and Professional Experience....6

III. Level of Ordinary Skill in the Art....10

IV. Relevant Legal Standards....11

V. Background....12

A. Virtual Machines....12

B. Firewalls....13

VI. Overview of the '780 Patent....14

A. Summary of the '780 patent....14

B. Prosecution History of the '780 patent....17

VII. Claim Construction....17

VIII. Identification of how the Claims are Unpatentable....18

A. Ground 1: Claims 1-7, 9-17, and 19-20 are obvious in view of Delco and Adams....18

1. Summary of Delco....18

2. Summary of Adams....22

3. Reasons to Combine Delco and Adams....24

4. Claim 1....26

5. Claim 2....57

6. Claim 3....59

7. Claim 4....62

8. Claim 5....64

9. Claim 6 71
10. Claim 7 72
11. Claim 9 74
12. Claim 10 76
13. Claim 11 79
14. Claim 12 81
15. Claim 13 82
16. Claim 14 82
17. Claim 15 83
18. Claim 16 83
19. Claim 17 84
20. Claim 19 84
21. Claim 20 84

B. Ground 2: Claims 8 and 18 are obvious in view of Delco, Adams, and Dadhia. 85

1. Summary of Dadhia. 85
2. Reasons to Combine Dadhia with Delco and Adams 86
3. Claim 8 90
4. Claim 18 100

IX. Conclusion 102

I, Prashant Shenoy, do hereby declare as follows:

I. INTRODUCTION

1. I am making this declaration at the request of Cisco Systems, Inc. in the matter of the *Inter Partes* Review of U.S. Patent No. 10,601,780 ("the '780 patent") to Hoy *et al*.

2. I am being compensated for my work in this matter at my usual hourly rate for this work. I am also being reimbursed for reasonable and customary expenses associated with my work and testimony in this investigation. 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 claims 1-20 ("the Challenged Claims") of the '780 patent are unpatentable as they 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 all of the limitations of the Challenged Claims would have been obvious to a POSITA.

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

a. the '780 patent, Ex.1001;

b. the prosecution history of the '780 patent ("'780 File History"), Ex.1002;

c. U.S. Patent No. 8,166,474 to Delco et al. ("Delco"), Ex.1005;

d. U.S. Patent Publication No. 2010/0077476 to Adams ("Adams"); Ex.1006; and

e. U.S. Patent No. 7,886,351 to Dadhia et al. ("Dadhia"), 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 any additional authoritative documents as cited in the body of this declaration; and

my own knowledge and experience based upon my work in the field of optical communication as described below, as well as the following materials:

a. U.S. Patent Publication No. 2006/0262916 to Marascio et al. ("Marascio"), Ex.1008;

b. U.S. Patent No. 8,468,256 to McArdle ("McArdle"), Ex.1009;

c. U.S. Patent Publication No. 20080267387A1 to Strathmeyer ("Strathmeyer"), Ex.1010;

d. U.S Patent Publication No. 2009/0241192 to Thomas et al. ("Thomas"), Ex.1014;

e. RFC 2131, Ex.1015.

6. Unless otherwise noted, all emphasis in any quoted material has been

added.

II. QUALIFICATIONS AND PROFESSIONAL EXPERIENCE

7. My 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.

8. I have more than two decades of experience in distributed operating systems, computer networking, and data management. During that time, I have published over 300 technical papers that have been cited in other publications more than 25,000 times.

9. I am a tenured professor in the College of Information and Computer Sciences at the University of Massachusetts Amherst ("UMass"), where I currently hold the position of Distinguished Professor of Computer Science. I am also an Associate Dean in the College of Information and Computer Sciences. I currently work in both these capacities. I have been a faculty member at UMass since 1998.

10. I received my Bachelor of Technology (B.Tech.) degree in 1993 from the Department of Computer Science and Engineering at The Indian Institute of Technology ("IIT"), Mumbai, India. IIT awarded me the Institute Medal for being the top ranking undergraduate student. In 1994, I received my Master of Science (M.S.) degree in Computer Sciences from The University of Texas at Austin. In 1998, I received my Doctor of Philosophy (Ph.D.) also from the Department of

Computer Sciences at The University of Texas at Austin. My dissertation, which was in the area of file systems, was recognized by the department as the Best Doctoral Dissertation of 1998-1999.

11. I have been involved in the research and development of distributed computing systems for more than 30 years. I have been a faculty member at the UMass since 1998 where I perform research and teach in the areas of distributed systems, operating systems, computer networking, data management, and multimedia systems, among others.

12. A major focus of my research has been the design of large-scale distributed computing systems. Over the years, I have led research and industry collaboration projects in areas such as computer networking, distributed systems, storage and file systems, Web and Internet systems, content distribution, streaming media, database systems, cloud computing, virtualization, RFID and sensor networks, and energy systems. During that time, I have consulted with both large companies and startups in a technical advisory role related to product architecture, IP issues, and applied research.

13. For example, starting in the early 1990s, my research focused on the development of storage and file systems. During that time, I developed multimedia file system and data management technologies that were designed for efficient storage and retrieval of large amounts of video and audio data. I collaborated with

AT&T Bell Laboratories to design a new media server for video streaming. These efforts led to resulted in U.S. Patent No. 6,079,028 on fault-tolerant architectures for continuous media servers.

14. In the late 1990s, I led the design of a new operating system called QLinux, that was based on the open-source Linux operating system. As part of this research, I developed resource management and scheduling techniques that provide isolation and service guarantees to applications. This technology provided the underpinnings for fair resource management and isolation techniques used in OS virtualization. It was adopted by a startup company for providing resource isolation in its web hosting products.

15. In the early 2000s, I led several projects on web systems and cloud computing. For example, I led research projects on web technologies such as web caching and scale web applications. I also led research projects on cloud computing platforms (which were known as web hosting platforms at the time) involving virtualization, dynamic resource management, replication, and storage. My group at UMass collaborated with technology companies such as IBM, AT&T, Sprint, Intel, and many others as part of this research.

16. My research on virtualization has involved design of technologies for virtual machines and container resource management and their deployment in cloud computing systems and data centers. I am co-inventor on US patent 8,473,557,

jointly assigned to AT&T and UMass, on virtual machine migration.

17. Over the years, I have authored or co-authored more than 300 articles in peer-reviewed journals and conference proceedings that have been cited over 25,000 times. Seventeen papers authored by myself and members of my research group on this project won awards for their technical contributions. Two papers have won Test-of-Time awards, an award given to a publication that has had the most impact over a ten year period since its publication.

18. I have served as an Editor for six leading journals and have served as the Technical Program Chair for twenty conferences and workshops in my field. I founded the ACM Special Interest Group on Energy Systems and also served on steering committee of conferences and journals.

19. I am also a Fellow of the ACM, IEEE, AAAS, and AAIA, four professional societies in computing and science.

20. I am a named inventor on four U.S. Patents (U.S. Patent Nos. 2014/0040206, 8,510,596, 8,473,557, 6,079,028. These patents are generally related to distributed and operating systems, virtualization, and cloud data centers.

21. As a faculty member at UMass, I have also taught undergraduate and graduate courses in operating systems, distributed systems, networking, and related topics for more than twenty years.

22. These courses teach concepts related to operating systems, virtual

machines, containers, cloud computing, and distributed systems. They also involve hands-on course projects on these topics.

III. LEVEL OF ORDINARY SKILL IN THE ART

23. 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.

24. A person of ordinary skill in the art ("POSITA") in the field of the '780 patent, as of its earliest possible filing date of January 27, 2011, would have been someone knowledgeable and familiar with virtualization technologies. A POSITA would have been familiar with virtual machines, guest operating systems, and network security technology that existed as of the early 2010s. That person would have a bachelor's degree in electrical engineering, computer science, or equivalent training, and approximately two years of experience working in the field of virtual computing systems. Lack of work experience can be remedied by additional education, and vice versa.

25. For purposes of this Declaration, in general, and unless otherwise noted, my statements and opinions, such as those regarding my experience and the understanding of a POSITA generally (and specifically related to the references I

consulted herein), reflect the knowledge that existed in the field as of the priority date of the '780 patent. Unless otherwise stated, when I provide my understanding and analysis below, it is consistent with the level of a POSITA prior to the priority date of the '780 patent.

IV. RELEVANT LEGAL STANDARDS

26. I am not an attorney. In preparing and expressing my opinions and considering the subject matter of the '780 patent, I am relying on certain basic legal principles that counsel have explained to me. These principles are discussed below.

27. I understand that prior art to the '780 patent includes patents and printed publications in the relevant art that predate the priority date of the alleged invention recited in the '780 patent. For purposes of this Declaration, I am applying January 27, 2011 as the earliest possible priority date of the '780 patent.

28. I have been informed that a claimed invention is unpatentable under 35 U.S.C. § 103 if the differences between the 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 person having ordinary skill in the art to which the subject matter pertains. I have also been informed by counsel that the obviousness analysis takes into account 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.

29. I have been informed by counsel that the Supreme Court has recognized several rationales for combining references or modifying a reference to show obviousness of claimed subject matter. Some of these rationales include the following: (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 one of ordinary skill to modify the prior art reference or to combine prior art reference teachings to arrive at the claimed invention.

V. BACKGROUND

A. Virtual Machines

30. Virtualization provides the "ability to concurrently support multiple operating system environments on a single physical computer system hardware platform." Delco, 1:18-20. "Each operating system environment, typically referred to as a virtual machine, logically encapsulates a separate instance of an operating

system and defines an execution space within which the operating system manages the execution of programs including user and server level applications." Delco, 1:20-23.

31. Virtual machines provide various advantages. "A principal advantage embodied by virtualization architectures is the ability to establish and enforce isolation between the multiple virtual machines that are concurrently hosted on a single, physical computer system." Delco, 1:37-40. "Programs executed in one virtual machine are essentially unaffected by and, conversely, essentially unable to affect the execution of programs in other virtual machines." Delco, 1:40-42.

B. Firewalls

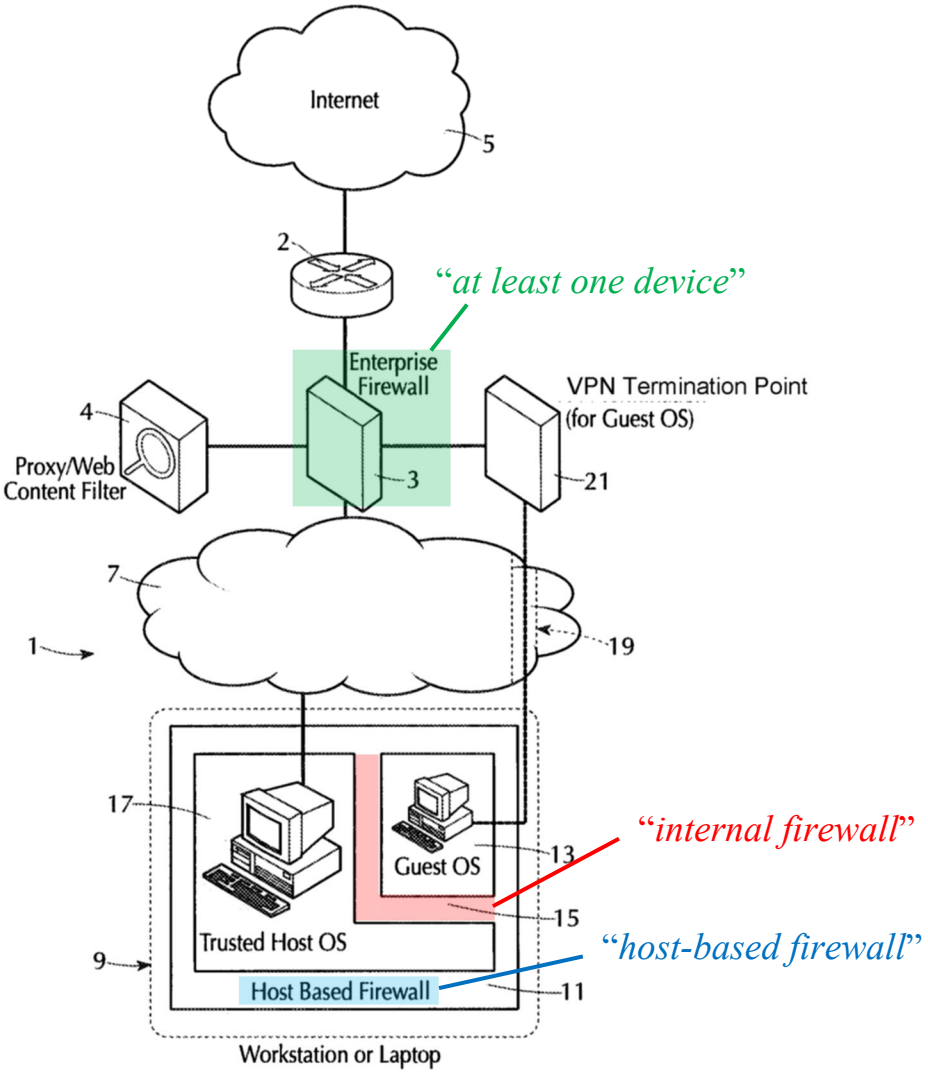
32. Firewalls are used to protect computer systems or networks from external threats. As an example, a "firewall system 30 is conventionally a network appliance implementing any or all of a variety of packet inspection and control functions." Delco, 4:65-67. Firewalls are sometimes referred to as packet filters. Packet filters may be embodied as "packet filtering packages, such as exemplified by the conventional IPTables and IPChains packages, content filters, such as the well-known SpamAssassin and Clam AntiVirus (ClamAV) packages, deep packet analysis and scrubbing packages, and other packages implementing network packet security functions." Delco, 5:2-7.

VI. OVERVIEW OF THE '780 PATENT

A. Summary of the '780 patent

33. The '780 patent describes and claims nothing more than well-known techniques to provide Internet security. The '780 patent relates to "the protection of computer systems from injurious software that can be encountered while browsing or accessing the Internet." Ex.1001, 1:17-29. To provide such protection, the '780 patent describes an environment that includes a "host computer [that] supports a virtual guest system running thereon." Ex.1001, abstract.

34. The purported novelty of the '780 patent—as described in the prosecution history—is "**three distinct methods of isolation** through a combination of an **internal firewall, a host-based firewall, and a network firewall** or web proxy as claimed." Ex.1002, 370. The independent claims recite (1) "***an internal firewall*** *is configured to separate the host system from the virtual system in the computer system*," (2) "***a host-based firewall*** *executed on the computer system is configured to implement network isolation between the computer system and the network*," and (3) "***at least one device*** *configured to implement at least one of a network firewall or a web proxy*." Ex.1001, Claim 1. These three distinct methods of isolation are illustrated in Fig. 1 below.



Ex.1001, Fig. 1 (annotated).

35. These isolation methods protect a host computer system such as a workstation. "The workstation 9 can be virtually any host computer system, including a PC or a laptop or any of a variety of other computer systems with electronically accessible data storage, a user visible display, and input devices such as a keyboard and a mouse." Ex.1001, 7:25-30. The host system includes "an operating system that allows the host system 9 to function, e.g., a Windows or

Linux operating system, as is well known in the art." Ex.1001, 7:32-35. The host system "hosts" a guest operating system 13. "The guest OS pseudo machine or virtual guest system 13 is has its own virtual architecture and configuration." Ex.1001, 8:36-37.

36. As described above, multiple mechanisms are used to protect the host system from malware and other threats. First, the host system is protected from any potential malware on the virtual machine through the "internal firewall." "[T]he hypervisor also provides an additional **internal host-supported firewall** indicated at 15 that separates and restricts interaction between virtual guest system 13 and the trusted-host operating system 17." Ex.1001, 8:13-17. Second, "[t]he host system 9 also has stored data in the form of software instructions that cause the host system to have **a host-based firewall 11** that provides restrictive egress from the computer 9." Ex.1001, 7:42-45. Third, "[a] proxy/web-content filter 4 is connected with and cooperates with the firewall hardware 3 to control the type of data permitted to pass from the Internet 5 into the local system 1, filtering e.g., pornographic material or data not intended for the system 1." Ex.1001, 7:10-14.

37. However, as will be explained in detail below, the arrangement in which three methods of isolation are used to protect a computing system was already known at the time the '780 patent was filed.

B. Prosecution History of the '780 patent.

38. The '780 patent was filed March 2, 2018 and ultimately claims priority to a provisional application filed January 27, 2011. In response to a rejection by the Examiner, the Applicant then argued without amendment that "none of the asserted references, either alone or in combination, teach or suggest **three distinct methods of isolation** through a combination of an **internal firewall, a host-based firewall, and a network firewall** or web proxy as claimed." Ex.1002, 370. The office then allowed the claims and the patent issued on March 24, 2020.

39. However, as evidenced by Delco and as outlined below, it was already known to use "three distinct methods of isolation" using an "internal firewall, a host-based firewall, and a network firewall" as described and claimed in the '780 patent.

VII. CLAIM CONSTRUCTION

40. It is my understanding that in order to properly evaluate the '780 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 be understood by one of ordinary skill in the art in light of the specification and prosecution history, unless the inventor has set

forth a special meaning for a term. I do not believe any of the terms require express construction.

VIII. IDENTIFICATION OF HOW THE CLAIMS ARE UNPATENTABLE

41. I have been asked to provide my opinion as to whether the Challenged Claims of the ’780 patent would have been obvious in view of the prior art. The discussion below provides a detailed analysis of how the prior art references identified below teach the limitations of the Challenged Claims of the ’780 patent.

42. As part of my analysis, I have considered the scope and content of the prior art and any differences between the alleged invention and the prior art. I describe in detail below the scope and content of the prior art, as well as any differences between the alleged invention and the prior art, on an element-by-element basis for each Challenged Claims of the ’780 patent.

43. As described in detail below, the alleged invention of the Challenged Claims would have been obvious in view of the teachings of the identified prior art references as well as the knowledge of a POSITA.

A. Ground 1: Claims 1-7, 9-17, and 19-20 are obvious in view of Delco and Adams.

1. Summary of Delco.

44. Delco, like the ’780 patent, describes an environment in which a host system supports a number of virtual machines. “A host system platform 12 includes a local processor 14 and memory 16 providing a program execution

space." Delco, 4:55-58. "[T]he host system platform 12 further supports the execution of guest operating systems, here represented as virtual machines 26, 28, the individual guest operating system environments." Delco, 5:18-21.

