

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

Netskope, Inc.

Petitioner,

v.

Fortinet, Inc.

Patent Owner.

Case IPR2023-00456
Patent No. 11,036,856

**DECLARATION OF DR. JOHN BLACK JR.
IN SUPPORT OF PATENT OWNER'S PRELIMINARY RESPONSE**

1. I have been retained on behalf of Fortinet, Inc. (“Patent Owner”) in connection with the above-captioned *inter partes* review (IPR). I have been asked to provide my opinions regarding Fortinet’s Patent Owner Preliminary Response. I am being compensated for my time at the rate of \$625 per hour. I have no interest in the outcome of this proceeding.

2. In preparing this Declaration, I have reviewed and am familiar with the Patent Owner’s Petition submitted for IPR2023-00456, U.S. Patent No. 11,036,856, (“the ‘856 patent”) and its file history, and all other materials cited and discussed in the Petition (including the declaration of Petitioner’s expert Dr. Michael Franz), as well as the materials cited and discussed in this Declaration.

3. The statements made herein are based upon the cited documents as well as my own knowledge and experience in the field. This Declaration represents only the opinions I have formed to date. I may consider additional documents as they become available or other documents that are necessary to form my opinions. I reserve the right to revise, supplement, or amend my opinions based on new information and on my continuing analysis.

I. QUALIFICATIONS

4. My qualifications can be found in my Curriculum Vitae (*see infra* Ex. 2002), which includes my detailed employment background, professional experience, and list of technical publications and patents.

5. I am an Associate Professor of Computer Science at the University of Colorado, Boulder. I received a B.S. in Mathematics and Computer Science from the California State University at Hayward (now "California State University, East Bay") in 1988. I received an M.S. in Computer Science in 1997, and a Ph.D. in Computer Science in 2000, both from the University of California at Davis.

6. I have taught more than 60 classes in computer science, on subjects including data structures, algorithms, networking, operating systems, software engineering, security, cryptography, discrete mathematics, and quantum computing. I have authored or coauthored more than 20 publications, primarily on issues relating to computer security. I have been involved with computers for over 40 years in both commercial and academic capacities.

7. My earliest interest was in networks and security. My first memories in this regard were around 1975 when a group of friends and I learned about the telephone network and its security. A few years later, personal computers became available and I spent most of my free time studying, programming, and modifying them. I worked extensively with various networking products throughout the 1980s, and developed an interest in cryptography soon thereafter. Although my undergraduate institution had no courses in cryptography or security in the 1980s, I decided to pursue self-study at the time, and opted to double major in Computer Science and Mathematics because cryptography is a blend of these two subject areas.

8. After earning my B.S. degree in 1988, I worked for six years at Ingres Corp as a software developer, writing and reviewing code written in C. My work primarily was directed at transaction logging, data type support, and security.

9. In 1995, I began my Ph.D. at UC Davis under cryptographer Phillip Rogaway. My area of focus was cryptography and security and my research involved encryption, authentication, hash functions, and network security.

10. After graduation I took a position as an Assistant Professor at the University of Nevada at Reno. In the Fall of 2001, I taught the networking class there, which included coverage of Ethernet, interior gateway protocols, exterior gateway protocols, ARP, DHCP/BOOTP, IP, UDP, TCP, HTTP, SMTP and other protocols. In 2001, a graduate student and I looked at the security of ARP and invented a new protocol "AuthARP" to add security to the protocol.

11. In 2002, I moved to the University of Colorado at Boulder where I am currently employed. In the Fall of 2002, I co-designed and co-taught a new course called "Foundations of Computer and Network Security," which included descriptions of security issues around both wired and wireless security challenges, mostly for public-facing network services including the world-wide web. I have taught this class seven more times since then, including modern topics such as wireless networking, the Internet of Things, and so forth.

12. In my career at the University of Colorado, I have published several more papers in the area of cryptography and network security. I have taught more than 30 courses in network security and cryptography, and have graduated several PhD students in these areas.

13. I also have worked for a consulting company at times, writing software on contract basis. Although most projects are covered by NDAs, many involved networks and/or aspects of computer security.

14. In 2011, I began technical consulting for a local company called Cardinal Peak, which focuses primarily on video encoding and delivery systems. My work for Cardinal Peak has largely been directed to video encoding, transcoding, compression, encryption, and DRM, but also has included networking projects and embedded device work. For example, I designed the security system for the Pro1 smart thermostat, implemented the DRM for the Yonder Music App, worked on 802.1X code for smart dog collars, and helped design the cryptography used in Fitbit devices for wireless transfer of a Fitbit watch to a phone or laptop.

15. In 2016, I took a two-and-a-half year leave of absence from the University of Colorado to start a company named "SecureSet" in Denver, Colorado. The objective of SecureSet is to take reasonably proficient technical people and turn them into computer and network security specialists via five months of intensive training. SecureSet was sold to WeWork in 2019 and continues to offer computer

security classes today. In 2018 I returned to my position at the University where I remain employed to the present day.

II. LEGAL UNDERSTANDING

A. The Person of Ordinary Skill in the Art

16. I understand that a person of ordinary skill in the relevant art at the time of the invention (also referred to herein as “POSITA”) is presumed to be aware of all pertinent art, thinks along conventional wisdom in the art, and is a person of ordinary creativity—not an automaton.

17. I have been asked to consider the level of ordinary skill in the field that someone would have had at the time the claimed invention was made. In deciding the level of ordinary skill, I considered the following:

- the levels of education and experience of persons working in the field;
- the types of problems encountered in the field; and
- the sophistication of the technology.

18. A person of ordinary skill in the art relevant to the ‘856 patent at the time of the invention would have had a Bachelor of Science degree in electrical engineering and/or computer science, and two years of work or research experience in the fields of network and data security, or a Master’s degree in computer

engineering and/or computer science and one year of work or research experience in related fields.

19. I have reviewed the declaration of Dr. Franz, including his opinions regarding the Person of Ordinary Skill in the Art. My description of the level of ordinary skill in the art is essentially the same as that of the Dr. Franz. The opinions set forth in this Declaration would be the same under either my or Dr. Franz's proposal.

20. I am well-qualified to determine the level of ordinary skill in the art and am personally familiar with the technology of the '856 Patent. By 2000, I had obtained a PhD in Network and Computer Security and therefore I was a person of at least ordinary skill in the art at the time of the priority date of the '856 patent, September 16, 2018.

B. Legal Principles

21. I am not a lawyer and will not provide any legal opinions. Though I am not a lawyer, I have been advised that certain legal standards are to be applied by technical experts in forming opinions regarding the meaning and validity of patent claims.

1. Obviousness

22. I understand that to obtain a patent, a claimed invention must have, as of the priority date, been nonobvious in view of prior art in the field. I understand

that an invention is obvious when the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art.

23. I understand that to prove that prior art, or a combination of prior art, renders a patent obvious, it is necessary to: (1) identify the particular references that singly, or in combination, make the patent obvious; (2) specifically identify which elements of the patent claim appear in each of the asserted references; and (3) explain how the prior art references could have been combined to create the inventions claimed in the asserted claim.

24. I understand that a patent composed of several elements is not proved obvious merely by demonstrating that each of its elements was, independently, known in the prior art, and that obviousness cannot be based on the hindsight combination of components selectively culled from the prior art to fit the parameters of the patented invention.

