

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

NETSKOPE, INC.,
Petitioner,

v.

FORTINET, INC.,
Patent Owner.

Case No. IPR2023-00458
U.S. Patent No. 9,280,678

PATENT OWNER'S PRELIMINARY RESPONSE

TABLE OF CONTENTS

	<u>Page</u>
I. INTRODUCTION	1
II. OVERVIEW OF THE '678 PATENT	3
III. OVERVIEW OF CERTAIN CITED ART	6
A. Ground 1 Cited Art.....	7
1. Cidon (Ex. 1004).....	7
2. Shikfa (Ex. 1006).....	9
3. Herrmann (Ex. 1005)	10
B. Ground 4 Cited Art.....	11
1. Auradkar (Ex. 1009)	11
2. Chiueh (Ex. 1011).....	13
3. Chambers (Ex. 1010)	14
IV. LEVEL OF ORDINARY SKILL IN THE ART.....	15
V. CLAIM CONSTRUCTION	16
VI. GROUND 1: PETITIONER FAILS TO ESTABLISH THAT CIDON, SHIKFA, AND HERRMANN RENDER CLAIMS 1- 3, 8-14, AND 16-27 OBVIOUS	17
A. Petitioner Fails to Show the References Render Obvious the “Policy for Managing Access to and Processing a File”	17
B. Petitioner Fails to Show the References Render Obvious “Encrypting ... Content of the File to Produce a Searchable Encrypted File”	21
C. Petitioner Fails to Establish or Sufficiently Explain the Alleged Combination of Cidon With Shikfa	23

D.	Petitioner Fails to Establish or Sufficiently Explain the Alleged Combination of Cidon/Shikfa With Herrmann 28
E.	Petitioner Fails to Show the References Render Obvious “A Policy Selected From a Group of Policies” 33
VII.	GROUND 2 AND 3 FAIL ON THE MERITS 34
VIII.	GROUND 4: PETITIONER FAILS TO ESTABLISH THAT AURADKAR, CHIUUEH, AND CHAMBERS RENDER CLAIMS 1-3, 8-19, AND 21-27 OBVIOUS 34
A.	Petitioner Fails to Show the References Render Obvious the “Policy for Managing Access to and Processing a File” 34
B.	Petitioner Fails to Establish or Sufficiently Explain the Alleged Combination of Auradkar With Chiueh 37
C.	Petitioner Fails to Establish or Sufficiently Explain the Alleged Combination of Auradkar/Chiueh With Chambers 42
IX.	GROUND 5 FAILS ON THE MERITS 48
X.	CONCLUSION 49

EXHIBIT LIST

No.	Exhibit
2001	Declaration of John Black, Jr., Ph.D.
2002	<i>Curriculum vitae</i> of John Black, Jr., Ph.D.

I. INTRODUCTION

The Petition fails to meet its burden to establish unpatentability for at least several different reasons. *First*, neither of the primary references—Cidon (Ground 1) and Auradkar (Ground 4)—disclose the claimed “policy for managing access to and processing a file,” which must both “define[] access rights of user[s]” and define “cryptographic key information” used to produce a “searchable encrypted file,” along with other recited requirements. While the claims clearly recite that these are all requirements of the same single “policy,” Petitioner completely ignores this claim requirement, erroneously conflating multiple disparate policies in the references as if they were one and the same. Critically, Petitioner never shows (or even alleges) that a POSITA would have understood the disparate policies in the references to be part of the same single “policy for managing access to and processing a file,” as the claims plainly require. And Petitioner also never shows (or even alleges) that it would have been obvious to a POSITA to combine the disparate policies in the references into the single recited policy. Petitioner’s wholesale failure to address this important claim requirement mandates denial of the Petition.

Second, as Petitioner admits, Cidon and Auradkar both also lack the claimed process for producing a “searchable encrypted file.” To address this important deficiency, Petitioner attempts to augment the primary references

with encryption techniques taught in two other references, Shikfa (Ground 1) and Chieuh (Ground 4). But Petitioner ignores that, for data security and privacy reasons, Shikfa and Chieuh both teach that their encryption techniques should be performed at the client device, and should *not* be performed at a gateway system as in the claimed inventions or at server systems like those in Cidon and Auradkar. For this reason, and others discussed below, Petitioner has not shown (and cannot show) a motivation to combine these references with Cidon and Auradkar to achieve the claimed searchable encryption process, or a reasonable expectation of success in making such combinations.

Third, as Petitioner admits, Cidon and Auradkar both also lack the claimed “gateway.” Hence, Petitioner purports to implement the disclosures of these primary references on the “gateway[s]” discussed in still two more references, Herrmann (Ground 1) and Chambers (Ground 4). But Herrmann and Chambers do not address cloud storage, let alone implementing file storage policies or searchable encryption techniques. In short, they are both non-analogous references in completely different fields of art. Moreover, for the reasons discussed below, Petitioner fails to show (and cannot show) a motivation to combine Herrmann and Chambers with Cidon/Shikfa and Auradkar/Chieuh to implement the claimed processes on the required “gateway,” or a reasonable expectation of success.

For all these reasons, as well as others discussed below, Petitioner fails to meet its burden and the Board should deny institution of *inter partes* review.

II. OVERVIEW OF THE '678 PATENT

The '678 patent provides a new vendor independent “secure cloud storage” solution, introducing novel systems and methods for “implementing a policy based framework for encrypting, storing, accessing, querying and managing data across one or more cloud platforms” (e.g., Dropbox, Google Drive, Amazon Web Services, etc.). Ex. 1001, 3:23-28, 6:9-14; Ex. 2001, ¶¶29-30. Referring to the example in Figure 2 below, annotated by Dr. Black, the system operates as an intermediary “Gateway” [204] that facilitates “interactions between enterprise users [202a, 202b, 202c] and cloud service providers [206a, 206b, 206c] for storage, retrieval, and searching of data and content stored by the providers.” Ex. 1001, 8:27-9:4; Ex. 2001, ¶¶31-32.