45. Also, like the '780 patent, Delco's system uses three distinct mechanisms for providing isolation. Delco describes (1) an internal firewall that separates the virtual machine from the host system, (2) a host-based packet filter application 32, and (3) an external firewall system 30.

46. The internal firewall that separates the virtual machines and guest operating systems from the host system is illustrated in Fig. 2 below.

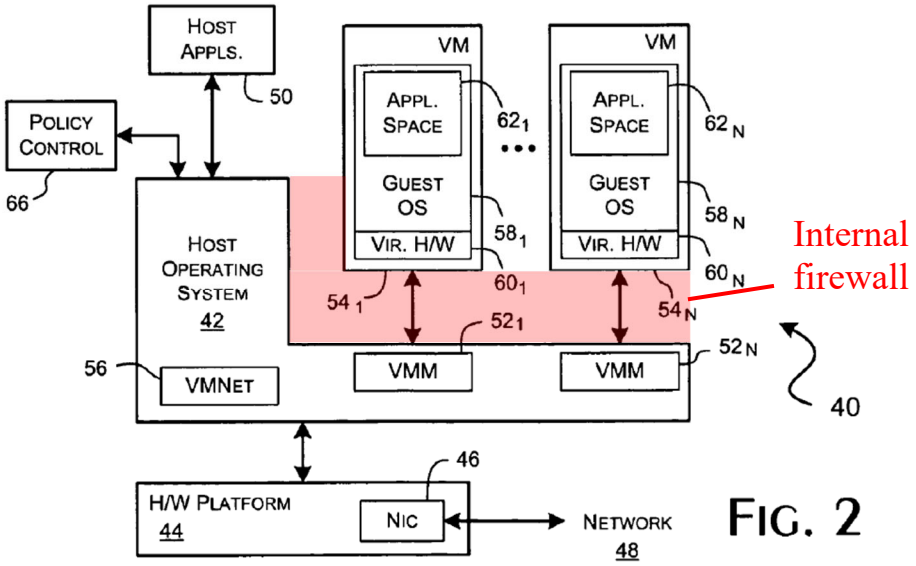
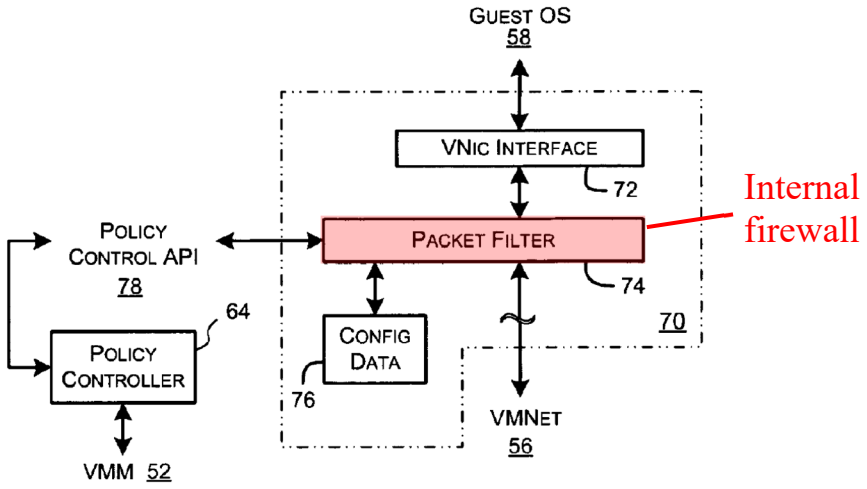


FIG. 2

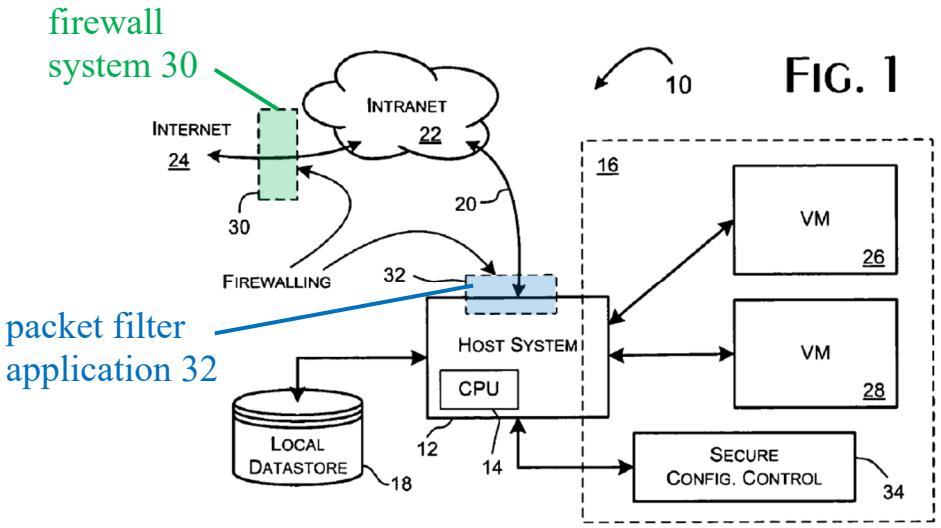
Delco, Fig. 2 (annotated).



Delco, Fig. 3 (annotated).

47. Delco describes separation between the guest operating systems and the host operating systems. For example, Delco describes a packet filter 74: "Network access, though only subject to the specific restrictions and controls **established by the corresponding packet filter 74**, is then available 108 to the guest operating system 58 1-N."

48. The external firewall 30 and host-based packet filter application 32 are illustrated in Fig. 1 below.



Delco, Fig. 1 (annotated).

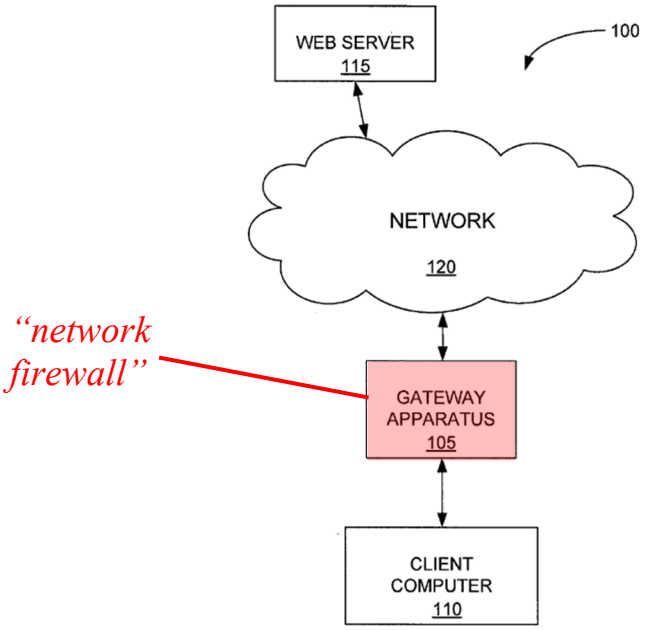
49. Delco's host system is connected to a network (intranet 22) that is protected by an external firewall 30. "[A]n independent firewall system 30 can be installed generally as shown to provide network protection to the intranet 22 including the host system platform 12. The firewall system 30 is conventionally a network appliance implementing any or all of a variety of packet inspection and control functions." Delco, 4:62-65. Delco's host system also includes its own firewall. "Alternately, though more typically in conjunction, a packet filter application 32 is conventionally implemented and executed within the program execution space of the host system platform 12." Delco, 5:8-10.

50. Accordingly, Delco provides evidence that using three distinct

mechanisms of isolation as described and claimed in the '780 patent was already known.

2. Summary of Adams.

51. Adams, like Delco and the '780 patent, relates to network security. Adams describes a "gateway apparatus 105" in Fig. 1, reproduced below.



Adams, Fig. 1 (annotated).

52. Adams' "gateway apparatus 105" may be any of "a Web proxy, a router, and a firewall appliance." Adams, [0021]. Like Delco's firewall system 30 in Delco is positioned between the internet 24 and the host system, the gateway apparatus 105 is positioned between the network 120 "which, in some embodiments, includes the Internet" and the client computer 110. Adams, [0021].

53. Adams also provides more details about the structure of the external

firewall. For example, Adams, Fig. 2 is "a functional block diagram of one type of gateway apparatus" such as the gateway apparatus 105 of Fig. 1. The gateway apparatus can include a "processor 205" and a "memory 235" as show in Fig. 2 below.

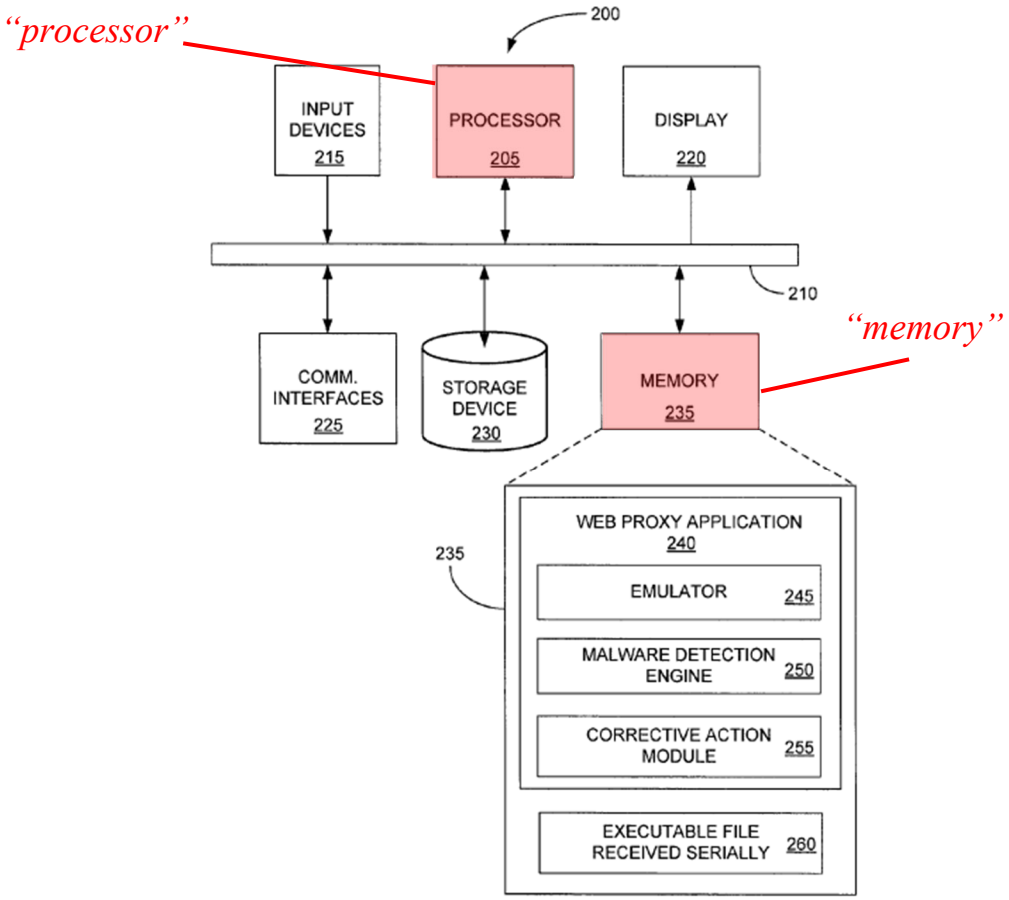


FIG. 2

Adams, Fig. 2 (annotated).

54. Adams explains that its gateway apparatus includes a web proxy application stored in memory. That application isolates the client computer 110 from the network 120. For example, the gateway "tests [executable files] for the

presence of malware." Adams, [0027]. If a file contains malware, the gateway "prevent[s] the executable file 260 from propagating further over network 120. That is, corrective action module 255 can block an executable file 260 that includes malware, thereby preventing it from reaching a client computer 110." Adams, [0033].

3. Reasons to Combine Delco and Adams.

55. A POSITA would have found it obvious for Delco's external firewall system 30 to include a processor and a memory as taught by Adams, and to perform network isolation between the Internet (an untrusted network) and Delco's networked system.

56. Delco describes "an independent firewall system 30 ... to provide network protection to the intranet 22 including the host system platform 12. The firewall system 30 is conventionally a network appliance implementing any or all of a variety of packet inspection and control functions." Delco, 4:62-65. While Delco does not describe the structural details of the firewall system 30, a POSITA would have found it obvious for such a "network appliance" to include a processor and memory. This is evidenced, for example, by Adams.

57. Adams describes a gateway apparatus 105, which like Delco's firewall system 30, is a "firewall appliance." Adams, [0021]. Adams provides additional structural details about the firewall appliance. Adams provides "a

functional block diagram of one type of gateway apparatus" such as the gateway apparatus 105 of Fig. 1. The gateway apparatus can include a "processor 205" and a "memory 235" as show in Fig. 2. Thus, Adams provides evidence that it would be obvious for a firewall network appliance such as Delco's to include a processor and a memory to perform its intended functions.

58. A POSITA would have had a reasonable expectation of success in combining the teachings of Delco and Adams because Adams' gateway apparatus provides firewall functionality like that provided by Delco's firewall system 30. Adams explains that "[i]n the computer industry, a firewall commonly refers to a device, set of devices, and/or software/firmware configured to permit or deny, encrypt, decrypt, or proxy all network traffic between different security domains in accordance with a set of rules or other criteria." Adams, [0036]. Delco similarly explains that its firewall acts as a packet filter "such as exemplified by the conventional IPTables and IPChains packages, content filters, such as the well-known SpamAssassin and Clam AntiVirus (ClamAV) packages, deep packet analysis and scrubbing packages, and other packages implementing network packet security functions." Delco, 5:2-7. A POSITA would have recognized these "packages" as software that were commonly stored in computer memory and executed by a computer processor. Delco, 2:5-6.

59. Accordingly, the combination of Delco and Adams achieves the

predictable result of using a processor and memory to perform the described firewall functions. Adams explains that "Web proxy application 240 and its functional modules shown in FIG. 2 are implemented as software that is executed by processor 205. Such software may be stored, prior to its being loaded into RAM for execution by processor 205, on any suitable computer-readable storage medium such as a hard disk drive, an optical disk, or a flash memory (see storage device 230)." Adams, [0028].

60. Accordingly, the combination of Delco and Adams merely represents applying a known technique (implementing a network appliance with a processor and memory as taught by Adams) to a known device (Delco's firewall appliance) ready for improvement to yield predictable results. The combination of Delco and Adams is also merely the combination of prior art elements (a processor and memory with a firewall appliance) according to known methods to yield predictable results (executing code from memory on the processor to perform firewall functions).

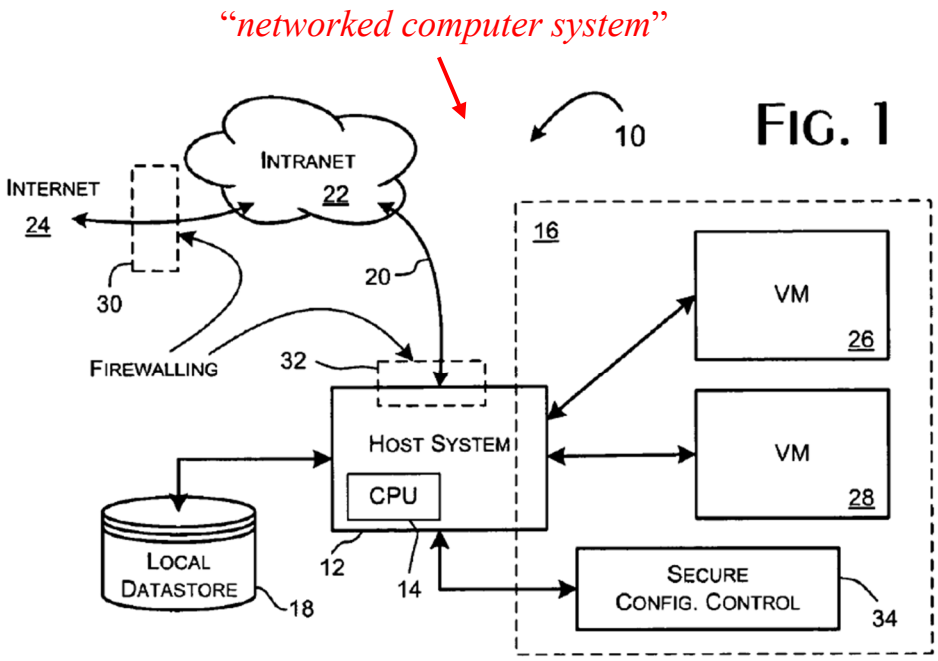
4. Claim 1

[1.0] ***A networked computer system comprising:***

61. To the extent the preamble is limiting, it is rendered obvious by Delco.

62. Delco's Fig. 1 illustrates what it refers to as a "preferred environment"

of its invention, which includes a computer system 10 ("*networked computer system*").



Delco, Fig. 1 (annotated).

> **A computer system 10**, demonstrating **a preferred environment for implementation of the present invention**, is shown in FIG. 1. A host system platform 12 includes a local processor 14 and memory 16 providing a program execution space. The host system platform 12 preferably supports a local datastore 18 and a network interface 20 permitting connection to a local intranet 22, as shown, and further, or directly in alternate scenarios, to a public communications network 24, such as the Internet.

Delco, 4:53-61.

63. Delco's computer system 10 is a "*networked computer system*" as

claimed. As shown in Fig. 1, Delco's computer system 10 includes a "host system platform 12" (Delco, 4:55) connected to a "local intranet 22" (Delco, 4:58-59). Delco's computing system 10 is thus *networked* because it includes a host system 12 that is connected to the local intranet 22.

64. Thus, because Delco describes a computer system 10 that includes a host system connected to the local intranet 22, Delco renders obvious a "*networked computer system*" as claimed.