25. I also understand that a reference may be said to teach away when a person of ordinary skill, upon reading the reference, would be discouraged from following the path set out in the reference, or would be led in a direction divergent from the path that was taken by the applicant. Even if a reference is not found to teach away, I understand its statements regarding preferences are relevant to a

finding regarding whether a skilled artisan would be motivated to combine that reference with another reference.

2. *My Understanding of Claim Construction Law*

26. I have been informed that patent claims are construed from the viewpoint of a person of a POSITA and that patent claims generally should be understood consistent with their ordinary and customary meaning at the time of the invention. A review of the patent claim language, the patent specification, and its prosecution history are also necessary to determine the proper meaning and scope of the term at issue.

27. I have further been informed that in the specification and prosecution history the patentee may define a claim term in a way that differs from the ordinary and customary meaning. I understand that during prosecution of the patent before the U.S. Patent and Trademark Office, the Applicant may make representations or provide definitions of terms that may affect the scope of the patent claims. In particular, the Applicant may, during the course of prosecution, limit the scope of the claims to overcome prior art and/or disavow claim coverage by making clear and unambiguous statements to that effect.

28. I have been informed that a POSITA may, among other things, consider dictionaries, publications, other patents, and treatises (*i.e.*, “extrinsic evidence”). I understand that extrinsic evidence may generally not be relied on if it contradicts the

meaning of claim language provided by the intrinsic evidence, such as express definitions given for terms in the specification.

III. OVERVIEW OF THE ‘856 PATENT

A. The ‘856 Patent Specification

29. The ‘856 Patent is entitled “Natively Mounting Storage for Inspection and Sandboxing in the Cloud.” It includes systems and methods for “scanning and/or sandboxing files” in order to “protect users from accessing infected files by natively mounting public file stores.” ‘856 Patent at Abstract. In one embodiment, a network security device makes a determination whether an untrusted file stored in a first repository of a public cloud file store is free of malicious content; if that check is affirmative, the invention copies the file to a second repository so that users can access it. *Id.*

30. The ‘856 Patent Specification explains that systems at the time of the invention were often “reactive” in the sense that such systems were capable of verifying data and could perhaps take steps to remove “bad” files once they were identified, though in many cases, the removal may have already been too late. *Id.* 1:51-56. It was also known that there were inefficiencies in how files were scanned in the cloud storage context: either files had to be duplicated in order to collect a sample for analysis, or cloud storage was attached to a VM in the cloud to allow files to be sent to a sandbox for analysis, or cloud storage was attached to a VM that ran

a static antivirus scan. *Id.* 2:5-11. As the specification explains, these approaches require significant resources (including storage and network resources) and/or may not be effective at finding all threats (in particular, AV scans will not detect so-called “zero day” attacks). Specifically, a “zero day” attack is an attack that exploits brand new vulnerabilities, such as a vulnerability in a piece of software that has not yet been patched (perhaps because the exploit is not was only recently known to the public or the software’s developer)—traditional AV software will be unable to detect new forms of malware if their signatures or their exploits are not known in advance. The specification also highlights the added challenges inherent in cloud-based computing environments including reverse engineering due to the fact that images are encapsulated, and the higher probability of misconfigurations of virtual environments. *Id.* 2:18-32.

31. The approach taken in the ‘856 Patent to address the problems above is to continuously scan and/or sandbox files using a first network security device to perform security operations on files before copying them to a second file store, if they have cleared the security operations. In one embodiment, this approach is implemented as depicted in simplified form in Figure 1B of the ‘856 Patent.

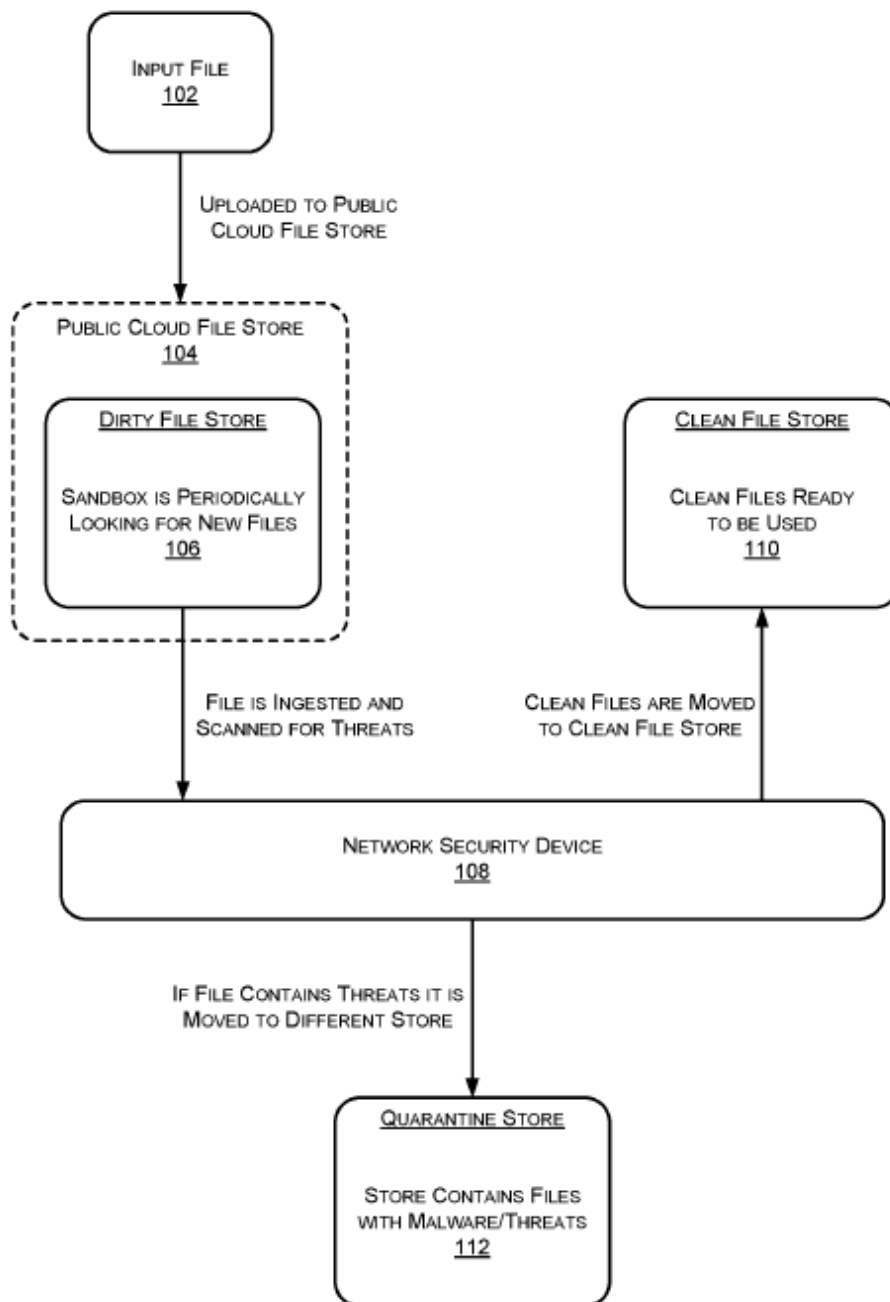


FIG. 1B

32. Here, input file 102 is initially untrusted when deposited in the “dirty file store” 106 on the public cloud file store 104. The public cloud file store 104 can

be natively mounted on the network security device 108, which in turn is monitoring the dirty file store 106 for new input files (or alternatively can receive a notification via an API that there are new files present). The network security device 108 can then perform various security operations on a given input file to determine whether it contains malicious content. *Id.* 7:40-8:25. If a given input file 102 is free from malicious content, the network security device 108 transfers or copies the file to the clean file store 110. *Id.* 8:53-57.