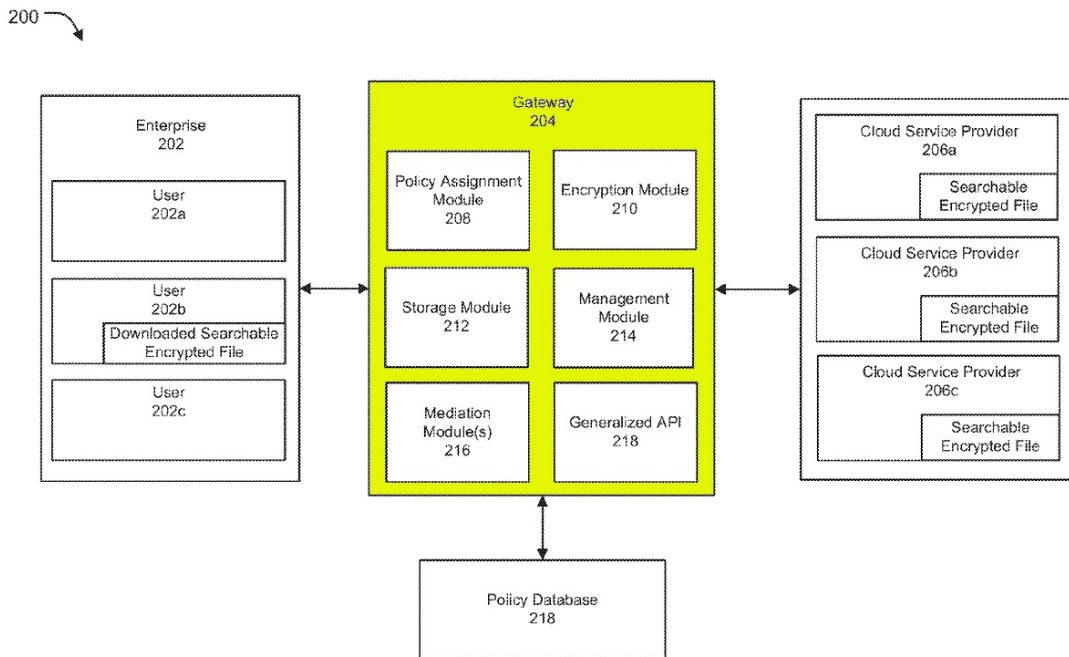


Fig. 2

Unlike prior art cloud storage systems that suffered from “rigid and non-flexible” interactions between enterprises and individual cloud service providers, the ’678 patent’s intermediary Gateway enables enterprises to more easily “store data across multiple cloud platforms to ensure more security, redundancy, and reduction in potential threat of data compromise,” and to implement global file storage policies that “allow storage of data across different service providers, allow different cloud storage access rights across users and computing devices, [and] allow searching of downloaded encrypted data across cloud service vendors.” Ex. 1001, 1:38-2:30. To this end, the Gateway may assign users a policy that “not only defines the manner in which the users can access and process content stored on the cloud, but can also

configure the mode in which the data is encrypted, stored, searched, and accessed to ensure secure and vendor independent cloud management.” *Id.*, 3:28-67; Ex. 2001, ¶33.

Because storing files on cloud platforms in an “unencrypted format” may expose them to “network attack[s] that may lead to the [files] being compromised” (Ex. 1001, 1:52-54), the Gateway can also “encrypt one or more files to be uploaded/stored” using “cryptographic key information” defined by the assigned policy. *Id.*, 4:1-12. But recognizing that conventional encryption techniques also “make[] the files ... unsearchable and hence unfriendly for user access” (*id.*, 1:54-57), the Gateway can produce a “searchable encrypted file” via a process that involves “dividing the file into [] chunks; creating namespaces for one or more of the [] chunks; and configuring the namespaces of the one or more chunks such that content of the file is encrypted in a manner that makes it searchable.” *Id.*, 19:62-20:3; Ex. 2001, ¶34.

Moreover, as Dr. Black explains, implementing this searchable encryption process at the intermediary Gateway provides important advantages over prior art systems. Ex. 2001, ¶35. For example, some prior art systems relied on the individual client device to encrypt files before storing them on the cloud platform, thereby inhibiting the enterprise from

implementing policies that allow other authorized client devices and users in the enterprise to search for and access those encrypted files. In contrast, the '678 patent's Gateway readily provides these capabilities. *See, e.g.*, Ex. 1001, 17:21-18:37. Other prior art systems relied on the cloud service provider to implement file encryption, thereby risking that the files could be “exposed to attack within the cloud, insider jobs and subpoena—all without the knowledge or consent of the data owner.” *Id.*, 1:59-62. In contrast, the '678 patent's Gateway can implement the patent's searchable encryption process “independent of the vendor/service provider of the cloud platforms.” *Id.*, 10:1-6; Ex. 2001, ¶35.

As Dr. Black explains, the combination of features in the '678 patent provides a more secure, flexible, and user-friendly cloud storage solution than what existed before. Ex. 2001, ¶¶36-40.