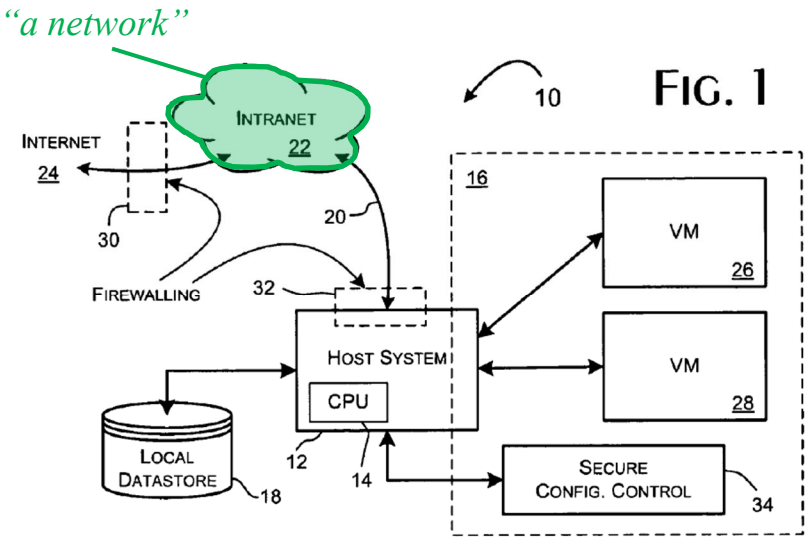
[1.1] ***a network;***

65. Delco's computer system 10 includes an intranet 22 ("*network*") that connects the host system 12 to the Internet 24.

> **A computer system 10**, demonstrating a preferred environment for implementation of the present invention, is shown in FIG. 1. A host system platform 12 includes a local processor 14 and memory 16 providing a program execution space. **The host system platform 12 preferably supports** a local datastore 18 and **a network interface 20 permitting connection to a local intranet 22, as shown, and further, or directly in alternate scenarios, to a public communications network 24, such as the Internet**.

Delco, 4:53-61.

66. The local intranet 22 is illustrated in Delco, Fig. 1, shown below and highlighted in green.



Delco, Fig. 1 (annotated).

67. Thus, because Delco describes a computing system 10 that includes a local intranet, Delco renders obvious "*a network*" as claimed.

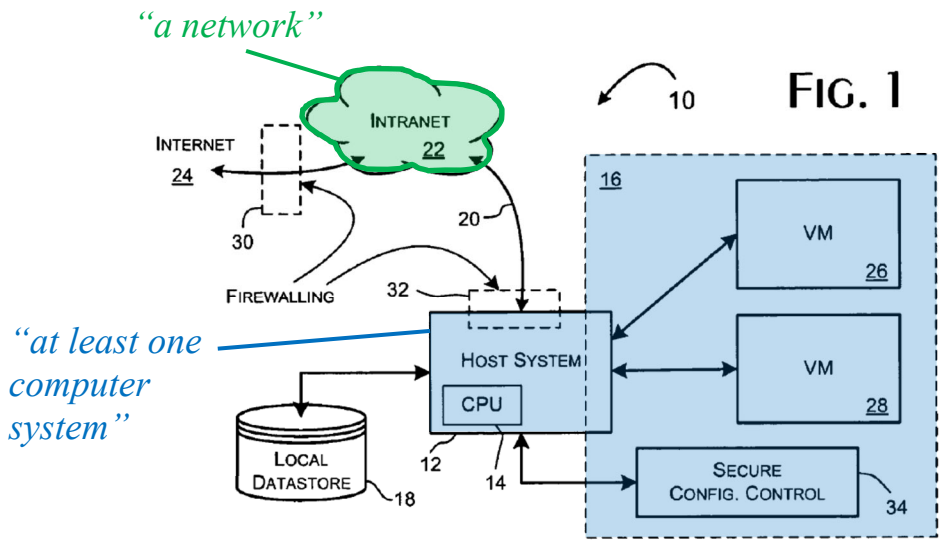
[1.2] ***at least one computer system configured to connect to the network,***

68. **First**, Delco's system 10 includes a host system platform 12 ("*at least one computer system*").

> A computer system 10, demonstrating a preferred environment for implementation of the present invention, is shown in FIG. 1. **A host system platform 12 includes a local processor 14 and memory 16 providing a program execution space**. The host system platform 12 preferably supports a local datastore 18 and a network interface 20 permitting connection to a local intranet 22, as shown, and further, or directly in alternate scenarios, to a public communications network 24, such as the Internet.

Delco, 4:53-61.

69. Regarding the description accompanying Fig. 1, Delco describes "a host system platform 12" with "a local processor 14 and a memory 16 providing a program execution space." Delco, 4:55-56. As relevant to limitation [1.3] *infra*, "the host system platform 12 further supports the execution of guest operating systems, here represented as virtual machines 26, 28." Delco, 5:18-20. Accordingly, the host processing system and its supported virtual machines together teach "*at least one computer system*" as highlighted in Fig. 1 below.



Delco, Fig. 1 (annotated).

70. While Fig. 1 illustrates the host system platform 12 within the "preferred environment" of the invention, Fig. 2 illustrates an example embodiment of the host system platform 12 in which a host operating system is executed on a hardware platform 44. Thus, a POSITA would have naturally read Delco's disclosures relating to Figures 1 and 2 as overlapping and intended to be

integrated and understood together.

> **A preferred embodiment of the present invention implemented in the context of a hosted virtualization framework 40 is shown in FIG. 2. A host operating system 42 is executed on a hardware platform 44 including a network interface controller 46**. The hardware platform 44 also preferably includes a central processor 14 and a local datastore 18. The network interface controller 46 is coupleable to an external communications network 48. The host operating system 42 supports the execution of any number of conventional applications 50 and the components of the virtualization framework, which include instances of a virtual machine monitor (VMM) $52_{1\text{-}N}$ provided to support a set of one or more virtual machines (VM) $54_{1\text{-}N}$ and a virtual network controller (VMNet) 56 that provides coordinated routing of network traffic between the virtual machines $54_{1\text{-}N}$ and the various conventional interface points of a conventional network protocol stack, including a hardware specific network interface controller 46 driver, as incorporated within the host operating system 42.

Delco, 6:13-30.

71. The hardware platform 44 in Fig. 2 is an implementation of the host system platform 12 in Fig. 1. Indeed, Delco explains that "The hardware platform 44 also preferably includes a central processor 14 and a local datastore 18 [shown in Fig. 1]." Delco, 6:16-18. The "local datastore 18," however, is depicted only in Figure 1 and not in Figure 2. This further confirms a POSITA's understanding that

the hardware platform 44 and the systems it supports are an implementation of the host system platform 12 described with respect to Fig. 1. Thus, a POSITA would have understood that the hardware platform 44 and the systems it supports are an implementation of the host system platform 12 described with respect to Fig. 1. It also would have been obvious to a POSITA to combine features of Delco's Fig. 1 and Fig. 2 disclosures. Accordingly, my analysis that refers to teachings from both Fig. 1 and Fig. 2 relates to a single implementation of Delco's Fig. 2 system within the Fig. 1 environment.

72. Delco provides more details of the hardware platform 44 as shown in Fig. 2. The text accompanying Fig. 2 describes that the hardware platform 44 supports a host operating system 42 and virtual machines $54_{1\text{-}N}$. "[H]ost operating system 42" (Delco, 6:15) executes "instances of a virtual machine monitor (VMM) $52_{1\text{-}N}$ provided to support a set of one or more virtual machines (VM) $54_{1\text{-}N}$" (Delco, 6:23-25). As explained above, Delco's Fig. 1 illustrates the host system platform 12 which supports virtual machines 26 and 28. The hardware platform 44 similarly supports a host operating system 42 and virtual machines $54_{1\text{-}N}$. Accordingly, Delco's host system platform 12, which may be implemented as hardware platform 44 to support host operating system 42 and virtual machines $54_{1\text{-}N}$ as shown in Fig. 2, teaches "*at least one computer system*" as claimed.

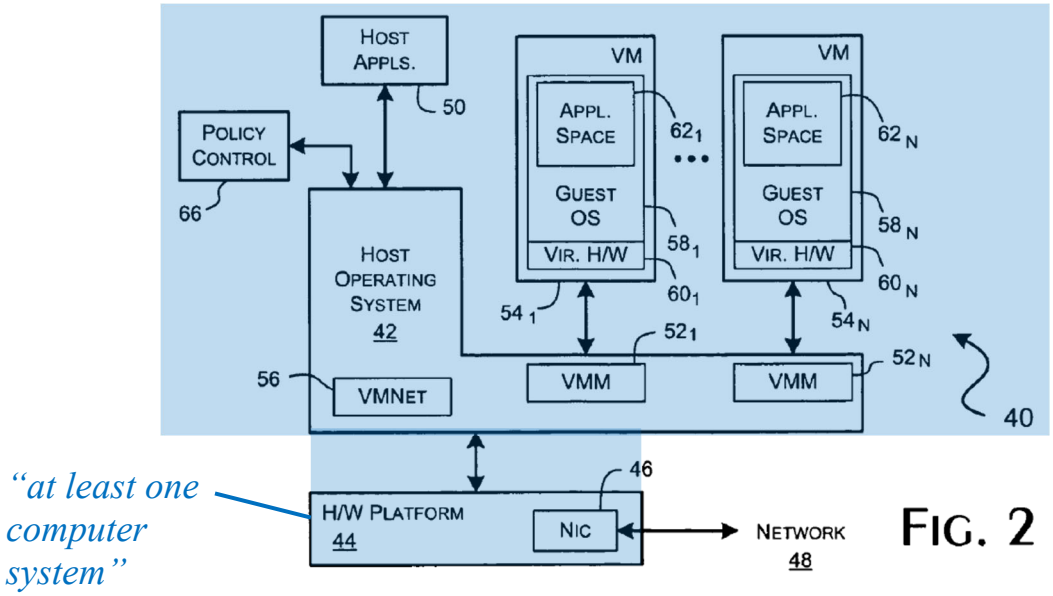


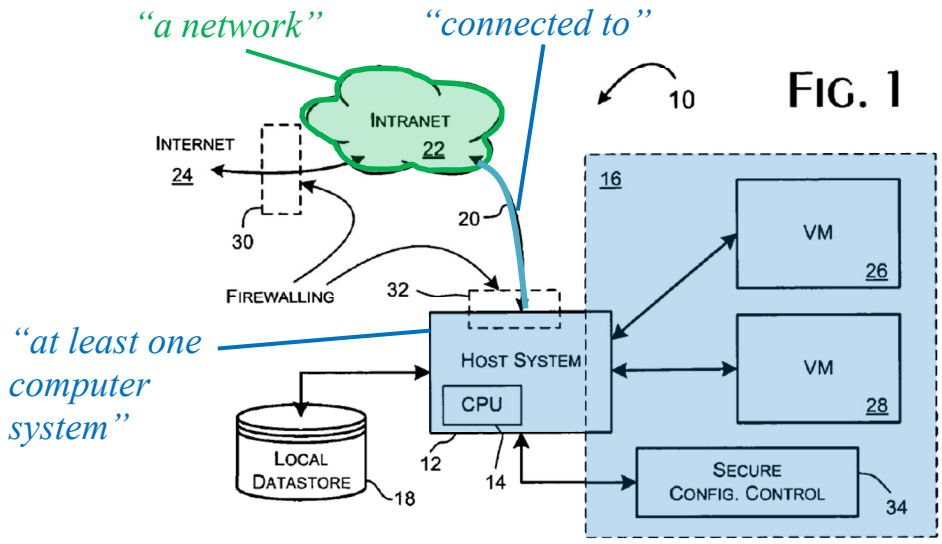
FIG. 2

Delco, Fig. 2 (annotated).

73. **Second**, Delco explains that the host system platform 12 is connected to the intranet 22 and is thus "*connected to the network*." As explained above at [1.1], Delco's computer system 10 includes an intranet 22 ("*network*") that connects the host system 12 to the Internet 24.

> **A computer system 10**, demonstrating a preferred environment for implementation of the present invention, is shown in FIG. 1. A host system platform 12 includes a local processor 14 and memory 16 providing a program execution space. **The host system platform 12 preferably supports** a local datastore 18 and **a network interface 20 permitting connection to a local intranet 22, as shown, and further, or directly in alternate scenarios, to a public communications network 24, such as the Internet**.

Delco, 4:53-61.



Delco, Fig. 1 (annotated).

74. Delco also explains that "configuration files" may be received by the host system 12 and stored "through a secure network connection with a remote configuration server accessible through the network 20." Delco, 6:4-7. Because network 20 connects to Intranet 22, which a POSITA would have understood is in turn connected to a remote configuration server, Delco renders obvious that the host system is "*configured to connect to the network.*"

75. Delco further teaches that in the example embodiment of Fig. 2, the hardware platform 44—which is an implementation of host system platform 12—is connected to a network. Fig. 2 illustrates hardware platform 44 connected to "external communications network 48." Delco, 6:19–20.

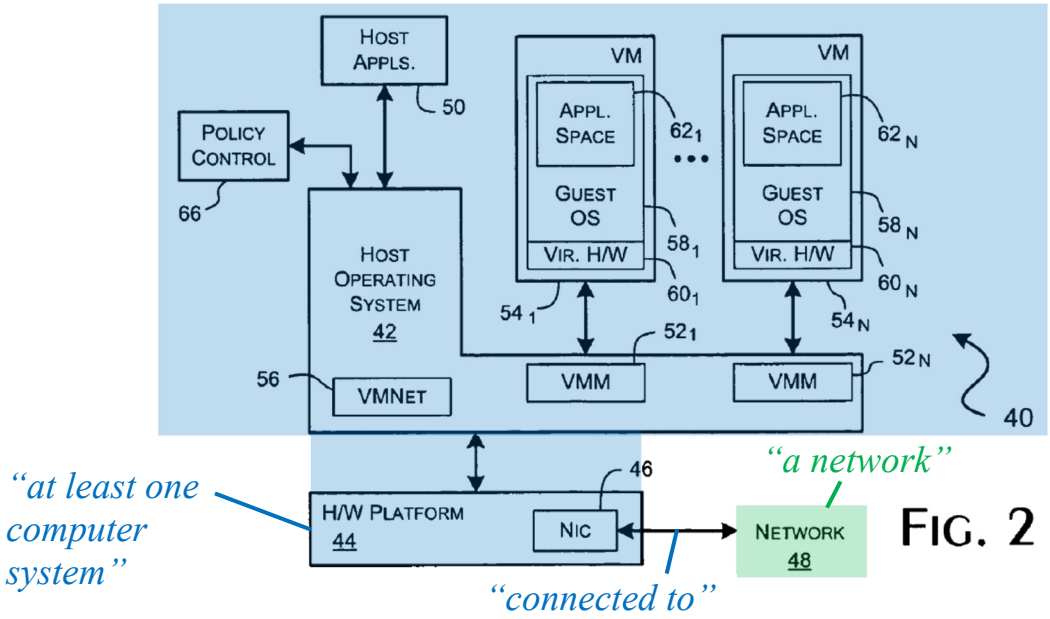


FIG. 2

Delco, Fig. 1 (annotated).

76. It would have been obvious that the network 48 may be or include features or characteristics of a local intranet, such as intranet 22. For example, Delco describes defining a "network zone" based on "characteristics of the network 48." Delco, 9:7-17; *see also* 10:51 (describing "probing the network 48 for remote changes"). Accordingly, the intranet 22 shown in Delco, Fig. 1 and/or the network 48 shown in Delco, Fig. 2 correspond to a network, as claimed.

77. Accordingly, Delco's Fig. 1 and accompanying text describe at least one computer system (the host system platform 12 in conjunction with virtual machines 26 and 28) configured to connect to the network (connected to intranet 22). Delco's Fig. 2 describes an implementation of the host system platform 12 as hardware platform 44 that supports the host operating system 42 in conjunction

with virtual machines 54 and is configured to connect to the network 48.

78. Thus, because Delco describes a network-connected host system platform 12 that is implemented as hardware platform 44 supporting a host operating system and virtual machines, Delco renders obvious "*at least one computer system configured to connect to the network*" as claimed.

[1.3] ***the computer system comprising a host system and a virtual system,***

79. **First**, Delco's host system platform 12, which may be implemented as hardware platform 44, supports a host operating system 42 ("*host system*").

> A preferred embodiment of the present invention implemented in the context of a hosted virtualization framework 40 is shown in FIG. 2. **A host operating system 42 is executed on a hardware platform 44** including a network interface controller 46. The hardware platform 44 also preferably includes a central processor 14 and a local datastore 18. The network interface controller 46 is coupleable to an external communications network 48. The host operating system 42 supports the execution of any number of conventional applications 50 and the components of the virtualization framework, which include instances of a virtual machine monitor (VMM) $52_{1\text{-}N}$ provided to support a set of one or more virtual machines (VM) $54_{1\text{-}N}$ and a virtual network controller (VMNet) 56 that provides coordinated routing of network traffic between the virtual machines $54_{1\text{-}N}$ and the various conventional interface points of a conventional network protocol stack, including a hardware specific network interface controller 46 driver, as incorporated within the host operating system 42.

Delco, 6:13-30.

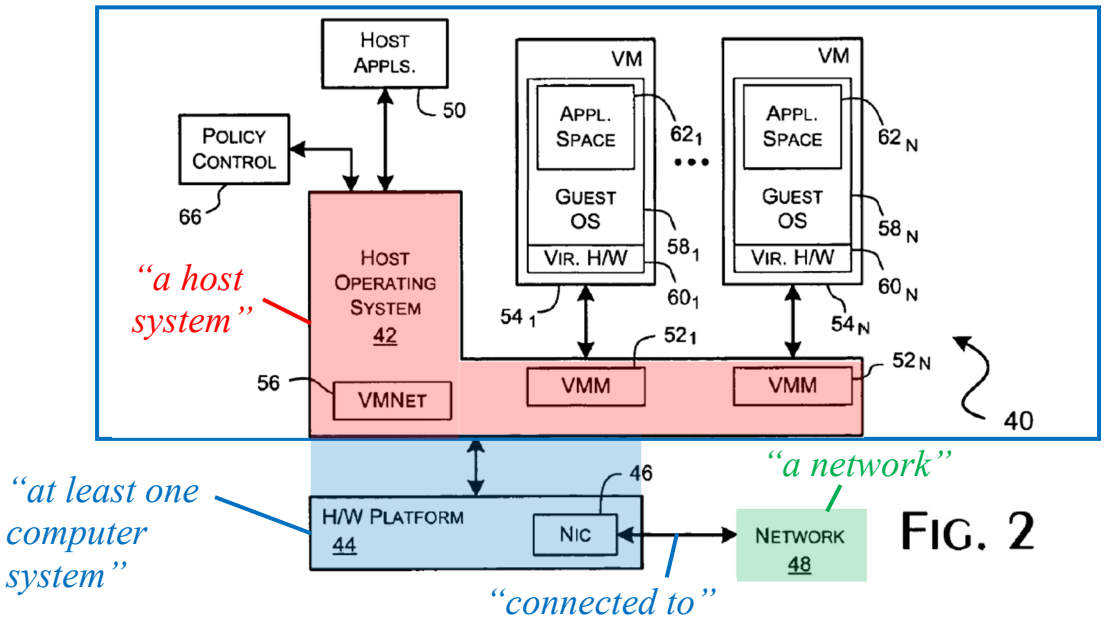


FIG. 2

Delco, Fig. 1 (annotated).