B. The ‘856 Patent Claims

33. I understand Petitioner challenges all 28 claims of the ‘856 patent.

Claim 1 recites:

1. A secure data transfer system comprising:

a non-transitory storage device having embodied therein one or more routines operable to prevent users of an enterprise network from accessing malware infected files that are stored within public cloud file stores; and

one or more processors coupled to the non-transitory storage device and operable to execute the one or more routines, wherein the one or more routines include:

an untrusted file processing module, which when executed by the one or more processors:

accesses an untrusted file stored within a first repository of a public cloud file store, wherein the public cloud file store is natively mounted on a network security device that is protecting the enterprise network and wherein the users do not have read access to the first repository of the public cloud file store; and

causes the network security device to make a determination regarding whether the untrusted file is a clean file that is free of malicious content by applying one or more security checks to the untrusted file; and

a clean file transfer module, which when executed by the one or more processors, makes the clean file accessible to the users by, when a result of the

determination is affirmative, copying the clean file from the first repository to a second repository that is accessible to the users.

34. Claim 1 of the ‘856 patent is exemplary. The claim encompasses the ideas from the specification involving an untrusted file stored in a first repository of a public cloud file store, natively mounted on a network security device. Users cannot access this first repository, since the untrusted file is yet to be evaluated for its trustworthiness. The network security device then applies various security checks to the untrusted file to determine whether it is “clean” or not. Clean files are then made accessible to users by copying the clean file to a second repository that users may access.

IV. OVERVIEW OF CERTAIN ART

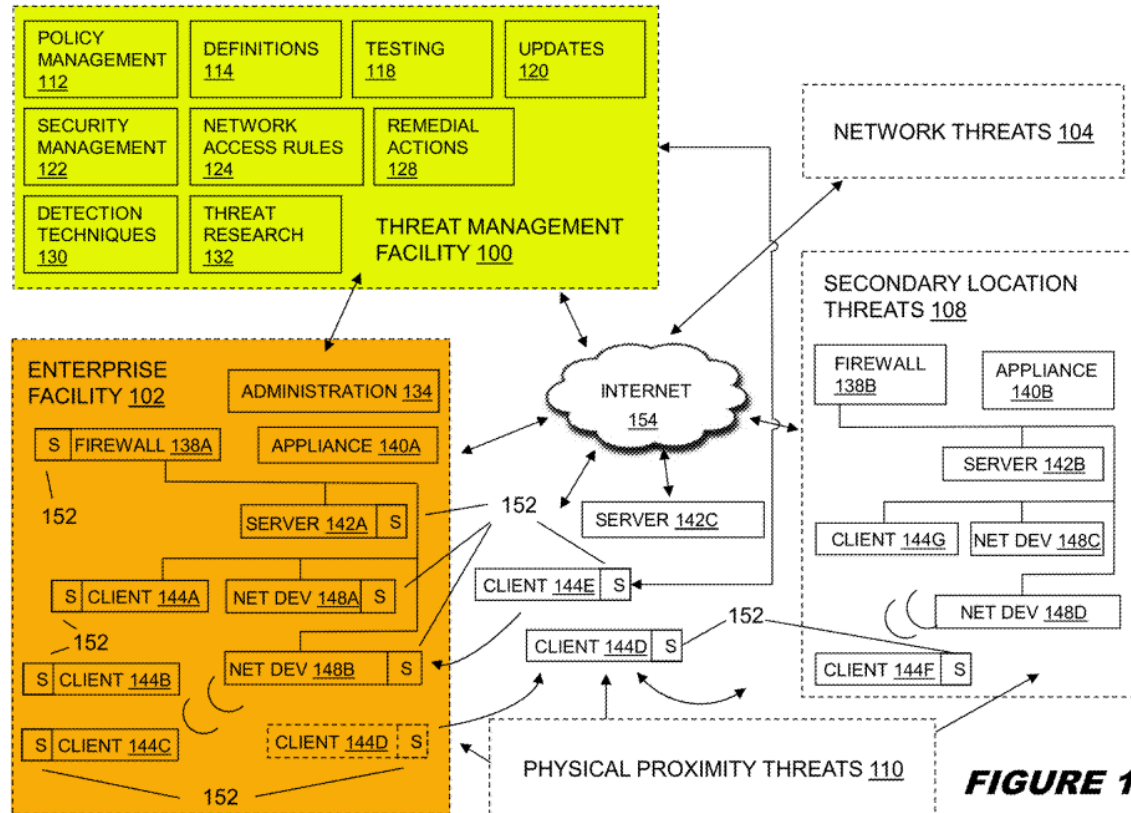
35. Although the petition cites to a number of references, the main references it relies on for its challenges against the independent claims are Burdett, FortiSandbox, Sharpe, Seese, and Kamalapuram. The petition’s first ground asserts the combination of Burdett and FortiSandbox. The petition’s second ground asserts the combination of Sharpe, Seese, and Kamalapuram. Below, I summarize Burdett, FortiSandbox, Sharpe, Seese, and Kamalapuram.