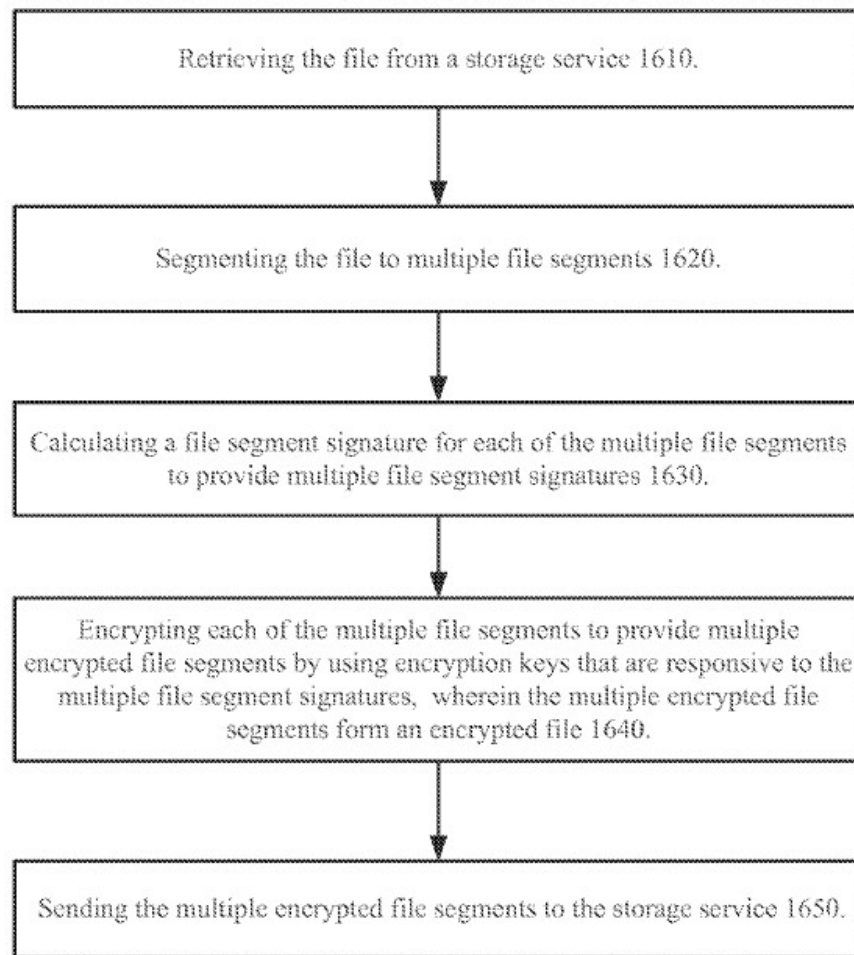
III. OVERVIEW OF CERTAIN CITED ART

The petition challenges the '678 patent's independent claims 1 and 16 on two grounds. Ground 1 asserts that the combination of Cidon, Shikfa, and Herrmann renders the independent claims obvious. Ground 4 asserts that the combination of Auradkar, Chiueh, and Chambers also renders the independent claims obvious. Below are short summaries of these six references.

A. Ground 1 Cited Art

1. Cidon (Ex. 1004)

Cidon is directed to systems and methods for encrypting files stored on cloud-based storage services. Cidon, ¶10. Cidon's basic method for encrypting a file is depicted in Figure 15, below. *See also id.*, Abstract, ¶10. Importantly, as Dr. Black explains, Cidon does ***not*** disclose that any of its file encryption methods encrypt the content of a file in a manner that makes it searchable. Ex. 2001, ¶45.



1600

FIG. 15

Cidon's file encryption methods are implemented by a "management server," a "complete software system which is responsible for encrypting files in the cloud and enforcing secure access to these cloud files by providing the right decryption keys to the right users." Cidon, ¶102, Fig. 2. To this end, the management server includes an "IT Policy Enforcement Module" that

enforces user “access policies” controlling user access to secure files. *Id.*, ¶108. However, as Dr. Black explains, Cidon does **not** disclose that these user “access policies” ever define cryptographic key information or otherwise control the encryption and storage of files on cloud storage systems. Ex. 2001, ¶46.

2. Shikfa (Ex. 1006)

Shikfa is directed to a “system and method for enabling searchable encryption of encrypted documents stored by a client on one or more storage providers.” Shikfa, Abstract. Shikfa’s system includes a “broker server” that “transfer[s] the encrypted documents between the client and the one or more storage providers[,] maintain[s] information indicating where the encrypted documents are transferred[,] [and] stores information for at least one encrypted index for the encrypted documents.” *Id.* However, Shikfa teaches that it is not this broker server, but the **client** that “encrypts the one or more documents” and “generate[s] [the] one or more encrypted indexes.” *Id.*, ¶29. As Shikfa explains, this “ensures that the broker server 12 cannot access and/or discover the content of the client's data because the broker server 12 does not know the encryption used on the encrypted documents.” *Id.*, ¶48. As Dr. Black explains, Shikfa teaches that this arrangement is important for

“maintaining the confidentiality of data external to the client.” Shikfa, ¶48; Ex. 2001, ¶47.

Shikfa further discloses that the client may encrypt documents and generate encrypted indexes using any “encryption mechanism known in the art.” Shikfa, ¶29. However, as Dr. Black explains, Shikfa does *not* disclose performing an encryption process that divides the documents into chunks and creates and configures namespaces for one or more of the chunks. Ex. 2001, ¶48.

3. Herrmann (Ex. 1005)

Herrmann describes a system and methods for “anti-virus cooperative enforcement.” Herrmann, Abstract, ¶¶48-50. In Herrmann’s system, “[b]efore ... a client computer[] is permitted to join a network or access protected resources, a check is made to evaluate the antivirus engine and virus definitions that are installed and in use on [the client computer].” *Id.*, ¶50; Ex. 2001, ¶49.

As an example, “when a client computer requests a VPN connection to a gateway server, the system ... determines whether or not a client computer is using an approved anti-virus application.” Herrmann, ¶50. The system makes this determination via a series of interactions between the client computer and a separate “integrity server.” *Id.*, ¶29; *see also id.*, Figs. 3-4.

Then, “[a]fter the integrity server 370 has evaluated compliance with anti-virus policies or rules, it notifies the gateway server about whether to permit or restrict access by this particular client device,” and “[t]he gateway server may then decide how much access to permit by this particular client device (i.e., the gateway server 350 assigns an access policy to the connection/session between the gateway server 350 and the client computer 310) based upon the notification.” *Id.*, ¶77; Ex. 2001, ¶50.

However, as Dr. Black explains, Herrmann does ***not*** disclose using its anti-virus cooperative enforcement system in connection with encrypting, storing, or controlling access to files stored on the cloud. It does not reference cloud storage at all. Ex. 2001, ¶51.

B. Ground 4 Cited Art

1. Auradkar (Ex. 1009)

Auradkar is directed to a comprehensive “trustworthy platform” in which “data services, including mathematical transformation techniques, such as searchable encryption, deassembling/reassembling or distribution techniques, for data, are provided in a way that distributes trust across multiple entities ... and decouples data protection requirements ... from the containers that hold the data (e.g., file systems, databases, etc.) enabling the data to act as its own custodian.” Auradkar, ¶¶9-10. Auradkar explains that Figure 1,

below, depicts “a general environment for providing one or more embodiments of secure, private and selectively accessible network data services as described herein.” *Id.*, ¶110; Ex. 2001, ¶52.