80. **Second**, Delco's host operating system 42 supports virtual machines $54_{1\text{-}N}$, that execute guest operating systems, any of which is "*a virtual system*" as claimed.

> A preferred embodiment of the present invention implemented in the context of a hosted virtualization framework 40 is shown in FIG. 2. A host operating system 42 is executed on a hardware platform 44 including a network interface controller 46. The hardware platform 44 also preferably includes a central processor 14 and a local datastore 18. The network interface controller 46 is coupleable to an external communications network 48. **The host operating system 42 supports the execution of any number of conventional applications 50 and the components of the virtualization framework, which include**

instances of a virtual machine monitor (VMM) $52_{1\text{-}N}$ provided to support a set of one or more virtual machines (VM) $54_{1\text{-}N}$ and a virtual network controller (VMNet) 56 that provides coordinated routing of network traffic between the virtual machines $54_{1\text{-}N}$ and the various conventional interface points of a conventional network protocol stack, including a hardware specific network interface controller 46 driver, as incorporated within the host operating system 42.

Delco, 6:13-30.

In the preferred embodiments, the virtual machines $54_{1\text{-}N}$ are preferably executed as individual instances of a virtual machine application loaded and managed through separate virtual machine execution processes (designated as VMX processes) instantiated within the program execution space of the host operating system 42. **The virtual machines $54_{1\text{-}N}$ functionally encapsulate guest operating systems $58_{1\text{-}N}$**. Each of the virtual machines $54_{1\text{-}N}$, when executing, are effectively isolated from one another and from the host operating system 42 as a function of the applied virtual machine world context switch. The hardware platform 44, as programmed in conjunction with the world context switch, establishes conventional memory space controls that restrict the accessible memory space to that defined for the executing virtual machine $54_{1\text{-}N}$.

In turn, each virtual machine $54_{1\text{-}N}$, as executed, then preferably further defines a guest operating system execution space above a virtual hardware component layer $60_{1\text{-}N}$. The guest operating system execution space represents the apparent physical memory address space

available to the guest operating system $58_{1\text{-}N}$. The individual guest operating system execution spaces are therefore logically coextensive with the instances of the guest operating systems $58_{1\text{-}N}$, including program execution spaces $62_{1\text{-}N}$ held available for the execution of applications by the guest operating systems $58_{1\text{-}N}$. **As is typical of conventional operating systems, the guest operating systems $58_{1\text{-}N}$ define and constrain the execution of programs within the program execution spaces $62_{1\text{-}N}$ to preclude, at a minimum, memory accesses beyond the boundaries of the program execution spaces $62_{1\text{-}N}$.**

Delco, 6:31-60.

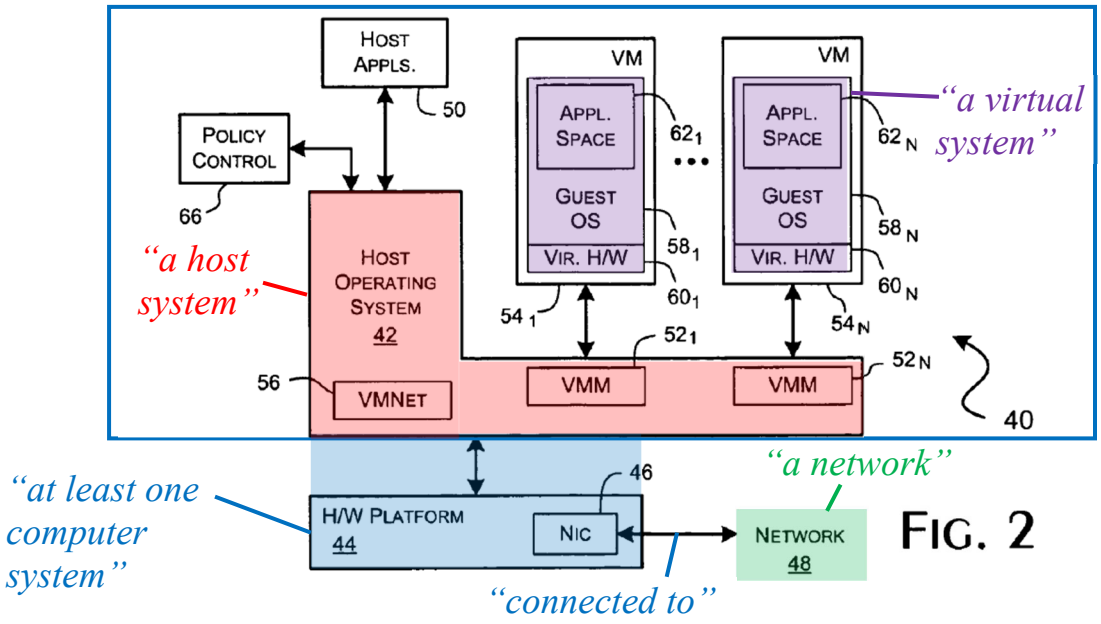


FIG. 2

Delco, Fig. 1 (annotated).

81. Accordingly, Delco's virtual machines support guest operating systems $58_{1\text{-}N}$. Any of the guest operating systems $58_{1\text{-}N}$, which execute on the virtual machines $54_{1\text{-}N}$, teach a "*virtual system*" as claimed.

82. Thus, Delco teaches a host system platform 12 implemented as a hardware platform 44 that executes a host operating system 42 supporting a number of virtual machines with guest operating systems, which renders obvious "*the computer system comprising a host system and a virtual system*" as claimed.

[1.4] ***wherein the virtual system is a separate operating system or a software module operating on the computer system[1],***

83. **First**, Delco's virtual machines $54_{1\text{-}N}$ support guest operating systems ("*virtual system[s]*") which are "*separate operating system[s]*" consistent with a POSITA's understanding of virtual machines. Delco describes virtual machines as "**separate** virtual machine execution processes" which are "effectively **isolated** from one another and from the host operating system." Delco, 6:31-60.

> In the preferred embodiments, **the virtual machines $54_{1\text{-}N}$ are preferably executed as individual instances of a virtual machine application loaded and managed through separate virtual machine execution processes** (designated as VMX processes) instantiated within the program execution space of the host operating system 42. **The virtual machines $54_{1\text{-}N}$ functionally encapsulate guest operating systems $58_{1\text{-}N}$. Each of the virtual machines $54_{1\text{-}N}$, when executing, are effectively isolated from one another and from the host operating system 42 as a function of the applied virtual**

[1] A POSITA would have understood that "the computer system" refers to "the at least one computer system."

> **machine world context switch**. The hardware platform 44, as programmed in conjunction with the world context switch, establishes conventional memory space controls that restrict the accessible memory space to that defined for the executing virtual machine $54_{1\text{-}N}$. In turn, **each virtual machine $54_{1\text{-}N}$, as executed, then preferably further defines a guest operating system execution space above a virtual hardware component layer $60_{1\text{-}N}$**. The guest operating system execution space represents the apparent physical memory address space available to the guest operating system $58_{1\text{-}N}$. The individual guest operating system execution spaces are therefore logically coextensive with the instances of the guest operating systems $58_{1\text{-}N}$, including program execution spaces $62_{1\text{-}N}$ held available for the execution of applications by the guest operating systems $58_{1\text{-}N}$. As is typical of conventional operating systems, the guest operating systems $58_{1\text{-}N}$ define and constrain the execution of programs within the program execution spaces $62_{1\text{-}N}$ to preclude, at a minimum, memory accesses beyond the boundaries of the program execution spaces $62_{1\text{-}N}$.

Delco, 6:31-60.

84. Delco also explains that guest operating systems are "*separate operating system[s]*" because the virtual machines "logically encapsulate[] a separate instance of an operating system." Delco, 1:21-23. Delco further explains that this "virtualization architecture…enforce[s] **isolation** between the mulitple virtual machines." Delco, 1:37-39. "This isolation enables the virtual machines to encapsulate and **execute a different operating system**[], whether based on type,

such as Windows7 and Linux7, specific OS variant, such as WindowsXPJ or RedHat7 Linux, or OS version." Delco, 1:43-47.

85. Like virtual machines 26 and 28 described with respect to Fig. 1, "[t]he virtual machines $54_{1\text{-}N}$ functionally encapsulate guest operating systems $58_{1\text{-}N}$." Delco, 6:36-37. These operating systems are *separate* because "[e]ach of the virtual machines 54, when executing, are effectively isolated from one another and from the host operating system 42." Delco, 6:37-41. This isolation is accomplished by "restrict[ing] the accessible memory space to that defined for the executing virtual machine $54_{1\text{-}N}$." Delco, 6:43-45.

86. Delco also teaches the alternatively claimed limitation in which the virtual machines are "*software modules*." Delco explains that "[t]he virtual hardware component layer $60_{1\text{-}N}$ of the virtual machines $54_{1\text{-}N}$ is preferably implemented **as a coordinated set of software components**." Delco, 6:61-63. Thus, while Delco's description of virtual machines 26, 28, and/or 54 as operating systems is adequate to render the claim language obvious, Delco also renders the alternatively claimed limitation obvious.

87. **Second**, Delco's guest operating systems $58_{1\text{-}N}$ "*operate on*" the hardware platform ("*the computing system*"). Delco explains that its systems "allow direct execution of the in-context virtual machine 26, 28 **on the host system 12**." Delco, 5:51-52. In addition, "the host system platform 12…supports

the execution of guest operating systems, here represented as virtual machines 26, 28." Delco, 5:18-20.

88. Like the virtual machines 26, 28 described with respect to Fig. 1, the virtual machines $54_{1\text{-}N}$ and their guest operating systems $58_{1\text{-}N}$ are executed on the hardware platform ("*the computing system*"). "In the preferred embodiments, the virtual machines $54_{1\text{-}N}$ are preferably executed as individual instances of a virtual machine application loaded and managed through separate virtual machine execution processes (designated as VMX processes) instantiated within the program execution space of the host operating system 42." Delco, 6:31-36. In addition, "the virtual machines $54_{1\text{-}N}$" operate "with support from the virtual machine monitor $52_{1\text{-}N}$, host operating system 42 resident drivers, and the underlying hardware platform 44." Delco, 6:65-67.

89. Thus, because Delco's guest operating systems running on the virtual machines are separate operating systems running on the host system platform, Delco renders obvious "*wherein the virtual system is a separate operating system or a software module operating on the computer system*" as claimed.

[1.5] ***wherein an internal firewall is configured to separate the host system from the virtual system in the computer system, and***

90. **First**, each of Delco's virtual machines includes a packet filter 74 ("*internal firewall*") that filters packets passing between the virtual system and the host system. By filtering packets between the virtual machine and the host system,

the packet filter is "*configured to separate the host system from the virtual system in the computer system*."

91. Delco describes a "virtual network interface component" as part of the virtual machine hardware.

> **A virtual network interface controller (VNic) is preferably implemented and executed as a component within the virtual hardware component layer 60_{1-N}**. As shown in FIG. 3, in accordance with a preferred embodiment of the present invention, **a virtual network interface controller component 70 includes**, representationally, a virtual network interface layer 72 combined with a packet filter layer 74. The network interface layer 72 preferably implements an interface emulation of a conventional network interface controller, enabling a conventional vendor supplied network interface controller driver to be loaded and used by the guest operating system 58_{1-N}. In effect, data link level network packet traffic is transferred by the network interface layer 72 between a guest operating system 58_{1-N} and the virtual network controller 56 through the packet filter layer 74.

Delco, 7:4-18.

92. The virtual network interface component 70 is shown in Fig. 3, below.

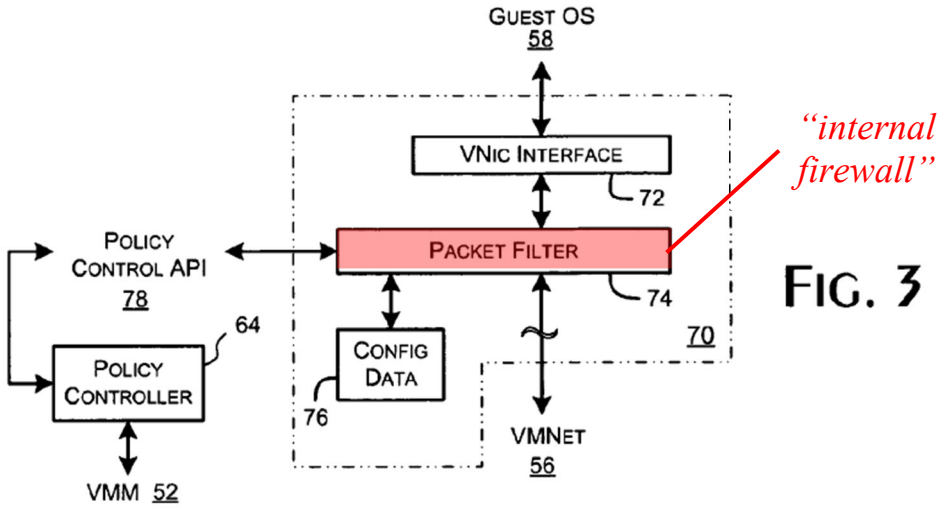


FIG. 3

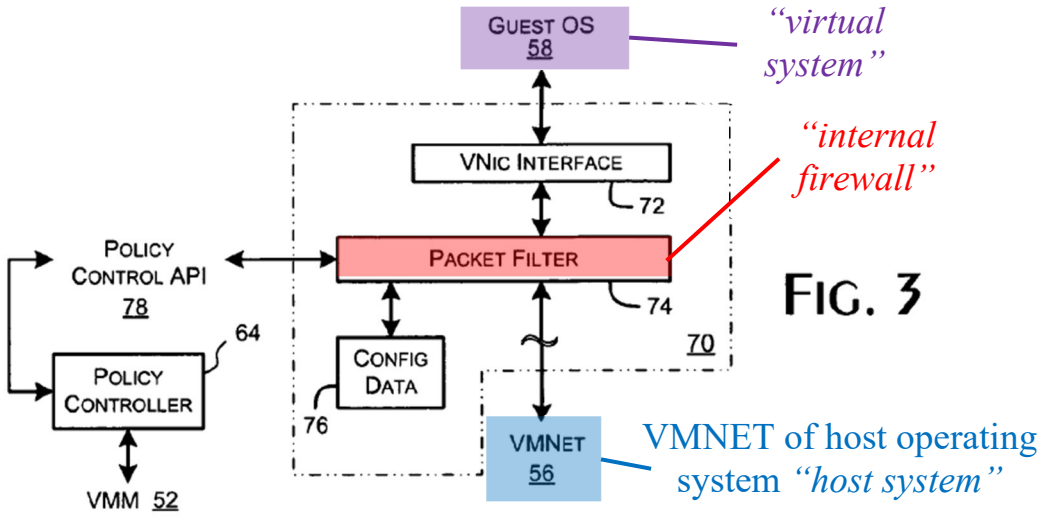
Delco, Fig. 3 (annotated).

93. Delco's packet filter layer 74 is an "*internal firewall*" because it performs the function of a firewall—filtering packets[2]—between the internal virtual machine and destinations that are external to the virtual machine (e.g., the host system 42 or over the Internet). Delco explains that the packet filter layer 74 may be configured according to "a set of policy rules." Delco, 7:29. Delco describes a wide range of different customized policy rules regarding "per-packet"

[2] *See e.g.*, Delco, 1:65-2:5 ("Firewall system architectures are conventionally well-known as implementing various stateful and stateless **network packet processing functions** to selectively control the network traffic passed through the firewall system. The packet processing functions typically include **discrete packet filtering**").

filtering (Delco, 8:36), which ports the virtual machine can access (see Delco, 8:43-45), etc.

94. **Second**, Delco's packet filter 74 is placed between the host operating system 42 and the guest operating systems 58 and is thus "*configured to separate the host system from the virtual system in the computer system.*" Specifically, the virtual network interface component 70 is placed between the Guest OS 58 ("*virtual system*") and the VMNet 56 of the host operating system ("*host system*") as shown below.



Delco, Fig. 3 (annotated).

95. Delco explains that "An advantage of the present invention is that the distributed implementation of packet filters in the individual virtual machines enables each virtual machine to discretely manage network communications with respect to **all external entities, including the host operating system** and other

virtual machines." Delco 3:40-46. By filtering traffic between the virtual system and the host system, the packet filter 74 acts as an internal firewall that implements isolation between the virtual machine(s) and the host operating system.

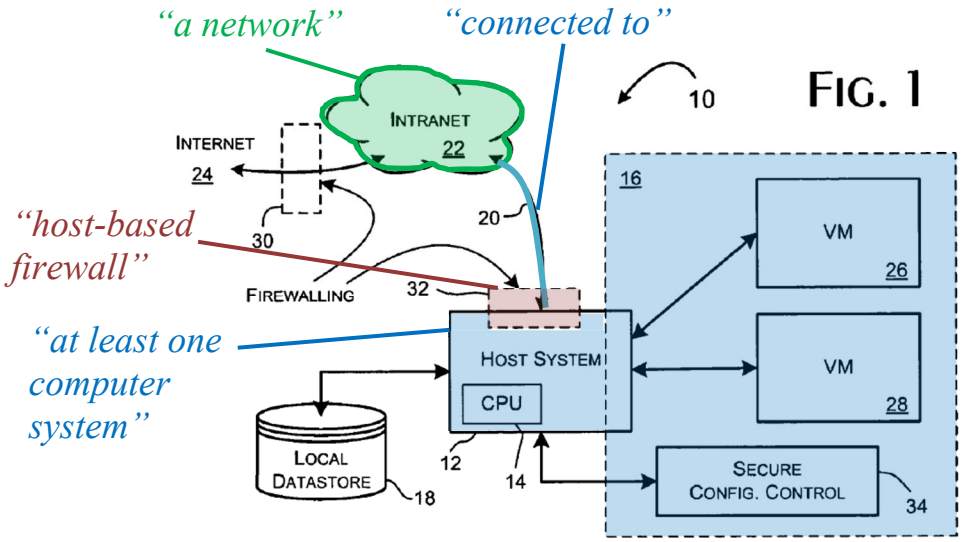
96. Thus, because Delco's packet filter acts as a firewall that separates the virtual machine from the host system and the network, Delco renders obvious "*wherein an internal firewall is configured to separate the host system from the virtual system in the computer system*" as claimed.