C. U.S. Patent No. 11,281,775 (“Burdett”) (Ex. 1004)

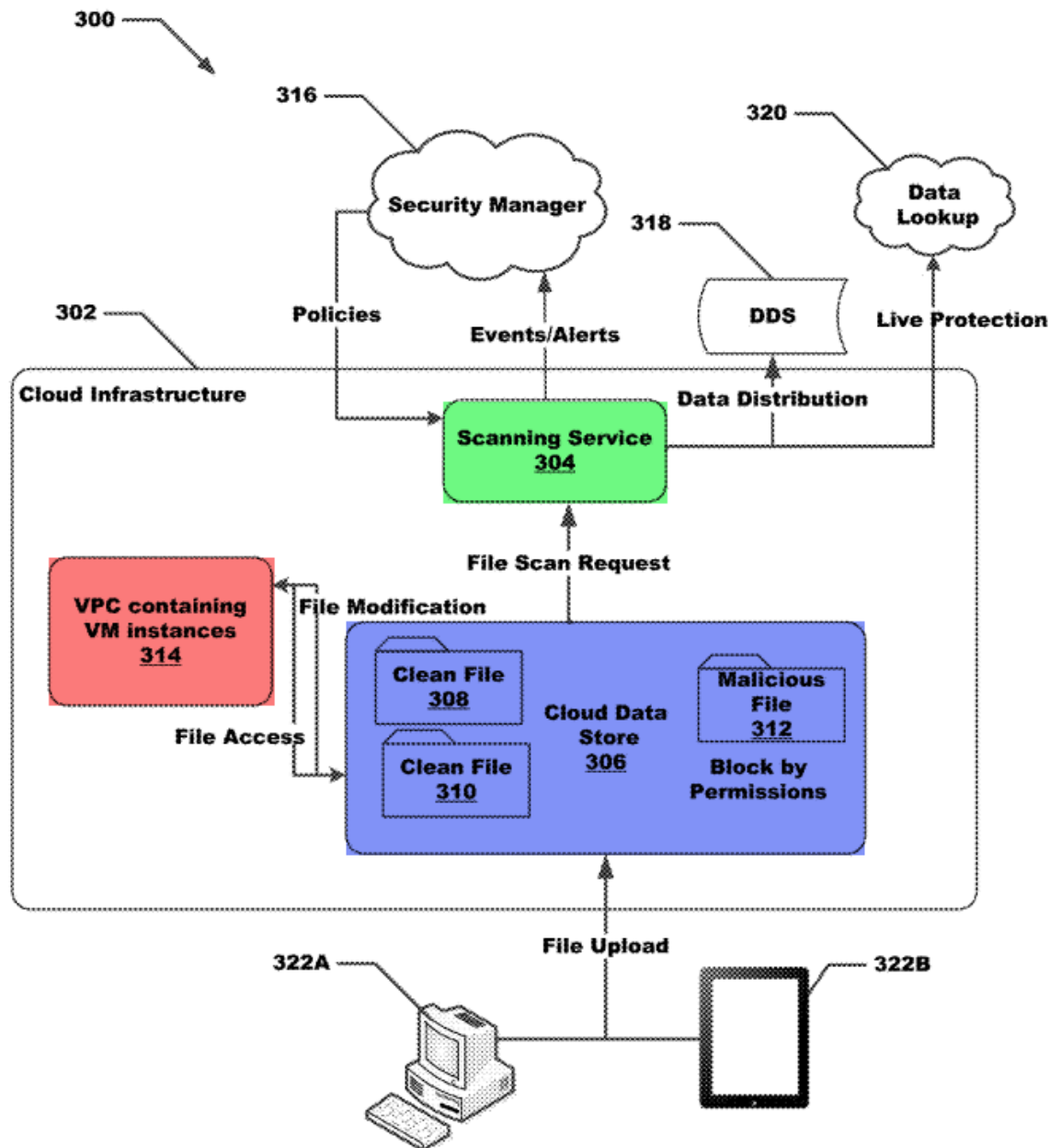
36. Burdett’s Figure 1 shows a “threat management facility 100” (yellow) protects a traditional “enterprise facility 102” (orange) by, for example, “provid[ing]

controlled access to the enterprise facility 102 networks.” Burdett at 11:57-58. This

Figure 1 is an example of a traditional, facility-based network.



37. Burdett’s Figure 3 shows “cloud infrastructure 302,” which includes “scanning service 304” for scanning files in “cloud data store 306” to prevent “applications instantiated on virtual machines within the virtual private cloud [VPC] 314” from accessing malicious files. Burdett at 23:33-38, 26:44-49.



38. Burdett's Figure 3 and accompanying description does not reference "threat management facility 100" or "enterprise facility 102" in Figure 1. Similarly, Burdett's Figure 1 and accompanying description does not reference "cloud infrastructure 302" or "VPC 314," "cloud data store 306," and "scanning service 304" found therein.

D. FortiSandbox (Ex. 1005)

39. FortiSandbox is an “Administration Guide” for FortiSandbox Version 2.3.2. FortiSandbox is designed to “execute[] suspicious files in the VM [virtual machine] host module to determine if the file is High, Medium, or Low Risk based on behavior observed in the VM sandbox module.” FortiSandbox at 8. FortiSandbox processes files that are uploaded on-demand, obtained by sniffing traffic passing on a network, received from another Fortinet appliance (e.g., FortiGate or FortiMail) or found on a remote network share. *Id.* at 88. FortiSandbox can scan files on a network share by specifying a “Mount Type,” “Server Name/IP,” and “Share Path,” and other variables. *Id.* at 114-15.

E. U.S. Patent No. 9,811,662 (“Sharpe”) (Ex. 1006)

40. The Petition focuses on Figure 42 from Sharpe:

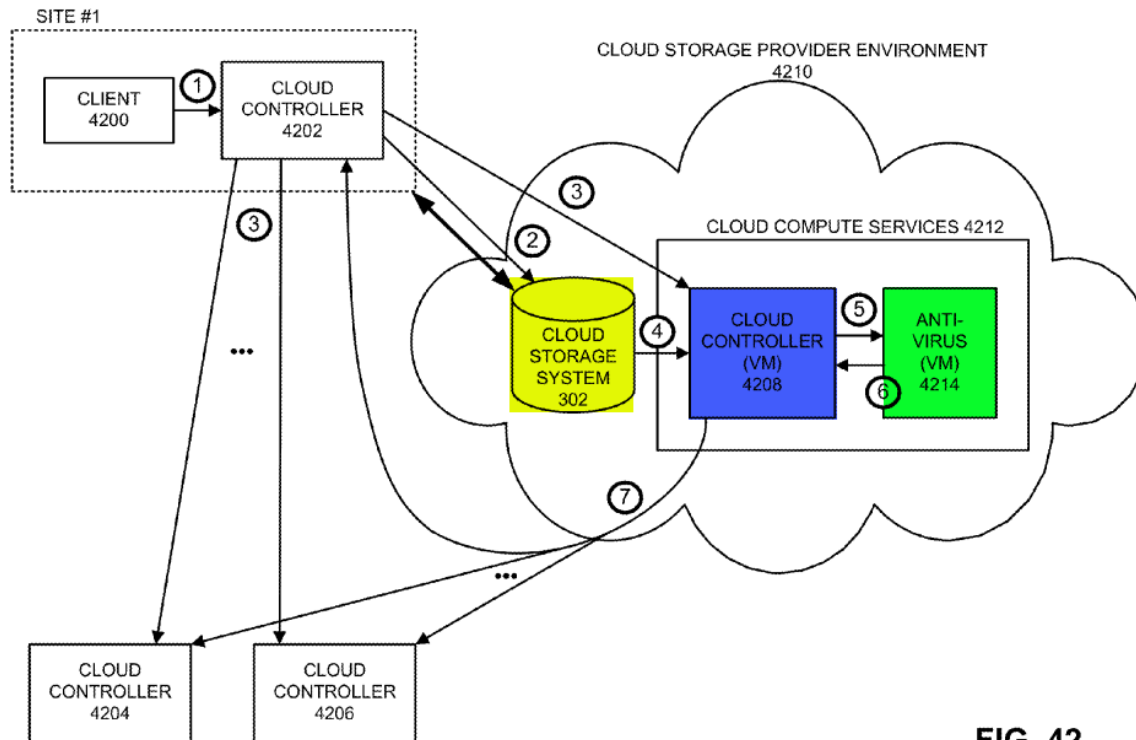
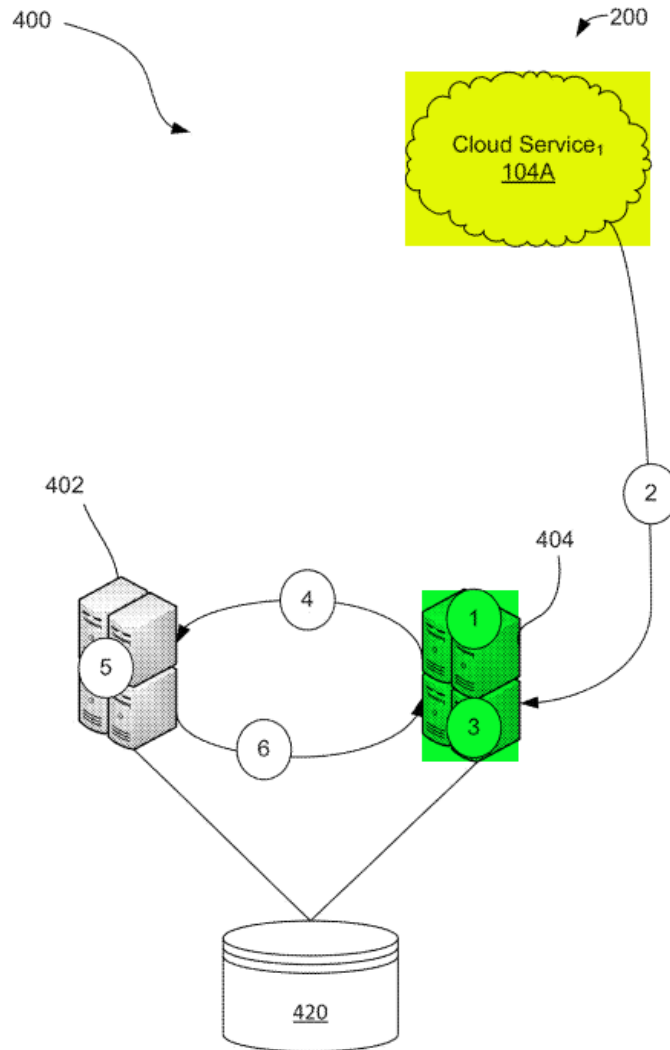