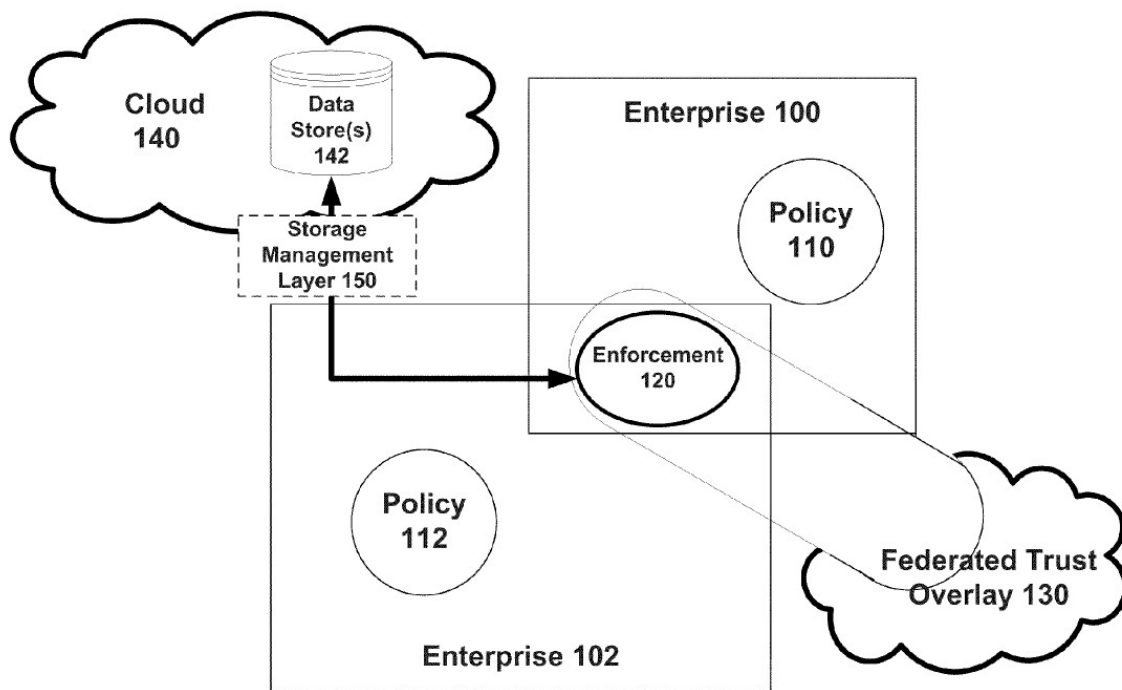


FIG. 1

Auradkar further explains that Figure 5, below, illustrates “aspects of data acting as its own custodian,” wherein “an enterprise 520 has enterprise staff 522 that can be subject to enterprise access policy 530 ... [that] can affect how data 512 stored in a data container 510 of a cloud container 500 can be accessed, manipulated, retrieved, searched, etc.” Auradkar, ¶¶116-117. However, as Dr. Black explains, Auradkar does *not* disclose that these “access polic[ies]” governing “users seek[ing] access to data” ever define

cryptographic key information or otherwise control the encryption and storage of this data on cloud storage systems. Ex. 2001, ¶53.

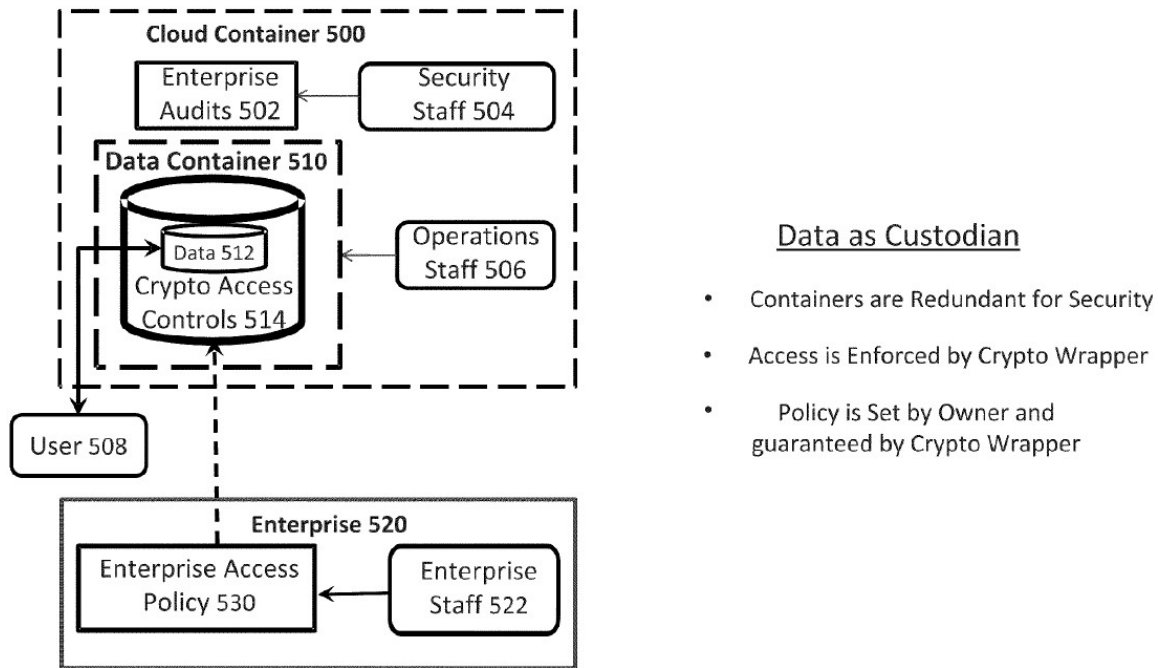


FIG. 5

2. Chiueh (Ex. 1011)

Chiueh is directed to a “document processing method and ... system” intended to “improve information security of remote storage documents, and facilitate a remote server to deal with various data processing requirements,” including keyword searching. Chiueh, ¶7. The system includes a “client [] connected to [a] remote server through a communication network.” *Id.*, ¶9, Fig. 1. Before storing a document, the client in Chiueh performs a series of steps to create “encrypted pages” and a “significant word set (SWS),” which

“the client transmits ... to [the] remote server for storage.” *Id.*, ¶¶8-9, Fig. 2; Ex. 2001, ¶54.

As Dr. Black explains, Chiueh (like Shikfa) teaches that its document processing method is performed by the *client*, and that Chiueh’s method should *not* be performed by remote servers. Ex. 2001, ¶55. Consequently, in Chiueh’s system, “the remote server does not have the keys [and it can] not decrypt the document pages and the significant word set.” Chiueh, ¶10. Chiueh explains that this is important because “the remote server is probably not reliable,” and if “the remote server ha[d] the decryption capability, the client could not prevent an unfaithful query processing of the remote server,” and “the document stored in the remote server could be peeked/leaked.” *Id.*, ¶5; Ex. 2001, ¶55.

3. Chambers (Ex. 1010)

Chambers is directed to “[t]echniques for managing access control policies” that control access to computer networks (Chambers, Abstract) and, more specifically, it describes “[t]echniques for automatically assigning access control policies (ACPs) based on device types of network client devices (NCDs).” *Id.*, ¶21. “When a network device first connects to the network, it is examined by the system, identified by type, and assigned an access policy according to the specified rules for that device type.” *Id.* The

assigned network access policies specify network parameters such as “bandwidth limits and traffic shaping rules, VLAN assignment, firewall rules, whether a captive portal should be applied to that device, etc.” *Id.*, ¶¶32-40; Ex. 2001, ¶56.