[1.6] ***a host-based firewall executed on the computer system is configured to implement network isolation between the computer system and the network; and***

97. **First**, Delco describes a packet filter application 32 ("*host-based firewall*") that is executed on the host system platform 12 ("*executed on the computer system*").

> Alternately, though more typically in conjunction, **a packet filter application 32 is conventionally implemented and executed within the program execution space of the host system platform 12**. Various network security and content filter applications for client system platform configurations are available from Symantec Corporation, Cupertino, Calif. Typically, these filter applications are loaded from the local datastore 18 and started subsequent to the initial execution of the host system platform 12 operating system.

Delco, 5:8-17.



Delco, Fig. 1 (annotated).

98. The packet filter 32 is a "*firewall*" because it performs the function of a firewall—filtering packets. *See e.g.*, Delco, 1:65-2:5. Indeed, Delco's Fig. 1 labels the packet filter application 32 as "FIREWALLING." Delco, 4:65, 5:8-9. Because the packet filter application 32 is positioned between the host system platform 12 and the intranet 22, it performs network isolation between the host system platform 12 ("*the computer system*") and the intranet 22 ("*the network*").

99. As shown in Fig. 1, the packet filter application 32 is positioned between the host system 12 and the intranet 22. Therefore, a POSITA would have understood that the packet filter application 32 isolates the host system 12 from the intranet 22. Delco explains that the firewall 32 is designed to protect the host system 12, and the virtual machines 26, 28, from vulnerabilities. *See* Delco, 5:22-24. A POSITA would have recognized that the filter 32 would selectively isolate

the host system 12 from the intranet 22 to prevent malicious files from being transferred to the host system 12 and vice versa.

100. A POSITA's understanding that Delco's packet filter application 32 renders obvious a firewall because it would perform the same functions as the host-based firewall claimed is further confirmed by the '780 patent itself. The '780 patent explains that its architecture "is preferably implemented by commercial-off-the-shelf (COTS) hardware, all of which is readily available. The computers described herein and the networks, both local area and the internet, rely on the standard well known network hardware now in common use." '780 patent, 4:50-55. The '780 patent then specifically admits that there is nothing novel about the host-based firewall specifically, explaining that it "may be implemented using software such as, for example, that sold under the names Symantec Endpoint Protection or MacAfee Host-Based Security Systems." '780 patent, 7:45-48. Delco similarly explains that "Various network security and content filter applications for client system platform configurations are available from Symantec Corporation, Cupertino, Calif." Delco, 5:12-14.

101. Thus, because Delco discloses the conventional host-based firewall (packet filter application 32) executed on the computer system (host system 12), which is configured to implement network isolation between the computer system and the network (to prevent malicious files from begin transferred to the host

system 12), Delco renders obvious "*a host-based firewall executed on the computer system is configured to implement network isolation between the computer system and the network*" as claimed.

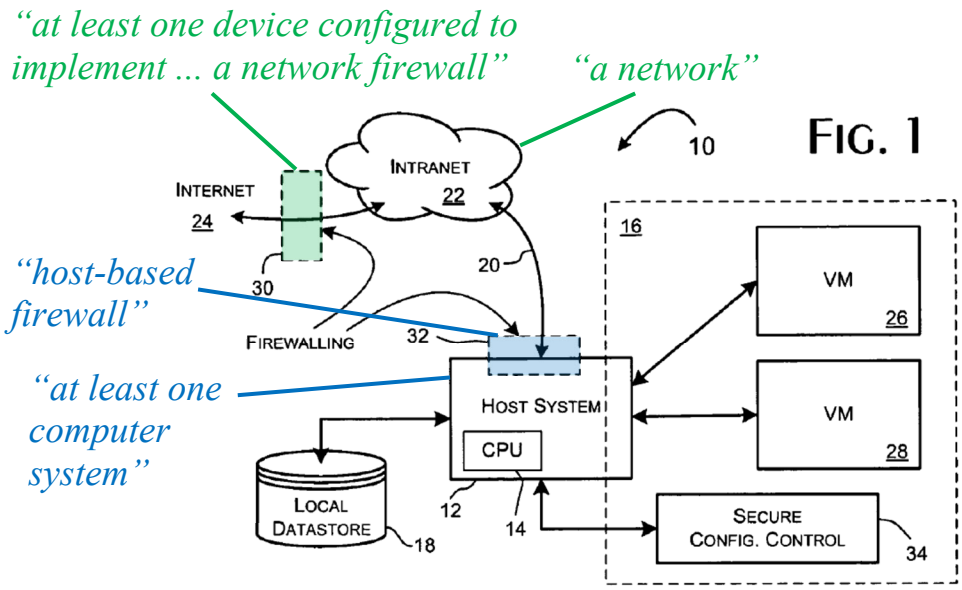
[1.7] ***at least one device configured to implement at least one of a network firewall or a web proxy,***

102. Delco further describes an independent firewall system 30 ("*network firewall*") that is external to the host system platform.

> In conventional use configurations, **an independent firewall system 30 can be installed generally as shown to provide network protection to the intranet 22 including the host system platform 12**. The firewall system 30 is conventionally a network appliance implementing any or all of a variety of packet inspection and control functions. For purposes of describing the present invention, the term packet filter is used to encompass packet filtering packages, such as exemplified by the conventional IPTables and IPChains packages, content filters, such as the well-known SpamAssassin and Clam AntiVirus (ClamAV) packages, deep packet analysis and scrubbing packages, and other packages implementing network packet security functions.

Delco, 4:62-5:7.

103. The firewall system 30 in Fig. 1 is positioned between the intranet 22 and the internet 24. Delco explains that this **firewall 30** "selectively filter[s] network traffic to and from the physical computer system." Delco, 1:63–67.



Delco, Fig. 1 (annotated).

104. Delco's firewall system 30 is analogous to the "network firewall" described and claimed in the '780 patent. The '780 patent explains that the network firewall is a conventional product. It first describes "a firewall 3 and a router 2 of conventional configuration that is well-known in the art." '780 patent, 7:9-10. Delco's firewall system is an example of the "firewall" that the '780 patent acknowledges is well-known in the art.

105. Thus, because Delco describes an external device that functions as a firewall system, Delco renders obvious "*at least one device configured to implement at least one of a network firewall*" as claimed. Alternatively, as discussed further for limitation [1.8] below, Adams describes a "gateway apparatus 105" that operates as a "Web proxy… and a firewall appliance." Ex.1006, [0021].

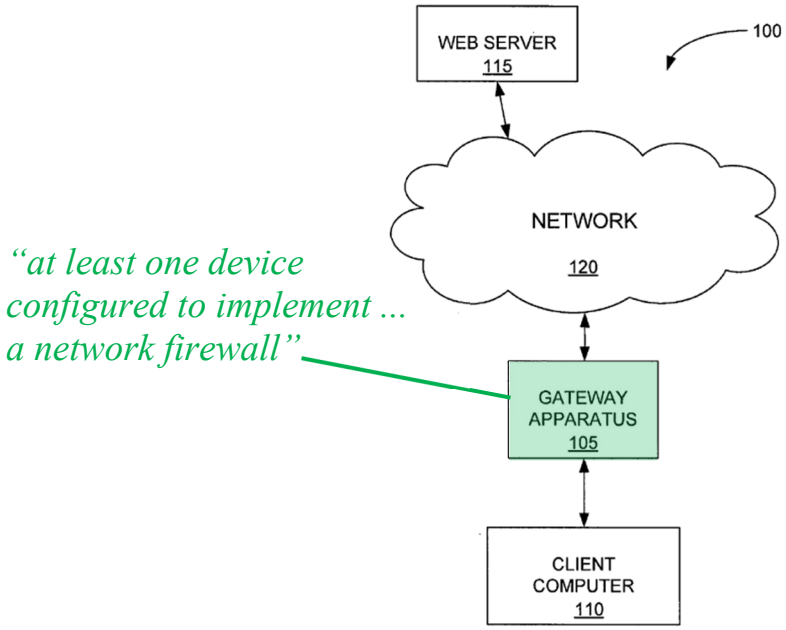
In view of Adams' disclosure, it would have been obvious for Delco's firewall system 30 to additionally, or alternatively, implement a *web proxy*.

[1.8] ***wherein the device comprises a processor and a memory configured to implement network isolation between one or more untrusted network destinations and the networked computer system.***

106. **First**, a POSITA would have found it obvious for Delco's firewall system 30 to include a processor and a memory, as evidenced by Adams.

107. Delco does not provide details regarding the structure of the firewall system 30, i.e., whether it is a device that includes *a processor and a memory*. However, in combination with Adams, a POSITA would have understood that a firewall, such as the firewall system 30, would include *a processor and a memory configured to implement network isolation between one or more untrusted network destinations and the networked computer system*. *See supra* VIII.A.3.

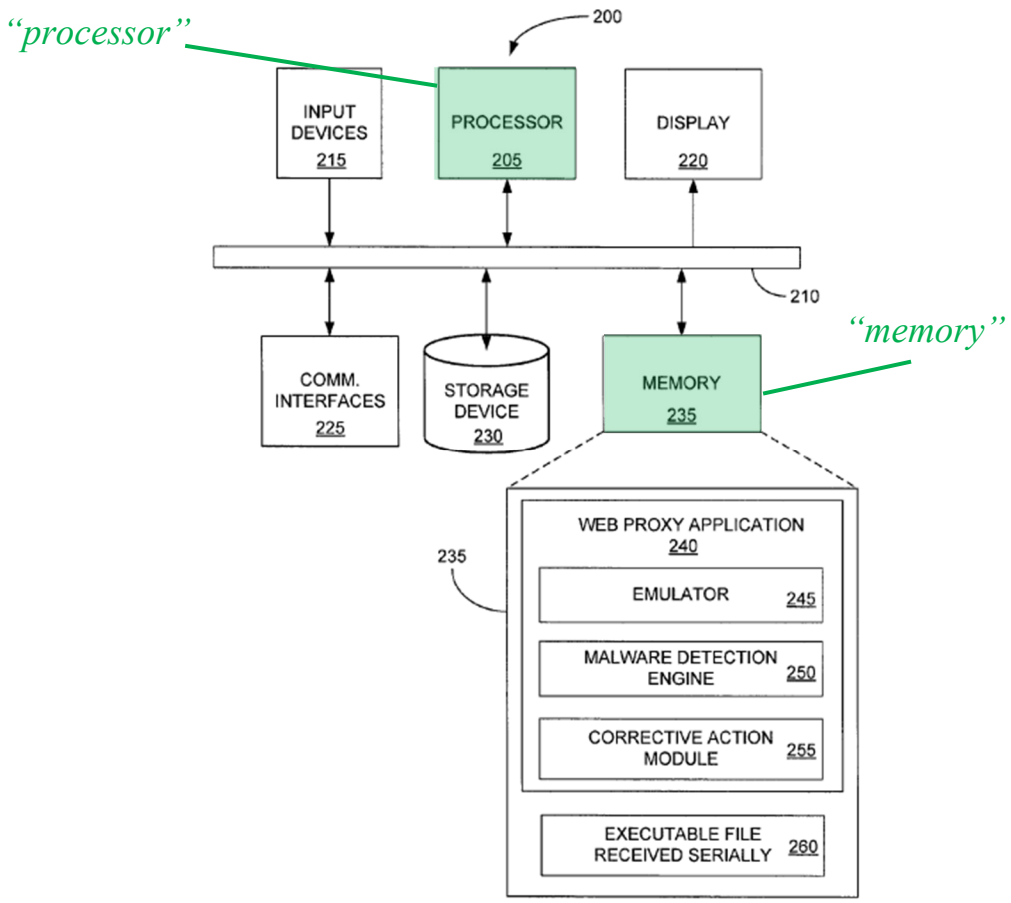
108. Adams explains that a firewall would include a processor and a memory to perform its intended functions. Adams illustrates a "gateway apparatus 105" in Fig. 1, reproduced below.



Adams, Fig. 1 (annotated).

109. Adams' "gateway apparatus 105" may be any of "a Web proxy, a router, and a firewall appliance." Adams, [0021]. Just as the firewall system 30 in Delco is positioned between the internet 24 and the intranet 22, the gateway apparatus 105 is positioned between the network 120 "which, in some embodiments, includes the Internet" and the client computer 110. Adams, [0021].

110. Adams, Fig. 2 is "a functional block diagram of one type of gateway apparatus" such as the gateway apparatus 105 of Fig. 1. The gateway apparatus can include a "processor 205" and a "memory 235" as show in Fig. 2 below.



Adams, Fig. 2 (annotated).

111. Adams' gateway apparatus also teaches a "*web proxy*" as alternatively claimed. Adams explains that its gateway apparatus includes a web proxy application stored in memory.

> Memory 235 may include, without limitation, random access memory (RAM), read-only memory (ROM), flash memory, magnetic storage (e.g., a hard disk drive), optical storage, or a combination of these, depending on the particular embodiment. In FIG. 2, memory 235 includes Web proxy application 240. As those skilled in the computer networking art are aware, a Web proxy server is a gateway apparatus

> that services the requests of client computers by forwarding those requests to other servers on the network.

Adams, [0025].

112. Adams' gateway apparatus isolates the client computer 110 from the network 120. For example, the gateway "tests [executable files] for the presence of malware." Adams, [0027]. If a file contains malware, the gateway "prevent[s] the executable file 260 from propagating further over network 120. That is, corrective action module 255 can block an executable file 260 that includes malware, thereby preventing it from reaching a client computer 110." Adams, [0033].

113. **Second**, Delco explains that its firewall system 30 implements network isolation between the Internet 24, which it describes as "a public communications network" and the intranet 22 and host system platform 12 ("*the networked computer system*"). Delco, 4:60. A POSITA would have understood that a destination on the Internet would be considered an *untrusted network destination*, as claimed. *See, e.g.*, Marascio, [0006] ("A firewall is a system or combination of systems ... that enforce a boundary between a trusted data network and an **untrusted data network ... such as the internet**."); McArdle, 1:30-32 ("When a computer is connected to **a public network, such as the Internet**, the computer is vulnerable to attack and compromise from other computers on the network."). Specifically, Delco's firewall system 30 "selectively filter[s] network traffic to and

from the physical computer system." Delco, 1:63-67.

114. In addition, "[t]he firewall system 30 is conventionally a network appliance implementing any or all of a variety of packet inspection and control functions." Delco, 4:62–67. Filtering network traffic and inspecting and controlling functions as described in Delco is *implementing network isolation* as claimed. Indeed, the function and purpose of a gateway or firewall is to provide isolation between trusted and untrusted networks. *See e.g.*, Strathmeyer, [0040] ("[G]ateway 120 may provide isolation between the untrusted and trusted networks or network domains ... gateway 120 may include a firewall to allow only certain types of packets or traffic to pass.").

115. Thus, the combination of Delco and Adams renders obvious *a processor and a memory* (processor 205 and memory 235 of gateway apparatus 105 in Adams) *configured to implement network isolation between one or more untrusted network destinations and the networked computer system* (firewall 30 placed between internet 24 and intranet 22 in Delco and gateway apparatus 105 placed between network 120 and client computer 110 in Adams).

116. Thus, because Adams describes the structural details (e.g., processor and memory) of a firewall such as the one described by Delco, which filters packets from the Internet (an untrusted network destination), Delco and Adams renders obvious "*wherein the device comprises a processor and a memory*

configured to implement network isolation between one or more untrusted network destinations and the networked computer system" as claimed.

5. Claim 2

[2.1] ***The networked computer system of claim 1, wherein the internal firewall is configured to prevent data from being communicated between the virtual system and the host system without an explicit user input.***

117. **First**, for the reasons explained above at [1.5], Delco's packet filter layer 74 ("*internal firewall*") prevents data from being communicated from the guest operating system 58 to the host operating system 42.

118. **Second**, Delco describes a "default packet filter configuration" that is configured "*without an explicit user input*."

> Advantageously, the packet filter of a virtual network interface controller 70 **is fully initialized prior to any execution of a guest operating system $58_{1\text{-}N}$ within the corresponding virtual machine $52_{1\text{-}N}$**. Even if a zone and controller identity criteria match is not found, an administratively **defined default packet filter configuration** is established before any possible compromize of the guest operating system $58_{1\text{-}N}$.

Delco, 10:9-16.

Because the packet filter has a default configuration upon initialization, it performs its function of preventing communication "*without explicit user input*."

119. Additionally, it would have been obvious for Delco's set of policy rules to be embodied as a "white list" such that only specifically allowed types of

communication are allowed to pass.. Delco explains that users may change the filtering policy through a management interface.

> In the preferred embodiments of the present invention, the packet filter layer 74 is implemented as a library component integral to the network interface layer 72. While direct coupling is preferred to reduce complexity and processing overhead, indirect coupling may be preferred to allow flexible, potentially dynamic in-configuration of different data packet filter components to provide alternate or additional feature sets beyond baseline packet examination, such as various combinations of externally defined network connection block lists and screening for executables, viral signatures, and content. **In all cases, a set of policy rules that define the operational behavior of the packet filter layer 74 is preferably stored as configuration data in an in-memory table 76. A management program interface 78 is preferably provided to allow the setting and updating of the policy rule set held in the table 76**. Preferably, the policy rule sets are provided preparsed from a text-based policy rule set representation to a compact configuration data form suitable for direct in-memory storage and use by the packet filter layer 74.

Delco, 7:19-37.

It was known for policy rules—such as the rule set held in table 76—to exist as a white list. *See e.g.,* Thomas, [0048] ("[A] policy database may be a block list, a black list, an allowed list, a white list, or the like"). Therefore, Delco's rule set—embodied as a white list—would have functioned to prevent communication absent

"*explicit user input*" as claimed.

120. Thus, because Delco's packet filter 74 filters packets and thus prevents certain types of data from passing between the guest operating system and the host operating system, Delco renders obvious "*wherein the internal firewall is configured to prevent data from being communicated between the virtual system and the host system without an explicit user input*" as claimed.