FIG. 42

41. In Figure 42, a client can upload a file to “cloud storage system 302” (yellow) via “cloud controller 4204.” Initially, “the file is not marked as clean,” and “other cloud controllers ... do not yet allow clients to actually access the new (or modified) file.” Sharpe at 97:27-36. The file is then scanned by “anti-virus service 4214” (green) via “cloud controller 4208” (blue). *Id.* at 97:37-53. If the file is clean, its metadata “is updated to indicate this clean status.” *Id.* at 97:56-58.

F. U.S. Patent Appl. Pub. No. 2015/0100357 (“Seese”) (Ex. 1007)

42. Seese describes “scan[ning] of files stored on a cloud service.” Seese at [0064].



43. Specifically, “DLP [data loss prevention] discovery server 404” (green) executes a “discovery scan task” based on “DLP rules” for “identify[ing] confidential data. Such as Social security numbers, financial data (e.g., credit card number, magnetic stripe data, and the like), username and password pairs, proprietary information, licensed data (e.g., copy righted works), or any other data that may compromise the enterprise if the data is misappropriated.” See at [0054], [0068]. To do so, the DLP discover server, “mounts” one “user share” at a time,

“pulls files from the user share on the cloud service 104A” (yellow), and “inspects the files against configured policies and rules to generate an incidents report.” *Id.* at [0067]-[0068]. “[A]s incidents are detected by the DLP discover server 404[,] they are sent to an incident database 420 and are viewable from the enforce server 402 for incidents review and remediation as necessary.” *Id.* at [0070].

G. U.S. Patent Appl. Pub. No. 2020/0050686 (“Kamalapuram”) (Ex. 1008)

44. Kamalapuram describes “a cloud only system” where uploaded files are first “scanned by a security analysis system.” Kamalapuram at [0121]. Initially, an uploaded file “may be stored in an isolated area of the network storage system,” and “[u]sers may be prevented from accessing the file, or the file may be limited to a view only mode until all scanning tools ... have been run.” *Id.* at [0122]. As part of that scan, “the network security analysis system” then “determine[s] whether the file may be accessed or downloaded by other hosts or other users.” *Id.* at [0121]. Once the file has been scanned, it “may be moved to a general storage area.” *Id.* at [0122].

V. CLAIM CONSTRUCTION

45. In my opinion, no claim terms presently require construction. I reserve the right to proffer a construction for a claim term at a later point if I believe it pertinent to any particular dispute.

VI. GROUND 1: BURDETT AND FORTISANDBOX

A. Argument 1: No Obviousness Theory for “Copying”

46. In my opinion, Petitioner has not shown that Burdett’s disclosure of “moving” a file discloses or renders obvious “copying” a file. All independent claims of the ’856 patent require “**copying** the clean file from the first repository to a second repository.” In contrast, dependent claims 12 requires “**moving** the untrusted file from the first repository to a third repository” for checking. The terms “copying” and “moving” have different meanings and require that different actions be performed on files—copying vs. moving. “[C]opying” information requires creating, at least temporarily, a second copy of portions of that information. “[M]oving” merely requires changing the location of the information within a file system, regardless of what mechanism is used. A POSITA would understand that, in a file system, moving and copying a file are different actions.

47. That a file has been “moved” does not inform a POSITA whether that file has been “copied,” since it is possible to change a file’s location by modifying the file system structure (thereby “moving” the file within that structure) without ever creating a second copy of even portions of that file (i.e., without any “copying”). To give a concrete example, in POSIX systems, a “mv” command is implemented via the “rename” system call that is very efficient if the source and destination are in the same file system. Specifically, in the case that the origin of a “mv” command

is the same file system as the destination of that mv command, a POSIX system will merely adjust the directory entries to reflect the new location of the moved file, but the contents on the disk will remain in place. In other words, only the file metadata (i.e. the location of the file) is updated, whereas the file contents on the disk remains the same. By contrast, when performing a “copy” command, at least part of the file’s contents is written to a new location.

48. Moreover, there are in some instances good reasons to “move” a file without performing any “copying.” For example, “moving” a file by modifying the file system structure, as described in the POSIX example above, generally requires only a constant amount of processing, since only the metadata is updated and therefore the processing necessary to make the modification is independent of the particular file being moved (and thus also independent of the file’s size). By contrast, the amount of processing required to “copy” a file generally increases linearly with file size, since the larger the file, the more data there is to copy. In other words, moving data is very fast, copying data (especially large files) can be very slow.

49. Looking at Petitioner’s arguments, Petitioner does not establish that Burdett discloses moving a file by copying it. Petitioner states that Burdett describes “changing the storage location of the file” based on Burdett’s disclosure that “the scanning service may move files ... in order to control access.” Pet. at 33 (quoting

Burdett at 25:26-27). Petitioner then concludes that, “[g]uided by Burdett, a POSITA would have found it obvious that one alternative to control access to a contaminated file was to **move** the file to a different location.” Pet. at 33.

50. But moving a file is not the same as copying it. In fact, Burdett expressly states that “mov[ing]” files is performed by “chang[ing] the name or the location (e.g., path in a file system) of a file.” Burdett at 25:28-31. In Burdett, the file remains in the same “file system,” so a POSITA would understand that “moving” the file can be efficiently implemented by only modifying the file’s directory metadata and without performing any “copying” of the file data. In fact, Burdett’s disclosure of “chang[ing] ... the location (e.g., path in a file system) of a file” makes it clear to a POSITA that Burdett is “moving” files by modifying file directory metadata rather than actually “copying” any files.

51. In my opinion, Petitioner does not provide any analysis or explanation of how or why Burdett’s teaching of “mov[ing]” files or any part of that process discloses or renders obvious the “copying” claim limitation. In fact, I find it telling that the petition does not even acknowledge (much less discuss) that there is a difference between “moving” a file and “copying” a file. Petitioner also does not present any reasons to believe a POSITA would have found it obvious that Burdett’s disclosure of “moving” a file includes or renders obvious “copying” a file. Petitioner, for example, does not identify any motivation to combine or provide any

analysis of reasonable expectation of success. Simply put, the petition does not present any obviousness analysis to support modifying Burdett to arrive at the “copying” limitation.