The network access policies are assigned and enforced by “network access devices (NADs) ... managed by a management server.” Chambers, ¶24, Fig. 1. The NADs “may operate as a gateway device, an access point (AP), a network switch, or a combination thereof.” *Id.* The network access policies are “downloaded from management server 101 to [the] network access device,” and then “[w]hen a client device ... is associated with [the] network access device 102, [the NAD’s] access control module 116 controls [network] access of [the] client device 108 by enforcing [the] ACPs.” *Id.*, ¶28, Fig. 6; Ex. 2001, ¶57.

As Dr. Black explains, Chambers (like Herrmann) does *not* disclose using its network access control policy management techniques in connection with encrypting, storing, or controlling access to files on cloud storage systems. Chambers does not reference cloud storage at all. Ex. 2001, ¶58.

IV. LEVEL OF ORDINARY SKILL IN THE ART

A person of ordinary skill in the art (“POSITA”) relevant to the ’678 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 electrical engineering and/or computer science and one year of work or research experience in related fields. Ex. 2001, ¶18.

Patent Owner's description of the level of ordinary skill in the art is almost the same as that of the Petitioner. *See* Pet. 15. The positions set forth in this preliminary response would be the same under either parties' proposal.

V. CLAIM CONSTRUCTION

In an *inter partes* review, claim terms receive their plain and ordinary meaning in light of the specification of the patent in which it appears, the same standard as used in district court. 37 C.F.R. § 42.100(b); *see Phillips v. AWH Corp.*, 415 F.3d 1303 (Fed. Cir. 2005). Petitioner states that “no constructions are necessary.” Pet. 15. Patent Owner agrees that no claim terms presently require construction to resolve any current dispute between the parties. Patent Owner reserves the right to proffer a construction for a claim term at a later point if it becomes pertinent to any particular dispute.

VI. GROUND 1: PETITIONER FAILS TO ESTABLISH THAT CIDON, SHIKFA, AND HERRMANN RENDER CLAIMS 1-3, 8-14, AND 16-27 OBVIOUS

A. Petitioner Fails to Show the References Render Obvious the “Policy for Managing Access to and Processing a File”

Independent claim 1 requires “a policy for managing access to and processing a file to be stored on one or more cloud platforms.” Ex. 1001, claim 1[a][i]. Independent claim 16 similarly requires a “policy [that] defines and manages access to and processing of a file to be uploaded on a cloud.”

Id., claim 16[c][i][A]. Both claims further require:

- (1) that “the policy defines access rights of the [one or more] user[s]” (*id.*, claims 1[a][ii], 16[c][i][B]);
- (2) encrypting content of the file “using cryptographic key information defined by the policy” (*id.*, claims 1[b], 16[c][ii][A]);
- (3) storing “the searchable encrypted file on the one or more cloud platforms based on the policy” (*id.*, claims 1[c], 16[c][iii]); and
- (4) managing “access to the searchable encrypted file by the [one or more] user[s] based on the policy” (*id.*, claims 1[d], 16[c][iv]).

As Dr. Black explains, Petitioner fails to show that the asserted references disclose or render obvious a single “policy” that meets all of these claim requirements. Ex. 2001, ¶¶61-62. To the contrary, in addressing these different claim requirements, Petitioner relies on several disparate policies in Cidon, erroneously treating them as if they were one and the same. For example, in alleging that Cidon discloses a policy that “defines access rights

of the one or more users,” Petitioner relies solely on Cidon’s disclosure of user “access policies” that control user access to secure files. *See* Pet. 29 (discussing Cidon, ¶¶102, 108, 112, 115), Pet. 53 (referring back). But then, in alleging that Cidon also discloses “storing the searchable encrypted file ... based on the policy,” Petitioner relies solely on Cidon’s separate disclosure of file “placement policies” that control placement of files based on their relationship with other files. *See* Pet. 48-49 (discussing Cidon, ¶¶208-209, 214-215), Pet. 55 (referring back). And this pattern of conflation of Cidon’s policies is pervasive. *See* Pet. 25, 29, 33-34, 48-49, 53-36; Ex. 2001, ¶62.

As Dr. Black explains, a POSITA would have understood that Cidon’s user “access policies” and file “placement policies” are completely different. Ex. 2001, ¶63. For example, user “access policies” are “define[d]” for “users” and “groups” of “[u]sers in a company” (Cidon, ¶108), whereas file “placement policies” are instead “defined on documents” or other files. *Id.*, ¶¶206-207. In addition, while user “access policies” are used to control user “access to [] secure files” in “file repositories” (*id.*, ¶108), file “placement policies” are used to control storage operations on files, such as “[p]lacement [of files] in specific directories or shares.” *Id.*, ¶¶206-209. And while user “access policies” are defined by “administrators” in order to control users’ access to secure files (*id.*, ¶108), in contrast, the “users [can] change the [file

placement] policy if needed.” *Id.*, ¶¶213-217. Moreover, Cidon discusses user “access policies” and file “placement policies” in totally different areas of the reference, and it never discloses that they are related in any way.

Despite all these clear differences, Petitioner never argues (or shows, or even alleges) that a POSITA would have considered Cidon’s disparate user “access policies” and file “placement policies” to be part of the same single “policy” recited by the claims. As discussed above, a POSITA would have considered them to be completely different policies. Ex. 2001, ¶64.

Moreover, importantly, Petitioner does not argue that it would have been obvious to combine Cidon’s disparate policies into the single claimed “policy.” For example, never shows (or even alleges) that it would have been obvious to a POSITA to combine Cidon’s user “access policies” and file “placement policies” to achieve the single claimed “policy.” Ex. 2001, ¶65. Hence, no such combination can be made. *See, e.g., In re Kotzab*, 217 F.3d 1365, 1370 (Fed. Cir. 2000) (“Even when obviousness is based on a single prior art reference, there must be a showing of a suggestion or motivation to modify the teachings of that reference.”); *Cisco Sys., Inc. v. Bockstar Tech. LLC*, IPR2014-01192 (P.T.A.B. Feb. 2, 2015), Paper 10 (denying institution on ground based on single reference obviousness because “[a]lthough both

routers are described in a single reference, Petitioner must still articulate a rationale for combining the features of the two devices”).