6. Claim 3

[3.1] ***The networked computer system of claim 1, wherein the host system is configured to store data in a host memory space and the virtual system is configured to store data in a virtual memory space that is segregated from the host memory space.***

121. **First**, Delco's host operating system utilizes a program execution space ("*host memory space*").

> A computer system 10, demonstrating a preferred environment for implementation of the present invention, is shown in FIG. 1. A host system platform 12 includes a local processor 14 and **memory 16 providing a program execution space**. The host system platform 12 preferably supports a local datastore 18 and a network interface 20 permitting connection to a local intranet 22, as shown, and further, or directly in alternate scenarios, to a public communications network 24, such as the Internet.

Delco, 4:53-61.

122. Delco's system stores data within the program execution space. For

example, "a packet filter application 32 is conventionally implemented and executed within the program execution space of the host system platform 12." Delco, 5:9-12. A POSITA would have understood that the packet filter application 32 would process packets for the purpose of filtering packets. The packets themselves are data, which are stored in host memory during processing. Accordingly, Delco's host system "*is configured to store data in a host memory space*."

123. Second, Delco's guest operating systems (each one a "*virtual system*") store data in their own memory space referred to as the guest operating execution space ("*virtual memory space*").

> In the preferred embodiments, the virtual machines $54_{1\text{-}N}$ are preferably executed as individual instances of a virtual machine application loaded and managed through separate virtual machine execution processes (designated as VMX processes) instantiated within the program execution space of the host operating system 42. The virtual machines $54_{1\text{-}N}$ functionally encapsulate guest operating systems $58_{1\text{-}N}$. Each of the virtual machines $54_{1\text{-}N}$, when executing, are effectively isolated from one another and from the host operating system 42 as a function of the applied virtual machine world context switch. **The hardware platform 44, as programmed in conjunction with the world context switch, establishes conventional memory space controls that restrict the accessible memory space to that defined for the executing virtual machine $54_{1\text{-}N}$**.

Delco, 6:31-45.

> In turn, each virtual machine 54 $_{1\text{-}N}$, as executed, then preferably further defines **a guest operating system execution space** above a virtual hardware component layer 60 $_{1\text{-}N}$. **The guest operating system execution space represents the apparent physical memory address space available to the guest operating system 58$_{1\text{-}N}$**. The individual guest operating system execution spaces are therefore logically coextensive with the instances of the guest operating systems 58$_{1\text{-}N}$, including **program execution spaces 62$_{1\text{-}N}$ held available for the execution of applications by the guest operating systems 58$_{1\text{-}N}$**. As is typical of conventional operating systems, the guest operating systems 58$_{1\text{-}N}$ define and constrain the execution of programs within the program execution spaces 62$_{1\text{-}N}$ to preclude, at a minimum, memory accesses beyond the boundaries of the program execution spaces 62$_{1\text{-}N}$.

Delco, 6:46-60.

124. Delco's guest operating systems thus store application data in their respective execution spaces. For example, "program execution spaces 62$_{1\text{-}N}$ held available for the execution of applications by the guest operating systems 58$_{1\text{-}N}$." Delco, 6:46-60. Accordingly, Delco's guest operating systems are "*configured to store data in a virtual memory space*."

125. **Third**, Delco's guest operating system execution spaces are "*segregated from*" Delco's host program execution space. Delco explains that "[a]s is typical of conventional operating systems, the guest operating systems 58$_{1\text{-}N}$

define and constrain the execution of programs within the program execution spaces $62_{1\text{-}N}$ to preclude, at a minimum, memory accesses beyond the boundaries of the program execution spaces $62_{1\text{-}N}$." Delco, 6:46-60. "The hardware platform 44, as programmed in conjunction with the world context switch, establishes conventional memory space controls that restrict the accessible memory space to that defined for the executing virtual machine $54_{1\text{-}N}$." Delco, 6:31-45.

126. Thus, because Delco describes the conventional arrangement in which the guest operating systems utilize their own separate program execution space that is segregated from the host's memory space, Delco renders obvious "*wherein the host system is configured to store data in a host memory space and the virtual system is configured to store data in a virtual memory space that is segregated from the host memory space*."

7. Claim 4

[4.1] ***The networked computer system of claim 1, wherein malware introduced to the computer system is prevented from moving to another computer system of the networked computer system.***

127. As explained above at [1.5], Delco's host system platform 12 includes a host-based firewall in the form of packet filter application 32. Delco's packet filter is configured to prevent malware from moving from the host machine to an external system. For example, Delco explains that its firewalling techniques are designed to overcome flaws in previous systems in which malware could spread.

> Even where the platform firewall application functions correctly, if a security breach, whether intentional or caused by the inadvertent execution of **malware**, arises from activity within one of the virtual machines, or from within the host operating system environment, the platform firewall application is unable to prevent the breach from freely spreading between the virtual machines and the host. The platform firewall application provides even less functional protection where the virtualization framework connects below the connection point of the platform firewall application to the platform network stack.

Delco, 2:48-58.

128. Delco further explains the function of its packet filters: "For purposes of describing the present invention, the term packet filter is used to encompass packet filtering packages, such as exemplified by the conventional IPTables and IPChains packages, content filters, such as the well-known SpamAssassin and Clam AntiVirus (ClamAV) packages, deep packet analysis and scrubbing packages, and other packages implementing network packet security functions." Delco, 4:67-5:7. Thus, by filtering out spam and computer viruses, Delco's packet filters 74 prevent malware from spreading from the host system to other systems on the intranet 22 ("*network*") as claimed. And a POSITA would have found it obvious that the intranet 22 would include other systems. Indeed, the purpose of an intranet is to connect multiple systems together using a network.

129. Thus, because Delco's packet filter applications 32 is configured to

filter out spam, viruses, and other forms of malware, Delco renders obvious "*wherein malware introduced to the computer system is prevented from moving to another computer system of the networked computer system*" as claimed.

8. Claim 5

[5.0] ***The networked computer system of claim 1, wherein the computer system is configured to:***

130. As explained above with respect to claim 1, Delco's host system platform 12 includes a host operating system and a guest operating system. As will be explained below, Delco's guest operating system accesses the network through a first connection, and Delco's host operating system accesses the network through a second, separate connection.

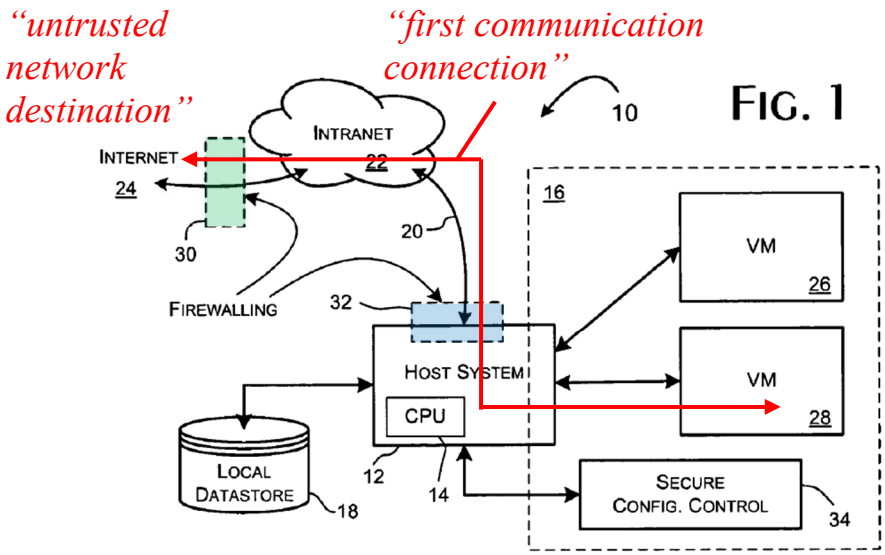
[5.1] ***communicate with the one or more untrusted network destinations via a first communication connection,***

131. Delco's virtual machines communicate with the Internet 24 ("*one or more untrusted network destinations*") through a first connection.

> A computer system 10, demonstrating a preferred environment for implementation of the present invention, is shown in FIG. 1. A host system platform 12 includes a local processor 14 and memory 16 providing a program execution space. **The host system platform 12 preferably supports** a local datastore 18 and **a network interface 20 permitting connection to** a local intranet 22, as shown, and further, or directly in alternate scenarios, to **a public communications network 24, such as the Internet**.

Delco, 4:53-61.

132. Delco notes that a "physical computer system may be used in a generally untrusted network environment." Delco, 2:10-12. A POSITA would have understood that the Internet is such an untrusted environment. *See e.g.,* Marascio, [0006] ("A firewall is a system or combination of systems ... that enforce a boundary between a trusted data network and an **untrusted data network ... such as the internet**.").



Delco, Fig. 1 (annotated).

133. The virtual machine's connection to the Internet passes through a virtual network interface.

> **A virtual network interface controller (VNic) is preferably implemented and executed as a component within the virtual hardware component layer $60_{1\text{-}N}$**. As shown in FIG. 3, in accordance

> with a preferred embodiment of the present invention, **a virtual network interface controller component 70 includes**, representationally, a virtual network interface layer 72 combined with a packet filter layer 74. The network interface layer 72 preferably implements an interface emulation of a conventional network interface controller, enabling a conventional vendor supplied network interface controller driver to be loaded and used by the guest operating system $58_{1\text{-}N}$. In effect, data link level network packet traffic is transferred by the network interface layer 72 between a guest operating system $58_{1\text{-}N}$ and the virtual network controller 56 through the packet filter layer 74.

Delco, 7:4-18.

134. The "virtual network interface controller component 70 includes, representationally, a virtual network interface layer 72 combined with a packet filter layer 74." Ex.1005, 7:8-10. Delco's packet filter is intended to filter out harmful packets from untrusted destinations such as the Internet. *See* Ex.1005, 2:26-59; Ex.1008, [0006]. Accordingly, a POSITA would have understood that Delco's virtual machine communicates with the Internet. Accordingly, the virtual machines connection to the Internet through a VNic of the virtual hardware component layer is a "*first communication connection*" as claimed.

135. Thus, because Delco's virtual machines use a virtual network interface controller to connect with the Internet, Delco renders obvious "*communicate with the one or more untrusted network destinations via a first*

communication connection" as claimed.

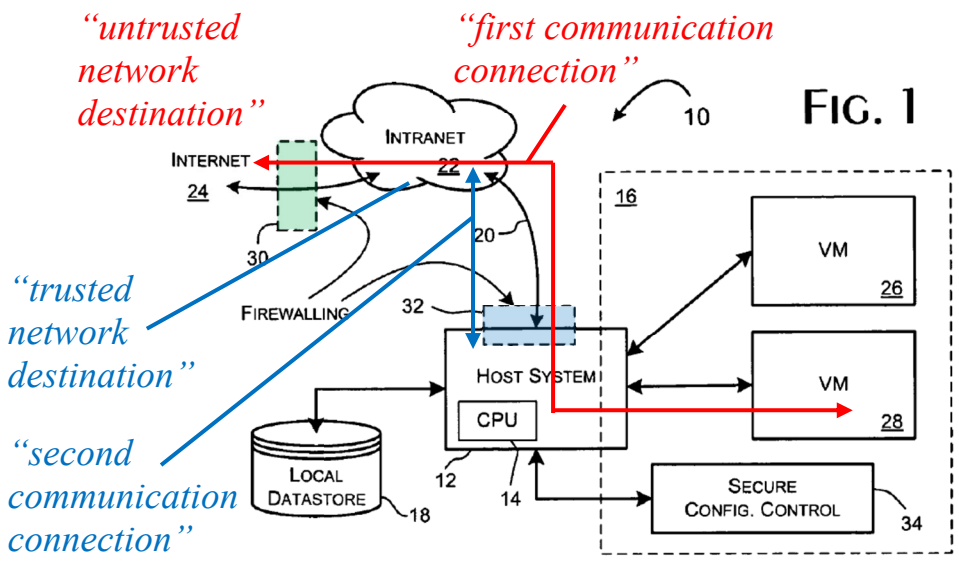
[5.2] ***communicate with one or more trusted network destinations via a second communication connection; and***

136. Delco's host computing system accesses the local intranet 22 ("*one or more trusted network destinations*") through a second connection.

> A computer system 10, demonstrating a preferred environment for implementation of the present invention, is shown in FIG. 1. A host system platform 12 includes a local processor 14 and memory 16 providing a program execution space. **The host system platform 12 preferably supports** a local datastore 18 and **a network interface 20 permitting connection to a local intranet 22**, as shown, and further, or directly in alternate scenarios, to a public communications network 24, such as the Internet.

Delco, 4:53-61.

137. Delco's intranet 22 is a "trusted" network because it is behind the firewall system 30. A POSITA would have understood that the intranet 22 is a "protected, typically corporate, network." Delco, 2:15-16. Delco explains that "In conventional use configurations, an independent firewall system 30 can be installed generally as shown to provide network protection to the intranet 22 including the host system platform 12." Delco, 4:62-66.



Delco, Fig. 1 (annotated).

138. The host system platform 12 includes "**a network interface 20 permitting connection to a local intranet 22**." Delco, 7:4-18. Accordingly, the host system's connection to the intranet through the network interface 20 is a "*second connection*" as claimed. Just like the '780 patent, Delco thus teaches a first connection through a virtual machine and a second connection through the host. *See e.g.*, Ex.1001, Fig. 2.

139. Thus, because Delco's host system connects to the intranet 22 through network interface 20, Delco renders obvious "*communicate with one or more trusted network destinations via a second communication connection*" as claimed.

[5.3] ***enforce separation of the first communication connection from the second communication connection.***

140. Delco's virtual machine connection ("*first communication*

connection") and host system connection ("*second connection*") are separated at least because they use different network controllers. Delco's host system establishes the exemplary "*second communication connection*" using the physical NIC. "A host operating system 42 is executed on a hardware platform 44 including a network interface controller 46." Delco, 6:15-17. Separately, the virtual machine's exemplary "*first communication connection*" uses a VNic: "A virtual network interface controller (VNic) is preferably implemented and executed as a component within the virtual hardware component layer $60_{1\text{-}N}$." Delco, 7:4-6. "[E]ach virtual machine 54, as executed, then preferably further defines a guest operating system execution space above a virtual hardware component layer $60_{1\text{-}N}$." Delco, 6:46-48.

141. The VNic of the virtual machine is separate from the NIC on the host system. Delco, 6:31-45. Delco explains that the virtual machines are "managed through separate virtual machine execution processes" and are "effectively isolated from one another and from the host operating system 42 as a function of the applied virtual machine world context switch." Delco, 6:31-45. Delco explains that "a host operating system 42 is executed on a hardware platform 44 including a network interface controller 46." Delco, 6:15-17. Separately, "a virtual machine monitor (VMM) $52_{1\text{-}N}$ [is] provided to support a set of one or more virtual machines (VM)$54_{1\text{-}N}$ and a virtual network controller (VMNet) 56 that provides

coordinated routing of network traffic between the virtual machines 54 and the ... hardware specific network interface controller 46 driver." Delco, 6:23-30. Accordingly, because Delco's system relies on the VNic for the virtual machine (using the "*first communication connection*") and relies on only the physical NIC 46 for the host system (using the "*second communication connection*"), Delco's system "*enforce[s] separation of the first communication connection from the second communication connection*" as claimed. The separate connections are illustrated in Fig. 2 below.

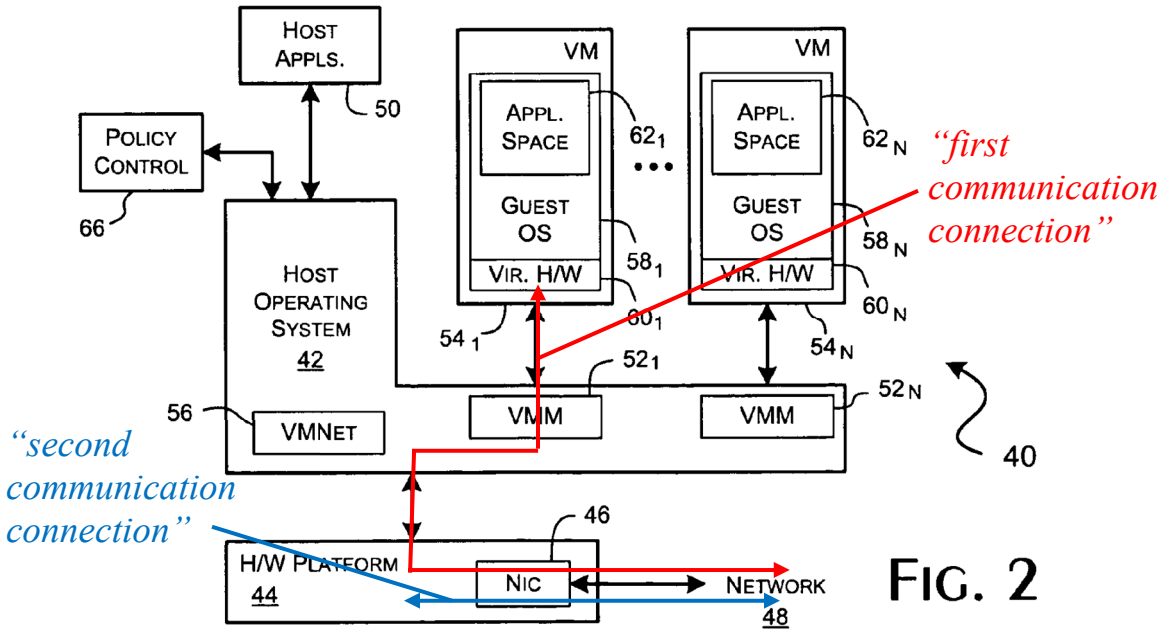


FIG. 2

Delco, Fig. 2 (annotated).

142. Additionally, the first and second communication connections are separated in that they use different filters with different policy rules. Delco's packet filter application 32 and packet filters 74 use different policy

configurations. "[A]nother advantage of the present invention is that the policy rule sets can be defined to implement security controls on **a per network interface basis**, including individually for ... individual virtual machines [and] for a defined group of virtual machines, **potentially including the host operating system**, and administratively determined combinations." Delco, 4:11-18. Thus, because Delco's virtual machine connection and host connection are processed by different packet filters, one on a separate virtual machine, Delco renders obvious "*enforce separation of the first communication connection from the second communication connection*" as claimed.