B. Argument 2: Non-Obviousness of “Natively Mount[ing]” a “Public Cloud File Store” on a “Network Security Device”

52. Petitioner states “Burdett did not explicitly describe mounting the public file store on a network security device, but FortiSandbox did.” Pet. at 28; *see also* Franz ¶ 85 (“Burdett did not explicitly describe natively mounting the public file store, but FortiSandbox disclosed this operation.”). I disagree that either reference describes this limitation.

53. I first note that a “public cloud file store” is distinguishable from other types of “cloud file store[s].” For example, a “public cloud file store” is different from a “private cloud file store.” Petitioner likewise contends that the term “public cloud file store” has a specific meaning, arguing that “a ‘public cloud file store’ generally refers to a file store maintained on shared infrastructure, as opposed to an enterprise’s private network.” Pet. at 24.

54. Neither Burdett nor FortiSandbox disclose “natively mount[ing]” a “public cloud file store” on a “network security device.” For Burdett, this is not disputed. For FortiSandbox, Petitioner merely asserts that “FortiSandbox disclosed this operation” (Franz ¶ 85), but presents no evidence or analysis that FortiSandbox’s “network share” corresponds to a “*public* cloud file store.”

FortiSandbox describes “five methods for importing files” that are then scanned for known threats, including “network share.” FortiSandbox at 88. “Network share allows you to scan files located on a remote file share as scheduled, and quarantine bad files.” *Id.* To use this feature, the user specifies a “Server Name/IP,” a “Share Path,” and a “Mount Type.” *Id.* at 115. But there is no analysis in the petition explaining how FortiSandbox’s “remote file share” corresponds to a “public cloud file store.”

55. The Petitioner’s only analysis is that “FortiSandbox’s use of mount types including CIFS and NFS protocols” mirrors the ’856 patent’s use of “NFS” and “CIFS.” Pet. at 29 (quoting ’856 patent at 11:7-10). However, even if the ’856 patent were to disclose that use of “NFS” and “CIFS” was part of the technology used to “natively mount[]” a “public cloud file store,” those protocols alone are not sufficient to show “natively mount[ing]” a “public cloud file store.” For example, the ’856 patent identifies specific software “for mounting an Amazon Web Services (AWS) Simple Storage Service (S3) bucket” as opposed to relying solely on generic protocols such as NFS and CIFS. ’856 patent at 5:28-32.

56. Fortinet’s product development efforts confirm that the Administrator Guide for the version of FortiSandbox cited by Petitioner does not disclose “natively mount[ing]” a “public cloud file store,” as claimed. Petitioner’s FortiSandbox reference corresponds to FortiSandbox version 2.3.2 released (according to

Petitioner) in 2016. FortiSandbox at 1; Pet. at 18 n.5. The only available “mount type[s]” in FortiSandbox version 2.3.2 were “CIFS” and “NFSv2” to “v4”:

Mount Type	Select the mount type of the network share folder. The following options are available: • CIFS • NFSv2 • NFSv3 • NFSv4
-------------------	---

57. By FortiSandbox version 4.2.4, released in 2023—long after the ’856 patent’s priority date—FortiSandbox included “mount type[s]” expressly corresponding to “public cloud file store[s],” such as Amazon Web Services (“AWS”) and Microsoft Azure:

Mount Type	Select the mount type. The following options are available: • CIFS (SMB v1.0, v2.0, v2.1, v3.0). • NFSv2, NFSv3, NFSv4. • AWS S3, AWS S3 BJ, AWS S3 NX. See AWS S3 Settings on page 100. • Azure File Share. See Azure File System on page 101. • Azure Blob Storage. See Azure Blob Storage on page 101. For domain-based DFS namespace, ensure the domain name can be resolved with the system Primary DNS server.
-------------------	--

Ex. 2003 at 74.

58. The addition of express support for “public cloud file store[s]” as “mount type[s]” in FortiSandbox version 4.2.4 shows that the FortiSandbox Administrator Guide for version 2.3.2 (without more) does not itself disclose the claim limitation of “natively mount[ing]” a “public cloud file store,” as claimed.

59. Also, in my opinion, Petitioner has not shown that it would be obvious to apply FortiSandbox's alleged teaching of "natively mounting" to Burdett's alleged teaching of a "public cloud file store."

60. **First**, Petitioner has not shown a reasonable expectation of success for any such hypothetical combination. Permitting the native mounting of a public cloud file store introduces privacy and compatibility complications that are not applicable to a private cloud file store. With a public cloud file store, you need to enable native mounting across systems controlled by two different entities, while at the same time ensuring access to an entity's files is not impermissibly granted to third parties. By contrast, in a private cloud file store, all aspects of the system are controlled by the same entity, which simplifies compatibility challenges and does not present the same privacy challenges. Petitioner never addresses these complications.

61. **Second**, Petitioner has not identified a sufficient motivation for applying FortiSandbox's alleged "native[] mount[ing]" to Burdett's alleged "public cloud file store." Petitioner argues combining Burdett and FortiSandbox "would have enabled the network security device to access files in the cloud file store and to natively communicate with the cloud file store to perform actions such as controlling file access" and "would have enabled the network security device to perform scheduled or on-demand bulk scans within the cloud file store." Pet. at 30.

But Burdett's scanning service already disclosed "controlling file access" **without** natively mounting its alleged "public cloud file store," and the ability of Burdett's scanning service to "perform scheduled or on-demand bulk scans" is already providing by giving it "permissions to enable it to access files" independent of any "natively mount[ing]." Burdett at 25:27-28 ("[T]he scanning service may move files or change the name of files in order to control access."); Burdett at 27:38-40. Petitioner does not explain how FortiSandbox's alleged "native mounting" would in any way improve on Burdett's disclosures.

62. Separately, Petitioner argues that "[c]onfiguring a device like the FortiSandbox system to perform Burdett's malware scans would have enabled an enterprise to preserve the privacy of its files in cloud storage while preventing enterprise users from accessing malicious files." Pet. at 19. To the extent this is even a benefit to Burdett's system, it goes to the scanning capabilities of FortiSandbox and does not provide any motivation for adding FortiSandbox's alleged "native[] mount[ing]" to Burdett.

63. **Third**, adding support for mounting "public cloud file shares" like AWS or Microsoft Azure was not a trivial development. It took seven years and numerous version to go from FortiSandbox version 2.4.2 cited by Petitioner, which supported NFS and CIFS protocols, to FortiSandbox version 4.2.4, which actually introduced express "mount type[s]" for a "public cloud file share." Compare

FortiSandbox at 83 *with* Ex. 2003 at 74. If it were obvious to modify FortiSandbox to permit for the native mounting of a “public cloud file share,” Patent Owner (and others) would have done so years earlier.

C. Argument 3: Improper Combining of Different Embodiments from Burdett

64. Independent claim 1 requires “prevent[ing] users of an enterprise network from accessing malware infected files that are stored within public cloud file store.” In my opinion, a separate reason to reject this ground is that Petitioner relies on different combining different embodiments in Burdett, which I understand requires a showing of obviousness. In my opinion, Petitioner has not made such a showing.

65. For the recitation of “enterprise network,” Petitioner cites Burdett at column 5:29-31, 7:14-16, and 14:18-21 for the “enterprise network” (Pet. at 22-23) and specifically quotes discussion of “threat management facility 100 ... provid[ing] threat protection to the enterprise facility 102, where the enterprise facility 102 may include a plurality of network components” (Burdett at 14:18-21).