Notably, for claim 1 only, Petitioner also relies on Herrmann for its purported disclosure of claim 1’s separate requirement of a “gateway” that “assign[s]” policies. *See* Pet., 25-29. But Petitioner does not rely on Herrmann to address the fact that the claims require a single “policy.” For example, Petitioner does not contend that Herrmann’s “access policy” discloses the single claimed “policy,” or that Herrmann would have motivated a POSITA to combine Cidon’s (or Herrmann’s) disparate policies to achieve the single claimed “policy.” *See id.* And as Dr. Black explains, a POSITA would have understood that Herrmann clearly doesn’t. For example, Herrmann’s “access policy” does not define “cryptographic key information” for encrypting a file, or otherwise relate in any way to “processing” and “storing” a file on the cloud. In fact, Herrmann has nothing to do with cloud file storage policies. Ex. 2001, ¶66.

At bottom, Petitioner wholly ignores that the Petition relies on multiple disparate policies to address the requirements of what the claims recite as a single “policy.” Ex. 2001, ¶67. This glaring deficiency is fatal to Ground 1.

**B. Petitioner Fails to Show the References Render Obvious
“Encrypting ... Content of the File to Produce a Searchable
Encrypted File”**

Independent claim 1 requires “encrypting ... content of the file to produce a searchable encrypted file.” Ex. 1001, claim 1[c]. Likewise, independent claim 16 requires “produc[ing] a searchable encrypted file by encrypting content of the file.” *Id.*, claim 16[c][ii][A]. For both these claim elements, Petitioner relies exclusively on Cidon, which purportedly “taught or suggested that [its] encrypted files were searchable.” Pet. 32-33, 54. Not so. Ex. 2001, ¶68.

A POSITA would have understood that encrypting content of a file to produce a “searchable encrypted file” (i.e., an encrypted file whose contents can still be searched) was highly unconventional at the time of the invention. In fact, all ordinary encryption methods at that time of the invention produced encrypted files that were *not* searchable (i.e., encrypted files whose contents could not be searched). If Cidon—a reference specifically directed to a system and methods for encrypting files—had intended to teach or suggest an encryption process that produced a “searchable encrypted file,” Cidon would have clearly said so. But as Dr. Black explains, Cidon contains no such disclosure. Ex. 2001, ¶69.

Petitioner's sole basis for asserting that Cidon "taught or suggested [its] encrypted files were searchable" is the fact that Cidon discloses functions for "[a]ccessing the complete user file and folder structure by browsing the web service account ... [and] functionalities such as document search." Pet. 32-33; Cidon, ¶¶89-91. But this portion of Cidon does not teach or suggest that the contents of Cidon's encrypted documents are searchable. Instead, it describes system management functions that enable users to find documents in the system, such as by performing a "document search" or browsing the system's "file and folder structure" (e.g., for a document with a particular filename). Cidon, ¶91. Petitioner further asserts, without any explanation, that Cidon describes extracting metadata from files that "may help in file search operation, using text/tags based search or one or more document references." Pet. 32; Cidon, ¶¶166, 171. But this portion of Cidon also does not teach or suggest that the contents of Cidon's encrypted documents are searchable. Instead, it describes the use of metadata to find and identify files in the system, such as by searching metadata text and/or tags (e.g., for a particular filename). Cidon, ¶¶165-171. As Dr. Black explains, a POSITA would have understood that the ability to search for and find files in a system is distinctly different from the ability to search the contents of the files themselves. Ex. 2001, ¶¶70-71.

Moreover, importantly, neither of the aforementioned disclosures in Cidon even refers to searching *encrypted* documents or files. *See* Cidon, ¶¶91, 166, 171. Given Cidon teaches that only some files are encrypted (*see, e.g., id.*, ¶¶224-228), and given the conspicuous absence of any reference to encrypted documents or files in the aforementioned disclosures, a POSITA would have understood that these disclosures do not refer to search operations performed on the contents of encrypted documents or files. In fact, as Dr. Black explains, there is no disclosure anywhere in Cidon that teaches or suggests its *encrypted* documents or files are searchable. Ex. 2001, ¶72.

For all these reasons, Petitioner fails to show that Cidon teaches, suggests, or renders obvious producing a “searchable encrypted file” as recited by the claims, and Ground 1 fails on this basis. Ex. 2001, ¶73.

C. Petitioner Fails to Establish or Sufficiently Explain the Alleged Combination of Cidon With Shikfa

Independent claims 1 and 16 recite a process for producing a searchable encrypted file that involves: “divid[ing] [a] file into [] chunks; creat[ing] namespaces for one or more of the [] chunks; and configur[ing] the namespaces of the one or more chunks such that content of the file is encrypted in a manner that makes it searchable.” Ex. 1001, claims 1[b][i]-[iii], 16[c][ii][B]-[D]. Petitioner admits that Cidon fails to disclose the last

part of this process. Pet. 40 (“Cidon ... did not explicitly disclose configuring the segment identifiers (namespaces) so the file content was encrypted in a searchable manner.”), Pet. 55 (referring back). In an attempt to address this important deficiency, Petitioner alleges that a POSITA “would have been motivated to combine Shikfa’s searchable encryption mechanism with Cidon’s encryption technique.” Pet. 43-44, 55. But as Dr. Black explains, Petitioner fails to show any such motivation. Ex. 2001, ¶¶74-75.

Cidon teaches that its “management server” is already a “*complete*” software system which is responsible for encrypting files in the cloud and enforcing secure access to these cloud files.” Cidon, ¶102. To this end, Cidon already describes a detailed and comprehensive set of file encryption methods implemented by the management server. *See, e.g., id.*, ¶¶234-402. As Dr. Black explains, in light of these teachings, Petitioner fails to show why a POSITA would have been motivated to modify Cidon’s “complete” system with just this one cherry-picked piece of Shikfa’s encryption method (i.e., Shikfa’s purported configuration of identifiers), rather than simply using the complete set of encryption methods that Cidon already discloses. Ex. 2001, ¶76. In fact, the only motivation is impermissible hindsight.

Petitioner alleges that the “modified system would have been benefitted from various advantages that Shikfa described,” including that “the system

would have allowed file content to be encrypted and searched ‘without requiring any such capability’ at a storage provider.” Pet. 44 (citing Shikfa, ¶50); *see also* Pet. 22 (identifying same alleged benefit). But Petitioner ignores that Cidon’s system already provides for encryption of file content without requiring any such capability at the storage provider—indeed, that is one of the primary purposes of Cidon’s system. *See* Cidon, Abstract, ¶¶97-99, 218-223. And, according to Petitioner, Cidon’s system also already provides for “searchable encryption” (as discussed above, Patent Owner disagrees). Pet. 21, 32-33. Hence, this purported benefit is something that Cidon allegedly already offers. It does not provide a motivation to combine. Ex. 2001, ¶77.