9. Claim 6

[6.1] ***The networked computer system of claim 1, wherein the host-based firewall is configured to prevent lateral communication and movement of malware between the computer system and other devices on the networked computer system.***

143. As explained above at [1.6], Delco's packet filter application 32 teaches a "*host-based firewall*" as claimed. Because the packet filter application 32 is positioned between the host system platform 12 and the intranet 22, it performs network isolation between the host system platform 12 ("*the computer system*") and the intranet ("*the network*"). By filtering packets between the host system platform 12 and other devices on the intranet 22, the packet filter application 32 prevents "*lateral communication and movement of malware*" between the host system platform 12 and "*other devices on the networked computer system*." A

POSITA would have understood the term "*lateral communication*" to include communication between devices on the intranet.

144. Furthermore, as explained above at [4.1], Delco's packet filters are configured to filter spam, viruses, and other forms of malware.

145. Thus, because Delco's host system platform 12 includes a packet filter application to prevent the transfer of malware between the host system platform and other devices on the intranet, Delco renders obvious "*wherein the host-based firewall is configured to prevent lateral communication and movement of malware between the computer system and other devices on the networked computer system*" as claimed.

10. Claim 7

[7.1] ***The networked computer system of claim 1, wherein the device is configured to prevent unauthorized communication between the computer system and the one or more untrusted network destinations.***

146. **First**, as explained above at [1.7], Delco's firewall system 30 teaches the claimed "*device*."

147. **Second**, as explained above at [1.8], Delco's firewall system 30 provides network isolation between the host system platform on the intranet 22, and the Internet 24 ("*one or more untrusted network destinations*").

148. **Third**, Delco's firewall system 30 filters unauthorized communication consistent with the known functionality of firewall systems. As Delco explains:

> Firewall system architectures are conventionally well-known as implementing various stateful and stateless network packet processing functions to selectively control the network traffic passed through the firewall system. The packet processing functions typically include discrete packet filtering, such as can be performed by the open source IPTables and IPChains software packages, and aggregated content packet filtering, as can be performed by **various spam filter applications**, all conventionally referred to generally as packet filtering.

Delco, 1:67-2:9.

149. Delco further explains:

> For purposes of describing the present invention, the term packet filter is used to encompass packet filtering packages, such as exemplified by the conventional IPTables and IPChains packages, content filters, such as the well-known **SpamAssassin and Clam AntiVirus (ClamAV)** packages, deep packet analysis and scrubbing packages, and other packages implementing network packet security functions.

Delco, 4:67-5:7.

150. Delco's firewall system, which filters for viruses, spam, or other security concerns, prevents "*unauthorized communication*" as claimed. Indeed, Delco describes the same type of firewalls and packet filters as does the '780 patent. *Compare* Delco, 5:11-13 ("Various network security and content filter applications for client system platform configurations are available from Symantec

Corporation") *with* Ex.1001, 7:46-47 ("Symantec Endpoint Protection or MacAfee Host-Based Security Systems"). Accordingly, a POSITA would have recognized that Delco's firewall system prevents the same type of unauthorized communications and performs the same types of filtering functions that are described in the '780 patent.

151. Thus, because Delco's firewall system 30 prevents viruses, spam, and other security concerns from transferring between the Internet 24 and the host system platform 12, Delco renders obvious "*wherein the device is configured to prevent unauthorized communication between the computer system and the one or more untrusted network destinations*" as claimed.

11. Claim 9

[9.1] ***The networked computer system of claim 1, wherein one or more applications or processes are configured to run in the virtual system, and***

152. Delco describes one or more applications that run on a guest operating system ("*virtual system*"). Delco also refers to "Applications executing in the other virtual machines co-resident on the physical computer system platform." Delco, 3:67-4:3. Delco further explains that the guest operating systems are configured to support applications.

> In turn, each virtual machine 54 $_{1\text{-}N}$, as executed, then preferably further defines a guest operating system execution space above a virtual hardware component layer 60 $_{1\text{-}N}$. The guest operating system execution

> space represents the apparent physical memory address space available to the guest operating system $58_{1\text{-}N}$. The individual guest operating system execution spaces are therefore logically coextensive with the instances of the guest operating systems $58_{1\text{-}N}$, including **program execution spaces $62_{1\text{-}N}$ held available for the execution of applications by the guest operating systems $58_{1\text{-}N}$**. As is typical of conventional operating systems, the guest operating systems $58_{1\text{-}N}$ define and constrain the execution of programs within the program execution spaces $62_{1\text{-}N}$ to preclude, at a minimum, memory accesses beyond the boundaries of the program execution spaces $62_{1\text{-}N}$.

Delco, 6:46-60.

153. Because the applications run on the virtual machine, they are "*configured to run in the virtual system*" as claimed. Accordingly, because Delco's virtual machines include execution space for the execution of applications by the guest operating systems, Delco renders obvious "*wherein one or more applications or processes are configured to run in the virtual system*" as claimed.

[9.2] *wherein the one or more applications or processes running in the virtual system are configured to communicate with the one or more untrusted network destinations.*

154. Traffic from the applications running on Delco's guest operating systems is allowed to communicate with an external network 48, which would include the Internet (i.e., an "*untrusted network destination*"). *See, e.g.*, Marascio, [0006] ("A firewall is a system or combination of systems ... that enforce a

boundary between a trusted data network and an **untrusted data network ... such as the internet**."). Delco notes there is a "need to provide appropriate security constraints between the **applications** running within the virtual machines, a host operating system if present, **and the connected network environment**." Delco, 1:59-62. As illustrated in Fig. 1, Delco's connected network environment includes an intranet 22 and the Internet 24. Delco's Fig. 2 illustrates and describes connections to external network 48, which a POSITA would understand to include both intranet 22 and the Internet 24. Indeed, it was known in the art for virtual machine applications to communicate with "untrusted" destinations. *See generally,* Thomas.

155. Thus, because the applications on Delco's virtual systems access the Internet, which is understood to be an untrusted network, Delco renders obvious "*wherein the one or more applications or processes running in the virtual system are configured to communicate with the one or more untrusted network destinations*" as claimed.

12. Claim 10

[10.1] ***The networked computer system of claim 1, wherein one or more applications or processes are configured to run in the host system, and***

156. Delco explains that conventional applications run on the host system.

> A preferred embodiment of the present invention implemented in the context of a hosted virtualization framework 40 is shown in FIG. 2. A

> host operating system 42 is executed on a hardware platform 44 including a network interface controller 46. The hardware platform 44 also preferably includes a central processor 14 and a local datastore 18. The network interface controller 46 is coupleable to an external communications network 48. **The host operating system 42 supports the execution of any number of conventional applications 50** and the components of the virtualization framework, which include instances of a virtual machine monitor (VMM) $52_{1\text{-}N}$ provided to support a set of one or more virtual machines (VM) $54_{1\text{-}N}$ and a virtual network controller (VMNet) 56 that provides coordinated routing of network traffic between the virtual machines $54_{1\text{-}N}$ and the various conventional interface points of a conventional network protocol stack, including a hardware specific network interface controller 46 driver, as incorporated within the host operating system 42.

Delco, 6:13-30.

157. Thus, because Delco's host system supports conventional applications, Delco renders obvious "*wherein one or more applications or processes are configured to run in the host system*" as claimed.

[10.2] ***wherein the one or more applications or processes running in the host system are configured to communicate with one or more devices on the network.***

158. Delco's applications 50, which run on the host system platform 44, communicate with an external network 48, which a POSITA would understand to include the intranet 22 ("*network*"). Delco's host system platform 12 uses a "network interface 20 permitting connection to a local intranet 22." Delco, 4:58-

59. A POSITA would have understood that by connecting to the intranet, the host system platform communicates with devices on the network, such as other computers. *See e.g.*, Dadhia, Fig. 2 (illustrating multiple devices connected to a local network); McArdle, Fig. 4A (illustrating multiple client systems on a local network). For example, Delco teaches that "[b]y default, DHCP protocol packets are enabled through the host-based packet filter." Ex.1005, 8:46-48. A POSITA would have been familiar with the dynamic host control protocol (DHCP), including the fact that a DHCP request message would be transmitted by the host to a DHCP server ("*one or more devices*") on intranet 22 ("*the network*"). See RFC 2131 at 13 ("The client broadcasts a DHCPDISCOVER message on its local physical subnet."). Delco also describes evaluating whether "whether a specific DHCP, DNS, or other server is known on or reachable through the network 48." Ex.1005, 9:10-17. A POSITA would have found it obvious for intranet 22 to have a DHCP server, a DNS server, or both. A POSITA would have further found it obvious for an application on Delco's host operating system to communicate with such servers.

159. Delco's applications 50 are referred to as "conventional applications." A POSITA would have thus understood that conventional applications communicate with other devices on a network. For example, Dadhia describes "exemplary scenarios ... as examples of the present technology for a network aware

firewall ... Abby attaches her laptop to her home network. Abby then opens the file sharing port on the laptop to share files between her laptop and her desktop." Accordingly, a POSITA viewing the teachings of Delco would have recognized that conventional applications communicate with other devices on a network.

160. Accordingly, because Delco's applications, which are described as conventional, run on the host system which has a network interface connected to the network, Delco renders obvious "*wherein the one or more applications or processes running in the host system are configured to communicate with one or more devices on the network*" as claimed.

13. Claim 11

[11.0] ***A method of network isolation in a networked computer system, the method comprising:***

161. To the extent the preamble is limiting, Delco renders it obvious.

162. **First**, consistent with the discussion above at [1.0], Delco teaches "*a networked computer system*" as claimed.

163. **Second**, Delco teaches a method by which a host system platform 12 within the networked computer system executes virtual machines which run guest operating systems in an isolated fashion. *See e.g.,* [1.5]. Thus, Delco teaches "*a method of network isolation*."

[11.1] ***providing a network and at least one computer system that is configured to connect to the network,***

164. First, consistent with the discussion above at [1.1], Delco teaches "*a network*" as claimed.

165. Second, consistent with the discussion above at [1.2], Delco teaches "*at least one computer system that is configured to connect to the network*."

[11.2] ***the computer system comprising a host system and a virtual system,***

166. For the reasons explained above at [1.3], Delco renders this limitation obvious.

[11.3] ***wherein the virtual system is a separate operating system or a software module operating on the computer system;***

167. For the reasons explained above at [1.4], Delco renders this limitation obvious.

[11.4] ***separating the host system from the virtual system using an internal firewall executed on the computer system;***

168. As explained above at [1.5], Delco teaches "*wherein an internal firewall is configured to separate the host system from the virtual system in the computer system.*" For similar reasons, Delco teaches "*separating the host system from the virtual system using an internal firewall executed on the computer system*" as claimed.

[11.5] ***implementing network isolation between the computer system and the network using a host-based firewall executed on the computer system;***

169. As explained above at [1.6], Delco teaches "a host-based firewall executed on the computer system is configured to implement network isolation

between the computer system and the network." For similar reasons, Delco teaches "*implementing network isolation between the computer system and the network using a host-based firewall executed on the computer system*" as claimed.

[11.6] ***providing at least one device configured to implement a network firewall or a web proxy; and***

170. Consistent with the discussion above at [1.7], Delco teaches "providing at least one device configured to implement a network firewall or a web proxy" as claimed.

[11.7] ***implementing network isolation, between one or more untrusted network destinations and the networked computer system, via the at least one device.***

171. As explained above at [1.8], Delco and Adams teach "*wherein the device comprises a processor and a memory configured to implement network isolation between one or more untrusted network destinations and the networked computer system.*" For similar reasons, at least Delco alone (as well as Delco and Adams) teach "*implementing network isolation, between one or more untrusted network destinations and the networked computer system, via the at least one device.*"

14. Claim 12

[12.1] ***The method of claim 11, wherein the internal firewall is configured to prevent data from being communicated between the virtual system and the host system without an explicit user input.***

172. For the reasons above at [2.1], Delco renders obvious "*wherein the*

internal firewall is configured to prevent data from being communicated between the virtual system and the host system without an explicit user input."

15. Claim 13

[13.1] ***The method of claim 11, further comprising: the host system storing data in a host data storage; and the virtual system storing data in a virtual data storage.***

173. **First**, as explained above at [3.1], Delco's host system, which utilizes a program execution space, teaches "whe*rein the host system is configured to store data in a host memory space*." For similar reasons, Delco's use of the host program execution space teaches "*the host system storing data in a host data storage*."

174. **Second**, for the reasons explained above at [3.1], Delco's virtual machines, which utilize guest execution space, teach "*the virtual system is configured to store data in a virtual memory space*." For similar reasons, Delco's use of the guest execution space teaches "*the virtual system storing data in a virtual data storage*."

16. Claim 14

[14.1] ***The method of claim 11, further comprising preventing malware introduced to the computer system from moving to another computer system of the networked computer system.***

175. As explained above at [4.1], Delco teaches "*wherein malware introduced to the computer system is prevented from moving to another computer system of the networked computer system*." For similar reasons, Delco teaches

"*preventing malware introduced to the computer system from moving to another computer system of the networked computer system*" as claimed.

17. Claim 15

[15.0] ***The method of claim 11, further comprising:***

176. *See* [11.0]-[11.7].

[15.1] ***communicating with the one or more untrusted network destinations via a first communication connection;***

177. For the reasons explained above at [5.1], Delco renders obvious "*communicating with the one or more untrusted network destinations via a first communication connection"*.

[15.2] ***communicating with one or more trusted network destinations via a second communication connection; and***

178. For the reasons explained above at [5.2], Delco renders obvious "*communicating with one or more trusted network destinations via a second communication connection*."

[15.3] ***enforcing separation of the first communication connection from the second communication connection.***

179. For the reasons explained above at [5.3], Delco renders obvious "*enforcing separation of the first communication connection from the second communication connection*."

18. Claim 16

[16.1] ***The method of claim 11, further comprising preventing, via the host-based***

firewall, lateral communication and movement of malware between the computer system and other devices on the networked computer system.

180. For the reasons explained above at [6.1], Delco renders obvious "*further comprising preventing, via the host-based firewall, lateral communication and movement of malware between the computer system and other devices on the networked computer system*."

19. Claim 17

[17.1] ***The method of claim 11, further comprising the at least one device preventing unauthorized communication between the computer system and the one or more untrusted network destinations.***

181. For the reasons explained above at [7.1], Delco renders obvious "*the at least one device preventing unauthorized communication between the computer system and the one or more untrusted network destinations*" as claimed.

20. Claim 19

[19.1] ***The method of claim 11, further comprising running one or more applications or processes in the virtual system that are configured to communicate with the one or more untrusted network destinations.***

182. For the reasons explained above at [9.1] and [9.2], Delco renders obvious "*running one or more applications or processes in the virtual system that are configured to communicate with the one or more untrusted network destinations*."

21. Claim 20

[20.1] ***The method of claim 11, further comprising running one or more***

applications or processes in the host system that are configured to communicate with one or more devices on the network.

183. For the reasons explained above at [10.1] and [10.2].

B. Ground 2: Claims 8 and 18 are obvious in view of Delco, Adams, and Dadhia.

1. Summary of Dadhia.

184. Dadhia, entitled "Network Aware Firewall," describes a technique that uses different security policies based on whether the detected network is public or private. Dadhia explains that "as increasing number of computers become mobile in 'home' and 'enterprise,' they connect to different types of networks at different times." Dadhia, 4:23-26.

185. To address this, Dadhia's "firewall keeps two sets of firewall policies locally. The first policy is associated with private (e.g., trusted) networks." Dadhia 4:59-61. "The second policy is associated with public (e.g., untrusted) networks." Dadhia, 4:63-64. Dadhia's network aware firewall is configured to "dynamically modify security settings on the laptop 120 in response to the network the computer is connected to." Dadhia, 5:36-38.

186. Dadhia provides an example in which file sharing is enabled on trusted networks but blocked or disabled on untrusted networks. When the computing system is on a home network that is "considered safe" (i.e., a trusted network), "a user may enable file sharing so that files may be shared between a

laptop computer system 120 and a home PC desktop computer system 115." Dadhia, 5:20-23. But "[w]hen the laptop computer system 120 is then used in a public network (e.g., at a local coffee house), the file sharing ... should be disabled to prevent other computer systems 160 from accessing information and/or applications on the laptop computer system 120." Dadhia, 5:26-30. Thus, "the firewall on the laptop 120 would disable file sharing when the laptop 120 is connected to the public network 150 and enable file sharing when the laptop 120 is connected to the home [i.e., trusted] network 110." Dadhia, 5:38-41.

187. Accordingly, Dadhia provides evidence of the obviousness of using different firewall policies depending on whether the computing system is connected to a trusted or an untrusted network.

2. Reasons to Combine Dadhia with Delco and Adams

188. A POSITA would have found it obvious for Delco's firewall (e.g., packet filter 32) to use different policies based on whether the host computing system is connected to a trusted or an untrusted network as taught by Dadhia. Doing so furthers Delco's goal of providing "a network traffic management system capable of performing firewall operations securely for multiple virtual machines and host operating system." Delco, 3:13-15. Indeed, Delco already provides the capability for changing firewall policy based on the detected network.

189. Delco explains that "the selection of policy rule sets by the policy

controllers 64, 66 is preferably further qualified by defined zones of application." Delco, 9:5-7. "[A] network zone is defined by the unique characteristics of the network 48 then accessible through the network interface controller 46." Delco, 9:7-9. Based on the detected network zone, a specific policy ruleset for the packet filters is used. "[S]tored configuration files are retrievable upon request by the policy controllers 64, 66 from the local datastore 18. The files or identifiers within the files allow policy rule sets to be identified by zone." Delco, 9:25-28. Delco's system "also actively monitors the network 46 for changes that might affect zone criteria matches." Delco, 10:44-46. Thus, if Delco's host computer system were to switch to a different network, the system would detect that change and update the firewall policy accordingly.

190. Delco's Fig. 1 illustrates the host system platform 12 connected to an intranet 22. That intranet is a protected network because it is behind firewall system 30. *See* Dadhia, 5:23-25 (explaining that a "home network can be considered safe, because the home router 125 may have a firewall that blocks malicious access attempts from the Internet 101."). But, if Delco's host platform system 12 were to connect to a different network (i.e., a different zone) that was not protected, a POSITA would have found it obvious for the policy for that network or zone to be more restrictive.