66. For the recitation of “public cloud file store,” Petitioner cites Burdett at column 25:26-28, 25:55-67, 26:44-58, 29:31-36, 29:58-61 (Pet. at 23 (emphasis added)). This discussion in Burdett corresponds to Figure 3 and specifically “scanning service 304” that restricts “applications in the [virtual private cloud or VPC] 314” from accessing files in “cloud data store 306.” Burdett at 25:47-55.

67. These disclosures are for different embodiments. The discussion of “prevent[ing] users ... from accessing malware infected files” from Burdett’s description of Figure 3 does not address Burdett’s Figure 1. Figure 1 depicts a traditional (i.e., not cloud-based) “networked computer-based infrastructure” (Burdett at 6:35-40), whereas Figure 3 depicts a separate “cloud infrastructure 302” (Burdett at 23:33-34). Petitioner cites Burdett at column 9:29, 11:57-66, 14:19-24, 24:22-25, and 26:14-28. None of Petitioner’s cited portions of Burdett state that the “scanning service 304” of Figure 3 can be used to restrict access to files by the “networked components” of “enterprise facility 102” in Figure 1 or that the “networked components” of Burdett’s “enterprise facility 102” in Figure 1 can be implemented as “applications running in the VPC 314” of Figure 3.

68. Yet despite relying on different embodiments within Burdett, Petitioner presents no obviousness analysis to support combining Burdett’s Figure 1 embodiment with Burdett’s Figure 3 embodiment. For example, there is no discussion of reasonable expectation of success or motivation to combine.

69. Petitioner does not rely on FortiSandbox for either the “enterprise network” limitation or the “public cloud file store” limitation. Petitioner argues that “at least some of the operations conducted by Burdett’s scanning service would have been performed by a device, like a FortiSandbox device.” Pet. at 29-30. But Petitioner’s obviousness combination does not purport to change that this “scanning

service” restricts access by “applications in the VPC 314” to files in “cloud data store 306” (see Burdett at 25:47-55), rather than the “enterprise facility 102.” Thus, FortiSandbox is not a replacement to showing that it would be obvious to combine Burdett’s Figures 1 and 3.

70. Petitioner also combines Burdett’s Figures 1 and 3 without performing the necessary obviousness analysis in its invalidity analysis for independent claims 13 and 21, and those arguments fare no better. Claim 13 requires “protecting an enterprise network,” but the words “enterprise network” do not even appear in Petitioner’s discussion of any claim limitation, which predominantly cites back to Petitioner’s analysis of claim 1. Pet. at 35-37. Claim 21 similarly requires that “users of an enterprise network [be] protected by the network security device,” but Petitioner’s discussion for this limitation merely cites back to Petitioner’s analysis for claim 1. Pet. at 37. Petitioner’s theories for claims 13 and 21 thus are analogous to Petitioner’s theories for claim 1.

VII. GROUND 2: SHARPE, SEESE, AND KAMALAPURAM

D. Argument 1: No Obviousness Theory for “Copying”

71. All independent claims of the ’856 patent require “*copying* the clean file from the first repository to a second repository.” As explained above in connection with Ground 1, the claimed “copying” of files is different from “moving” files.

72. Petitioner acknowledges that “Sharpe did not explicitly describe that the ‘clean’ file would be copied to a second repository” (Pet. at 67) and relies on Kamalapuram for this limitation. But Petitioner admits Kamalapuram only describes “moving” files:

In Kamalapuram, a “file may be stored in an isolated area of the network storage system while waiting to be scanned, and may be ***moved*** to a general storage area only after the file has been scanned by a security analysis system of the network storage system.” (Kamalapuram, [0122].) Accordingly, Kamalapuram described ***moving*** the “clean” file from a first repository (an isolated area of the network storage system) to a second repository (a general area of the network storage system)

Pet. at 67-68 (emphasis added).

73. Again, moving a file is not the same as copying a file. In fact, the cited paragraph ([0122]) of Kamalapuram states that the file was in the same “network storage system” even prior to being scanned. Because the file remains in the same storage “system,” a POSITA would understand that “moving” the file would be implemented by modifying directory metadata about the file and without performing any “copying” of the file’s data, as that would be the most efficient process and is how all file systems operate.

74. In my opinion, Petitioner does not provide any analysis or explanation of how or why Kamalapuram’s teaching of “mov[ing]” files or any part of that process discloses or renders obvious the “copying” limitation. In fact, I find it telling that the petition does not even acknowledge (much less discuss) that there is a

difference between “moving” a file and “copying” a file. Petitioner also does not present any reasons to believe a POSITA would have found it obvious that Kamalapuram’s disclosure of “moving” a file includes “copying” a file. Petitioner, for example, also does not identify any motivation to combine or provide any analysis of reasonable expectation of success. Simply put, the petition does not present any obviousness analysis to support modifying Kamalapuram to arrive at the “copying” limitation.

E. Argument 2: Non-Obviousness of “Natively Mount[ing]” a “Public Cloud File Store” on a “Network Security Device”

75. All independent claims of the ’856 patent require that a “public cloud file store is natively mounted on a network security device that is protecting the enterprise network.” In my opinion, Petitioner has not shown that either Sharpe or Seese disclose or render obvious natively mounting a public cloud file store.

76. With respect to Sharpe, Petitioner admits that “Sharpe did not describe mounting the distributed filesystem to a network security device, and did not discuss mounting the filesystem in the context of scanning the files.” Pet. at 62. In addition, Petitioner also has not shown that Sharpe discloses natively mounting any “public cloud file store” for any purpose. Petitioner contends that Sharpe’s “cloud storage system 302” is the claimed “public cloud file store” (Pet. at 57-59), but Petitioner has not shown that the alleged “native[] mount[ing]” in Sharpe’s Figure 31 is for

“cloud storage system 302” (highlighted in green below) rather a different system component, cache 3112 in storage pool 3108 (highlighted in red below).