Petitioner further asserts that “the modified system would have ‘accommodate[d] document searchability’ without ‘compromis[ing the] ... privacy needs,’ thereby protecting confidential information.” Pet. 44 (citing Shikfa, ¶50). But as Dr. Black explains, this aspect of Shikfa actually *teaches against* the purported combination. Ex. 2001, ¶78. While Shikfa’s system includes an intermediary “broker server” that “transfer[s] [] encrypted documents between the client and the [] storage providers” (Shikfa, Abstract), Shikfa expressly teaches it is only the *client* that “encrypts the [] documents” (*id.*, ¶29), thereby “ensur[ing] that the broker server 12 *cannot* access and/or

discover the content of the client's data.” *Id.*, ¶48. This is in direct contrast to the ’678 patent, which, as discussed above, teaches that its searchable encryption processes are beneficially implemented at its intermediary Gateway, not at the client. Hence, a POSITA considering Shikfa would have understood that, to fulfill Shikfa’s stated privacy needs, Shikfa’s encryption techniques should **not** be implemented on an intermediary gateway system (as the challenged claims require), or on the server systems disclosed in Cidon (and in Herrmann). Ex. 2001, ¶78.

For all these reasons, Petitioner fails to show that a POSITA would have been motivated combine one cherrypicked piece of Shikfa’s encryption technique (i.e., Shikfa’s purported configuration of identifiers) with the encryption methods in Cidon’s “complete” system (and, for claim 1, with Herrmann’s gateway server) in the specific manner of the challenged claims. If anything, Shikfa teaches away from such a combination. Ex. 2001, ¶79.

Petitioner also fails to show a reasonable expectation of success. Petitioner merely asserts, without any explanation, that a POSITA would have reasonably expected success “because searchable encryption mechanisms had been well-known and would have been compatible with Cidon and Herrmann’s systems, and the combination would have yielded expected results.” Pet. 45. Petitioner’s conclusory and unsupported statement is

insufficient to establish a reasonable expectation of success. But as Dr. Black explains, it is also wrong on the merits. Searchable encryption is an extremely complex technology that was also highly unconventional at the time of the invention. Moreover, Shikfa's searchable encryption technique would have been incompatible with Cidon's system (and, for claim 1, with Herrmann's gateway server), including because Cidon's "management server" is already a "**complete** software system" (Cidon, ¶102) performing encryption methods that encrypt individual "file segments" (Cidon, Abstract, ¶10), whereas Shikfa's encryption technique is performed on entire documents (Shikfa, ¶29) and is not intended to be performed on intermediary server systems like that of Cidon (or Herrmann). In light of these differences, Petitioner fails to show how the references could be successfully combined, much less in the manner of the challenged claims. Ex. 2001, ¶80.

Hence, Petitioner fails to show a motivation to combine and reasonable expectation of success in combining Shikfa with Cidon to achieve the claim limitation requiring "configur[ing] the namespaces of the one or more chunks such that content of the file is encrypted in a manner that makes it searchable." Ex. 2001, ¶81. For this reason as well, Ground 1 fails.

D. Petitioner Fails to Establish or Sufficiently Explain the Alleged Combination of Cidon/Shikfa With Herrmann

Independent claim 1 recites that multiple method steps—including “assigning,” “encrypting,” and “storing”—are performed “by a gateway device.” Ex. 1001, claim 1. Petitioner concedes that, at a minimum, Cidon fails to disclose that the claimed “assigning” step is performed “by a gateway device.” Pet. 25 (“Cidon ... did not explicitly disclose assigning such a policy ‘*by a gateway device.*’”). In an effort to remedy this deficiency, Petitioner alleges that a POSITA “would have added Herrmann’s gateway server and its access policy assignment functionality to Cidon’s management server.” Pet. 27; *see also id.*, 21-22, 25-30, 44-46 (referring to Cidon/Shikfa/Herrmann combination in connection with multiple claim steps). But as Dr. Black explains, Petitioner does not show a motivation to combine and reasonable expectation of success. Ex. 2001, ¶82.

As an initial matter, Petitioner fails to even consistently articulate the purported combination. In some instances, Petitioner alleges the combination would have “would have added Herrmann’s gateway server ... *to Cidon’s management server.*” Pet. 27; *see also id.*, 22 (“incorporating Herrmann’s techniques to Cidon’s system”). In other instances, Petitioner alleges the same combination “would have entailed implementing Cidon’s [management

server operations] *on Herrmann’s gateway device.*” *Id.*, 21-22; *see also id.*, 29 (“Herrmann’s gateway server could have accommodated Cidon’s management server operations”). Given this inability to even articulate the same consistent combination for each part of the challenged claim, Petitioner cannot meet its burden with respect to the combination. Ex. 2001, ¶83.

Moreover, Petitioner does not even attempt to articulate a specific motivation to combine the references (or reasonable expectation of success) with respect to multiple claim steps. As an example, for the claim step requiring “storing, by the gateway device, the searchable encrypted file,” Petitioner baldly asserts that “[i]n the Cidon-Herrmann-Shikfa combination, the gateway device including the functionality of Cidon’s management server would store the encrypted files on a cloud platform based on the policy.” Pet. 46. But Petitioner does not offer any explanation or rationale for combining the references in the specific manner required by this “storing” claim step. *See also id.*, 29-30 (another example). For this reason as well, Petitioner cannot meet its burden with respect to the combination. Ex. 2001, ¶84.

Petitioner also does not (and cannot) show any legitimate motivation to combine Cidon with Herrmann. Cidon teaches that its “management server” is already a “**complete** software system which is responsible for encrypting files in the cloud and enforcing secure access to these cloud files.” Cidon,

¶102. And, relatedly, Cidon’s management server is already “connected to consumer-based file repositories” (*id.*, ¶100) and can already “communicate with user devices.” *Id.*, ¶¶101-102, Fig. 2. Hence, as Dr. Black explains, a POSITA would not have been motivated to modify Cidon’s already “complete” system to incorporate Herrmann’s gateway server, given that Cidon’s management server is already capable of performing all these functions. Ex. 2001, ¶85.