191. Dadhia provides an example in which on a protected network, file

sharing is enabled. However, on a public or unprotected network, file sharing is disabled. *See* Dadhia, 5:17-41. A POSITA would have thus found it obvious for Delco's host system platform to similarly allow file sharing on the intranet (i.e., protected network) and disallow file sharing on a public or untrusted network. Doing so provides the benefit of allowing desired functionality such as file sharing when on a trusted network while automatically blocking such features when connected to public, unsafe networks.

192. A POSITA would have had a reasonable expectation of success when combining the teachings of Dadhia with those of Delco. Dadhia explains that its network aware firewall "can be local to a computer system and can be implemented in hardware as well as software." Dadhia, 5:45-46. Delco's host system platform 12 is such a computer system: "A host system platform 12 includes a local processor 14 and memory 16 providing a program execution space." Delco, 4:55-56. Additionally, Delco's packet filter application is implemented in software "a packet filter application 32 is conventionally implemented and executed within the program execution space of the host system platform 12." Delco, 5:8-11. Thus, a POSITA would have expected that Dadhia's different firewall policies for different networks would be compatible with Delco's system.

193. Moreover, like Dadhia's system, Delco's system includes a

mechanism for accessing the network to which the host computer system is attached. Dadhia describes "a network accessor 302 for accessing a network connection. A connection determiner 304 determines the type of the connection." Dadhia, 5:47-50. Delco similarly includes a mechanism that determines the zone (i.e., network) to which the computer system is attached. "The current zone characteristics are preferably determined by the policy controller 64, 66 upon initialization, at periodic intervals, and in response to notices of a potentially significant change in the connected network as may be conventionally generated by the host operating system 42 and dedicated kernel 82." Ex.1005, 9:17-23. Accordingly, a POSITA would have expected success when combining the teachings of Delco and Dadhia to result in a network aware firewall that uses more strict policies on untrusted networks.

194. The resultant combination of Delco and Dadhia—Delco's packet filter application 32 providing stricter policies when connected to an untrusted network—is predictable because Dadhia's network aware firewall technique performs the same function within Delco that it does in Dadhia. For example, Dadhia explains that "the firewall on the laptop 120 would disable file sharing when the laptop 120 is connected to the public network 150 and enable file sharing when the laptop 120 is connected to the home network 110." Ex.1005, 5:38-42. It would have been obvious that this technique would work similarly in Delco. For

example, the packet filter application 32 on Delco's host system platform 12 would disable file sharing when connected to a public, untrusted network and enable file sharing when connected to the trusted intranet 22.

195. A POSITA would have thus been motivated to implement Delco's host-based firewall to have different policies for different environments as taught by Dadhia. Accordingly, the combination of Delco and Dadhia is merely the use of a known technique (Dadhia's different firewall policies for different networks) to improve similar devices (Delco's packet filter application 32) in the same way (using stricter policies when connected to an untrusted network).

3. Claim 8

[8.1] ***The networked computer system of claim 1, wherein the host-based firewall is configured to implement a first policy when the computer system is connected to the network and a second policy when the computer system is connected to another network,***

196. First, as explained above at [1.6], Delco's host system platform includes a packet filter application 32 ("host-based firewall"). The packet filter application 32 is controlled by policy controllers: "In the currently preferred embodiments of the present invention, **the individual policy controller components 64, 66 operate** independently of one another in managing the various virtual network interface controller components 70 and **host platform packet filter**." Delco, 8:7-11. The policies used by Delco's policy controllers 64, 66 are dependent on the detected "network zone."

> **The selection of policy rule sets by the policy controllers 64, 66 is preferably further qualified by defined zones of application**. For purposes of the present invention, a network zone is defined by the unique characteristics of the network 48 then accessible through the network interface controller 46. The applicable characteristics can include network distinguishing features such as the current IP address or addresses dynamically assigned to the controller 46, the local IP subnet, whether a specific DHCP, DNS, or other server is known on or reachable through the network 48, whether certain named computers or network components are accessible within defined hop counts, and the resolved DNS domain or fully qualified names of machines attached to the local subnet. The current zone characteristics are preferably determined by the policy controller 64, 66 upon initialization, at periodic intervals, and in response to notices of a potentially significant change in the connected network as may be conventionally generated by the host operating system 42 and dedicated kernel 82.

Delco, 9:5-22.

> Once retrieved, the policy role set configuration file is then parsed by the policy controller 64, 66 subject to the current zone characteristics and identified virtual network interface controller 70 being initialized. The first policy rule set in the policy role set configuration file, whose zone and controller identity criteria match the current zone characteristics and identified virtual network interface controller 70, is then passed to and installed 102 by the corresponding packet filter 74. To reduce false failures to identify a zone, the policy rule set may specify unique network characteristics to match, but also meta-

> characteristics, such as match percentage or particular combinations of characteristics that are require to qualify a zone match.

Delco, 9:63-10:8.

197. For the reasons explained above at VIII.B.2, it would have been obvious for Delco's packet filter to use different policies for trusted networks versus untrusted networks. Dadhia provides an example of such a model.

> In one embodiment, a firewall keeps two sets of firewall policies locally. **The first policy is associated with private (e.g., trusted) networks.** Examples of a private network include home, a virtual private network (VPN) to a trusted network, **a trusted Intranet**, etc. **The second policy is associated with public (e.g., untrusted) networks**. Examples of public networks include **WiFi at a local coffee shop, an airport connection**, etc. In one embodiment, the private policy can be only enabled when connected to the home network directly or through a VPN whether the network connected to is home or not can be determined automatically by analyzing the network attributes—and at all other times the public profile is enabled. This provides the simplest solution to users that are not interested in managing the profiles. In another embodiment, the user is asked whether a network is "public" or "private" every time a network connection is made.

Dadhia, 4:59-5:7.

198. Accordingly, Dadhia describes a "*first policy*" when connected to a trusted Intranet ("*the network*") and a "*second policy*" when connected to a coffee

shop or airport Wifi network ("*another network*").

199. Thus, because Delco describes setting the packet filter policy based on the network zone, and Dadhia describes two different policies for untrusted and trusted networks, Delco and Dadhia render obvious "*wherein the host-based firewall is configured to implement a first policy when the computer system is connected to the network and a second policy when the computer system is connected to another network*" as claimed.

[8.2] ***wherein the first policy permits one or more communications between the host system of the computer system and one or more resources on the network and,***

200. Dadhia's gives an example in which the "first policy" for the trusted network allows file sharing.

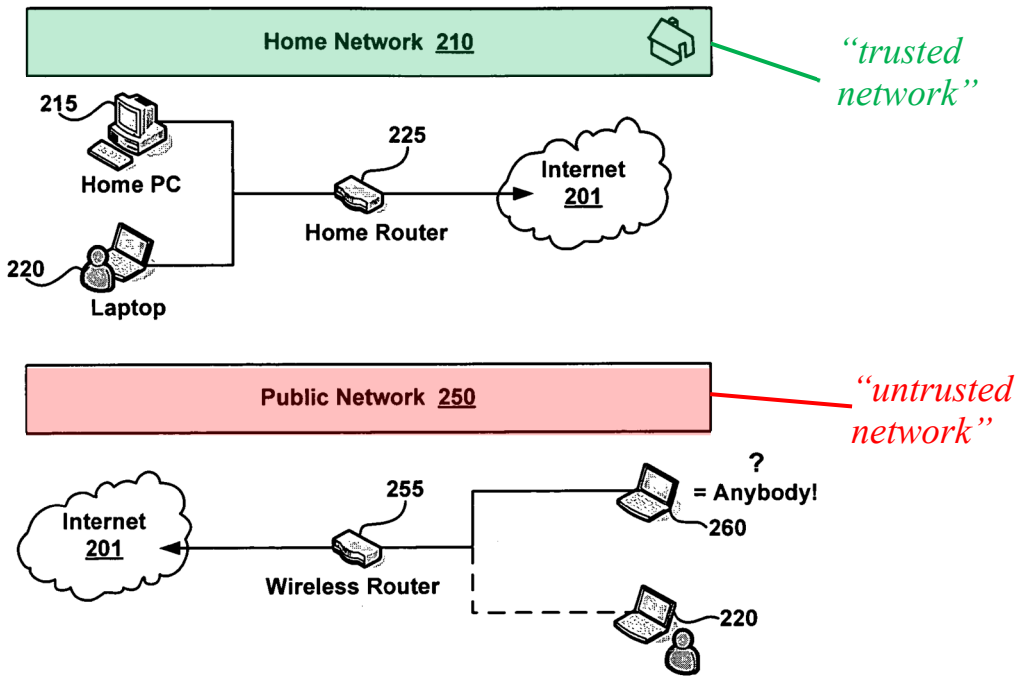
> FIG. 2 is a diagram of an exemplary home network 210 and an exemplary public network 250 used in accordance with embodiments of the present technology for network aware firewall. When connected to a home network 110, **a user may enable file sharing so that files may be shared between a laptop computer system 120 and a home PC desktop computer system 115**. The home network can be considered safe, because the home router 125 may have a firewall that blocks malicious access attempts from the Internet 101.

Dadhia, 5:17-25.

> Embodiments of the present technology for a network aware firewall dynamically modify security settings on the laptop 120 in response to

> the network the computer is connected to. For example, in the case described above, the firewall on the laptop 120 would disable file sharing when the laptop 120 is connected to the public network 150 and **enable file sharing when the laptop 120 is connected to the home network 110**.

Dadhia, 5:35-42.



Dadhia, Fig. 2 (annotated).

201. Accordingly, by enabling file sharing with another system on the trusted network, "*the first policy permits one or more communications between the host system of the computer system and one or more resources on the network*" as claimed.

202. Thus, because Delco describes a packet filter application 32 that may

use a first policy as described by Dadhia, in which file sharing is enabled on a trusted network, Delco and Dadhia render obvious "*wherein the host-based firewall is configured to implement a first policy when the computer system is connected to the network and a second policy when the computer system is connected to another network*" as claimed.

[8.3] ***the second policy blocks communications egressing from the host system of the computer system but allows communications egressing from the virtual system.***

203. First, Dadhia gives an example in which the "*second policy*" for the untrusted network disables file sharing.

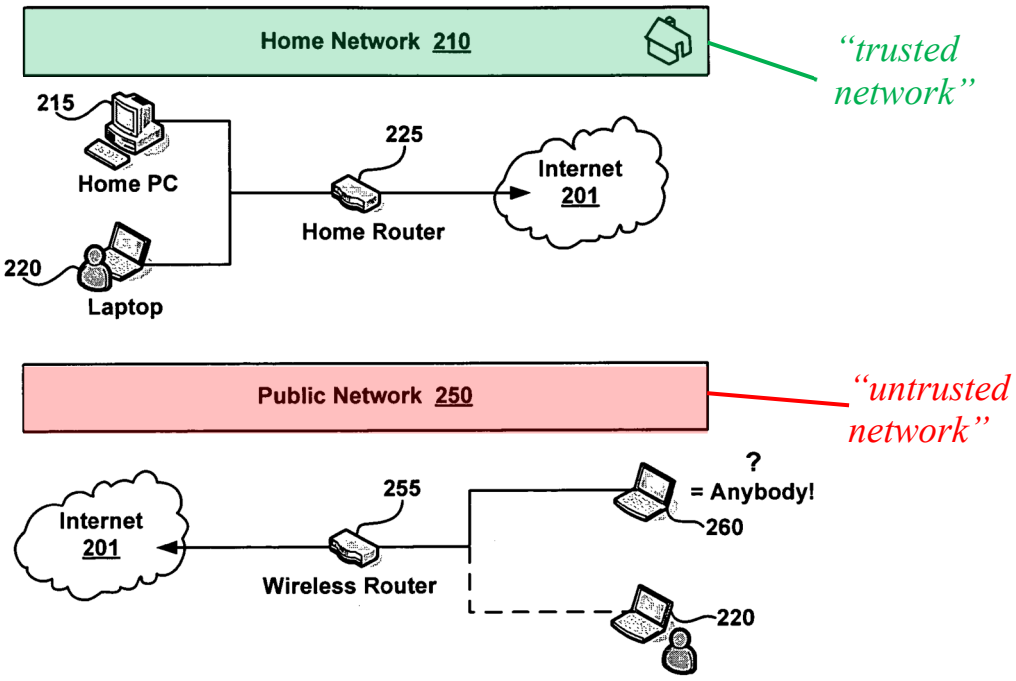
> When the laptop computer system 120 is then used in a public network (e.g., at a local coffee house), **the file sharing** enabled within the home network 110 **should be disabled to prevent other computer systems 160 from accessing information and/or applications on the laptop computer system 120**. Within the public network environment 150, the laptop is sharing the network with unknown computers. Also the security infrastructure in some public networks is largely non-existent.

Dadhia, 5:26-34.

> Embodiments of the present technology for a network aware firewall dynamically modify security settings on the laptop 120 in response to the network the computer is connected to. For example, in the case described above, **the firewall on the laptop 120 would disable file sharing when the laptop 120 is connected to the public network 150**

and enable file sharing when the laptop 120 is connected to the home network 110.

Dadhia, 5:35-42. [3]



Dadhia, Fig. 2 (annotated).

204. Accordingly, by disabling file sharing with another system on the untrusted network, "*the second policy blocks communications egressing from the host system of the computer system*" as claimed.

205. **Second**, while Delco's packet filter application 32 ("*host-based*

[3] I note that Dadhia's description uses different reference numerals in the figures than it does in the text. For example, the described home network 110 and laptop 120 are illustrated as home network 210 and laptop 220.

firewall") disables file sharing for the host, , the filter application 32 is not responsible for communication between the virtual machine and Internet—thus allowing incoming and outgoing communication between the virtual machine and external networks. In other words, even when the packet filter 74 of the virtual machine prevents egress, the filter application 32 on the host would still "*allow[] communications egressing from the virtual system.*"

206. To the extent the claim requires that there be an actual egress through the virtual machine, it would have been obvious for the virtual machine to allow communication over the untrusted network, even when communication from applications running directly on the host system are blocked by the host's packet filter application. For example, as explained above at [5.0]-[5.3], Delco uses different communication sessions for the host system and the virtual systems. Thus, disabling file sharing on the communication session for the host system does not affect the policies of the virtual system. Indeed, Delco explains that "The policy rule sets can be further individualized based variously on the identity, role, and other operative characteristics of the **virtual machines**, **the host computer system**, and the connected network, thereby allowing different levels of qualified security." Thus, Delco allows for different policies between the host computer system and the virtual machines.

207. A POSITA would have understood that in Delco's system, the virtual

machine packet filter policy may be more or less restrictive than the host packet filter policy. Delco explains that "[p]rograms executed in one virtual machine are essentially unaffected by and, conversely, essentially unable to affect the execution of programs in other virtual machines." Ex.1005, 1:40-43. Similarly, Delco's virtual machines are "managed through separate virtual machine execution processes" and are "effectively isolated from one another and from the host operating system 42 as a function of the applied virtual machine world context switch." Ex.1005, 6:31-45. This separation between the virtual machines and the host system thus allows for variation in permissiveness of their respective packet filters. A POSITA would have appreciated that the risks associated with contacting an untrusted destination are reduced by performing that communication from a virtual machine (e.g., if a virus is acquired, it will infect only that individual virtual machine and not the other virtual machines or the host system itself). As a result, it would have been obvious for Delco's virtual machine to have a more permissive security policy than the host system, such that an application running in Delco's virtual machine would be able to communicate with an untrusted destination even when an application running directly on the host system would be prevented from such communications.

208. Accordingly, Delco's multiple packet filters act independently to restrict or allow different types of traffic.

The policy controllers 64 preferably operate to initially and dynamically direct the configuration of the different packet filter layer 74 instances as implemented in the virtual machines $54_{1\text{-}N}$. The policy controller 66 preferably performs equivalently with respect to the platform packet filter application. Policy information is initially defined in terms of policy rule sets designateable as applicable to the host operating system 42, specified individual or groups of virtual machines $54_{1\text{-}N}$, or to specific packet filters 74. In the preferred embodiments of the present invention, the policy rule sets determine for the applicable packet filter 74 the network traffic that is to be restricted on a per-packet basis or that is to be monitored and conditionally restricted based on stateful analysis. Factors defined by rule sets for evaluation can include source and destination addressing, whether based on MAC, IP, IPX, or similar addresses or address ranges, the type of network traffic, such as broadcast, unicast, and multicast, the packet transmission protocol, such as ARP, IPX, IP, TCP, UDP, HTTP, and the like, the packet designated source and destination ports, including whether falling within privileged and unprivileged port ranges, the direction of the network traffic, and packet size. By default, DHCP protocol packets are enabled through the host-based packet filter as desired to support basic network to platform functions. Factors defined by the rule sets for consideration under stateful analysis can include the frequency of traffic related by source or type, and keywords, signatures, and other defining content discernable within the headers and payload content of individual and statefully related network traffic. Policy rule sets may further define process qualifications, including for example, the number of exceptions permitted before a particular

network traffic stream is terminated and whether certain network traffic is to be logged.

Delco, 8:25-57.

Since it would be obvious for a virtual machine packet filter to be more permissive than the packet filter for the entire host system, Delco and Dadhia render obvious that "*the second policy blocks communications egressing from the host system of the computer system but allows communications egressing from the virtual system*" as claimed.

4. Claim 18

[18.0] ***The method of claim 11, further comprising:***

209. *See* [11.0]-[11.7].

[18.1] ***implementing a first policy for the host-based firewall when the computer system is connected to the network, the first policy for the host-based firewall permitting one or more communications between the host system of the computer system and one or more resources on the network; and***

210. For the reasons explained above at [8.1], Delco renders obvious "*implementing a first policy for the host-based firewall when the computer system is connected to the network, the first policy for the host-based firewall permitting one or more communications between the host system of the computer system and one or more resources on the network*."

[18.3] ***implementing a second policy for the host-based firewall when the computer system is connected to another network, the second policy for the host-based firewall blocking communications egressing from the host system of the computer system but allowing communications egressing from the virtual***

system.

211. For the reasons explained above at [8.2], Delco renders obvious "*implementing a second policy for the host-based firewall when the computer system is connected to another network, the second policy for the host-based firewall blocking communications egressing from the host system of the computer system but allowing communications egressing from the virtual system*."

IX. CONCLUSION

212. This declaration and my opinions herein are made to the best of my knowledge and understanding, and based on the material available to me, at the time of signing this declaration. I declare that all statements made herein on my own knowledge are true and that all statements made on information and belief are believed to be true, and further, that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 or Title 18 of the United States Code.

Date: 7/25/24

Prashant Shenoy