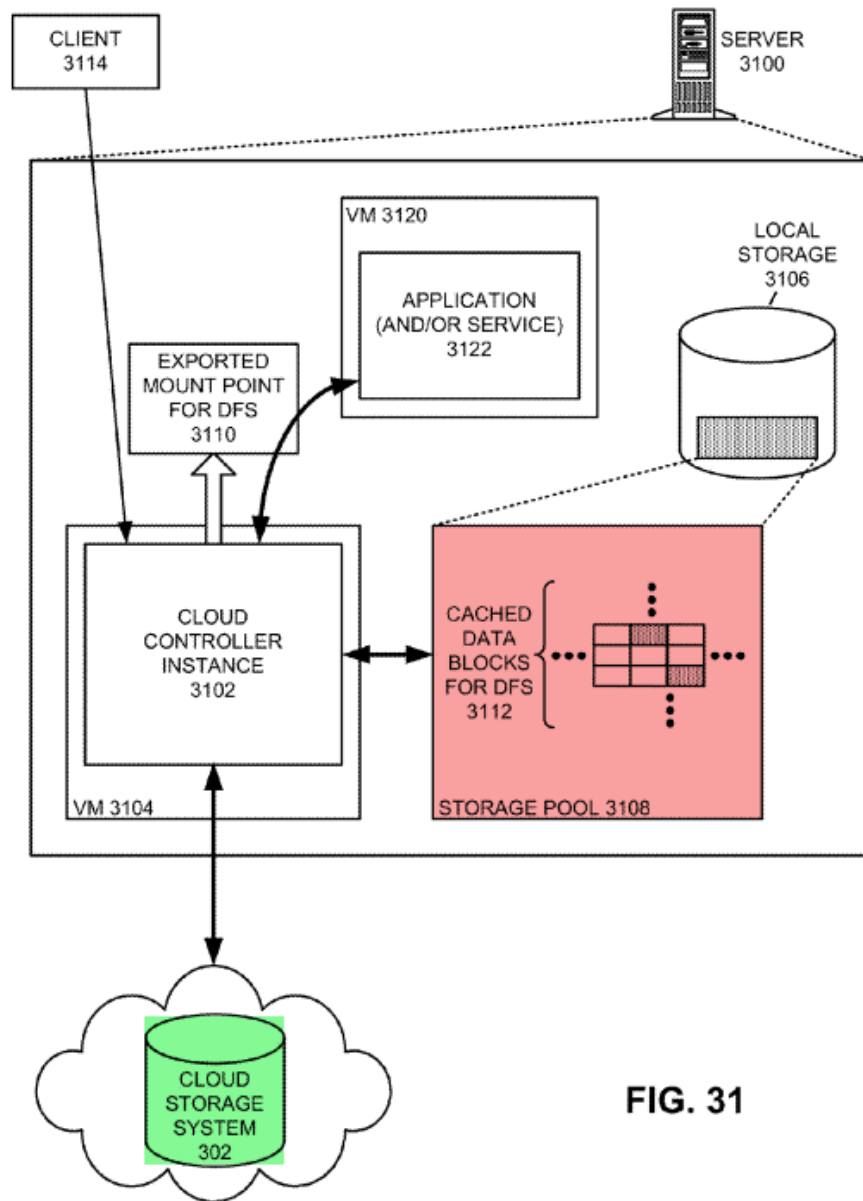


FIG. 31

77. Specifically, while Sharpe’s “cloud controller instance 3102 supports an additional interface that enables full, transparent access to the data stored in the distributed file system,” this is implemented by “cloud controller instance 3102

ensur[ing] that file blocks requested ... are loaded from cloud storage system 302 and cached 3112 in storage pool 3108.” Sharpe at 61:20-26. Thus, in Sharpe, to the extent Sharpe even discloses any “native[] mount[ing],” that “native[] mount[ing]” is for local storage pool 3108, and there is no analysis in the petition establishing that allegedly “natively mount[ing]” this local storage pool 3108 somehow qualifies as “natively mount[ing]” Sharpe’s cloud storage system 302.

78. With respect to Seese, according to Petitioner, “Seese ... describe[s] mounting a public cloud file system in the context of scanning the files to protect an enterprise against data loss” and also “mounting the public files on a network security device.” Pet. at 62-63. Petitioner provides no analysis or evidence to show, however, that Seese’s “cloud service” is a “**public** cloud file store” as opposed to (for example) a private cloud. Seese itself acknowledges in its “Background” section that “[t]here are three traditional types of cloud storage systems”: “public cloud storage,” “private cloud storage,” and “hybrid cloud storage.” Seese at [0004]-[0006]. Petitioner cites Seese’s paragraphs [0063]-[0068] as allegedly disclosing “natively mount[ing]” a “user share on the cloud service 104A” (Seese at [0068]), but does not discuss whether Seese’s “cloud service 104A” corresponds to “public cloud storage” or “private cloud storage.” Indeed, Petitioner’s analysis for the claimed “public cloud file store” never analyzes whether that limitation is met by Seese. See Pet. at 57-59.

79. But even if Petitioner's theory were interpreted as relying on a combination of both Sharpe and Seese for this limitation, in my opinion, Petitioner has not met its burden of showing that it would be obvious to apply Seese's alleged teaching of "natively mounting on a network security device" to Sharpe's alleged teaching of a "public cloud file store."

80. **First**, Petitioner contends that "Sharpe ... provided a motivation or suggestion to a POSITA for performing [the operation of mounting the distributed filesystem to a network security device], making it obvious to combine Sharpe with Seese." Pet. at 62. But as explained above, Petitioner has not shown that Sharpe's "native[] mount[ing]" even applies to a "public cloud file store," even though Petitioner contends that Sharpe discloses such a "public could file store." Sharpe cannot supply a "motivation or suggestion" to apply Seese's alleged "native[] mount[ing]" on a "network security device" to Sharpe's alleged "public cloud file store" when Sharpe allegedly already had "native[] mount[ing]" and a "network security device," yet chose not to combine those features in the claimed manner. In my opinion, Petitioner's attempt to combine the two is clearly driven by hindsight.

81. **Second**, Petitioner does not address whether the combination would have a "reasonable expectation of success," which I understand is a requirement for any obviousness analysis. The petition contains a single clause within a conclusory statement that pays lip service to "reasonable expectation of success" (Pet. at 52) but

presents no analysis on that subject. Petitioner's expert repeats that same sentence verbatim and adds no analysis. Franz ¶ 138.

82. **Third**, Petitioner does not offer a sufficient reason for why a POSITA would be motivated to combine Sharpe and Seese in the proposed manner. Petitioner argues that Sharpe and Seese are directed to similar prior art fields (malware detection and policy compliance) (Pet. at 53), but I understand that at most shows they are analogous art and does not supply a motivation to combine them. Petitioner also argues that "Configuring Seese's server to perform Sharpe's malware scans would have enabled an enterprise to preserve privacy of the enterprise's files stored within a cloud storage system, one of Sharpe's expressly-stated goals" (Pet. at 53), but Petitioner never explains why "natively mount[ing]" a "public cloud file share" on a "network security device" furthers those alleged "goals." Finally Petitioner argues that "mount[ing] a public cloud file store, as taught by Seese and suggested by Sharpe, on a network security device as taught by Seese" would have "enable[d] a network security device to access files in the cloud file store and to natively communicate with the cloud file store to perform actions such as controlling access to the files." Pet. at 64. But Sharpe's "cloud controllers" were already capable of "acces[ing] files in the cloud file store" and "controlling access to the files" without incorporating the teachings of Seese. Sharpe at 97:39-62 ("[C]loud controller 4208 may be configured ... to immediately access cloud storage system 302" and, "if the

file was found to be infected,” to “quarantine the file”). Petitioner does not provide any explanation for why Seese’s teachings would improve on Sharpe’s ability to access files in the cloud. Thus, I do not think this is a sufficient motivation to combine the two references.

VIII. GROUND 3

83. Ground 3 of the petition only challenges dependent claims. Thus, Ground 3 relies on the analysis of Ground 2. In my opinion, Ground 3 fails for the same reasons I discussed above with respect to Ground 2.

IX. CONCLUSION

84. In signing this declaration, I recognize that the declaration will be filed as evidence in a contested case before the Patent Trial and Appeal Board of the United States Patent and Trademark Office. I also recognize that I may be subject to cross-examination in the case and that cross-examination will take place within the United States. If cross-examination is required of me, I will appear for cross-examination within the United States during the time allotted for cross-examination.

85. I hereby declare that all statements made herein of my own knowledge are true and that all statements made on the 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 of Title 18 of the United States Code.

Executed: June 20, 2023

A handwritten signature in black ink, appearing to read 'J. Black Jr.', written over a horizontal line.

John Black Jr., Ph.D.