A POSITA also would not have been motivated to make the purported combination because Cidon and Herrmann are non-analogous references in different fields of art. While Cidon teaches a complete server system that encrypts and stores files in the cloud and then controls user access to those files (Cidon, ¶¶10, 102, Fig. 2), Herrmann teaches a gateway system that enforces the installation and use of approved anti-virus applications on client computers before they can access a network or other protected resources. Herrmann, ¶¶29, 50, Figs. 3-4. Herrmann does **not** disclose using its anti-virus enforcement system in connection with encrypting, storing, or controlling user access to files stored in the cloud. In fact, Herrmann does not address cloud storage at all. As Dr. Black explains, for these reasons, a POSITA would not have looked to Herrmann’s gateway to modify Cidon’s complete system focused on encrypting files in cloud storage. Ex. 2001, ¶86.

Petitioner merely asserts that a POSITA would have been motivated to combine Cidon with Herrmann “because instead of waiting for an administrator to assign a policy manually, the modified system would have allowed automatic policy assignment and implementation.” Pet. 28; *see also* Pet. 22 (identifying same alleged benefit). But Petitioner ignores that Herrmann’s gateway server only “assigns an access policy to the connection/session between the gateway server 350 and the client computer” *after* its separate “integrity server” determines the client computer is in compliance with applicable anti-virus rules. Herrmann, ¶¶60, 77. Hence, as Dr. Black explains, a POSITA would not have looked to Herrmann’s gateway server’s automatic policy assignment to improve Cidon’s system, which has no such separate “integrity server,” and which must assign its policies *before* compliance determinations are made regarding access to secure files. *See, e.g.,* Cidon, ¶¶108-109. Ex. 2001, ¶87. For this reason, as well as those discussed above, there would have been no motivation to combine Cidon (and Shikfa) with Herrmann.

Petitioner also fails to show (or even directly allege) a reasonable expectation of success. Petitioner merely asserts, without any explanation, that “[t]his combination would have been straightforward because gateway devices were well-known for network security-related functionalities” and

because Herrmann’s gateway server purportedly “could have accommodated Cidon’s management server operations and yielded predictable results.” Pet. 28-29. Petitioner’s conclusory and unsupported statements are insufficient to establish a reasonable expectation of success, and they are also wrong. As discussed above, Cidon’s “management server” was already a “*complete* software system” (Cidon, ¶102) that would have been incompatible with Herrmann’s gateway server (which requires a separate “integrity server” to function). As Dr. Black explains, Herrmann’s simple gateway server could not have accommodated Cidon’s complicated file encryption, storage, and access control operations, nor the complex searchable encryption techniques taught in Shikfa. Ex. 2001, ¶88. Petitioner fails to show how all these teachings could be successfully combined.

Hence, Petitioner fails to show a motivation to combine and reasonable expectation of success in combining Cidon and Shikfa with Herrmann to meet the “assigning,” “encrypting,” and “storing” steps of claim 1. Ex. 2001, ¶89. Petitioner fails to meet its burden for claim 1, and the challenged claims that depend from it.

E. Petitioner Fails to Show the References Render Obvious “A Policy Selected From a Group of Policies”

Independent claim 16 requires implementing “a policy selected from a group of policies for a user.” Ex. 1001, claim 16[c][i]. For this claim element, Petitioner relies solely on Cidon, asserting that “it would have been obvious to a POSITA that Cidon’s IT policy enforcement module would select a policy for the appropriate user/group.” Pet. 53. But the lone basis for this conclusory assertion is Petitioner’s false contention that “Cidon described that different users/groups were assigned different policies.” *Id.* (citing Cidon, ¶108).

As an initial matter, Cidon never describes that different users/groups are *assigned* different policies. To the contrary, Cidon discloses that an “administrator” (i.e., a person) “*define[s]* access policies” for users “through a web browser interface.” Cidon, ¶108. As Dr. Black explains, POSITA would have understood that defining a policy is distinctly different than “select[ing] [a policy] from a group of policies.” Ex. 2001, ¶91. For example, the former involves specifying a new policy, whereas the latter does not. Moreover, given Cidon expressly teaches that its policies are defined by an “administrator” (i.e., a person), a POSITA would not have found it obvious that Cidon’s IT policy enforcement module would, instead, define (or assign, or select) those policies. Ex. 2001, ¶¶90-91.

Petitioner fails to set forth any legitimate rationale for why a POSITA would have found it obvious that Cidon's IT policy enforcement module would "select[] [a policy] from a group of policies for a user," as claim 16 requires. And given Cidon's explicit teaching to the contrary, this would not have been obvious. Ex. 2001, ¶92. Thus, Petitioner fails to meet its burden for claim 16 and the challenged claims that depend from it.

VII. GROUNDS 2 AND 3 FAIL ON THE MERITS

As discussed above, Petitioner fails to establish that Cidon in view of Shikfa and Herrmann renders independent claims 1 and 16 obvious. Because Grounds 2 and 3 rely on Ground 1 for the limitations of the independent claims, Grounds 2 and 3 also fail for the same reasons. Ex. 2001, ¶93.

VIII. GROUND 4: PETITIONER FAILS TO ESTABLISH THAT AURADKAR, CHIUEH, AND CHAMBERS RENDER CLAIMS 1-3, 8-19, AND 21-27 OBVIOUS

A. Petitioner Fails to Show the References Render Obvious the "Policy for Managing Access to and Processing a File"

As discussed above, independent claims 1 and 16 recite a single "policy for managing access to and processing a file" (or similar), along with a number of other claim requirements regarding the same "policy." *See supra* VI.A.; Ex. 1001, claims 1, 16. Once again, Petitioner completely ignores that the claims recite only a single "policy," and fails to show that the references in Ground 4 disclose or render obvious a single "policy" that meets all the claim

requirements. To the contrary, as Dr. Black explains, Petitioner incorrectly relies on several disparate policies in Auradkar as if they were one and the same. Ex. 2001, ¶¶94-95. And Petitioner never shows (or even alleges) that a POSITA would have considered these disparate policies in Auradkar to be the same single “policy” required by the claims, or that it would have been obvious to a POSITA to combine Auradkar’s policies as such.

As an example, in alleging that Auradkar “disclosed that its policy defined user access rights,” Petitioner relies solely on Auradkar’s disclosure of “access policies” (e.g., “enterprise access policy 530”) that control user access to data. *See* Pet. 87-88 (discussing Auradkar, ¶¶112, 117), Pet. 116 (referring back). But then, in alleging that Auradkar also “disclosed [] storing the searchable encrypted file ... based on the policy,” Petitioner relies solely on Auradkar’s separate disclosure of data storage policies that control “how [and] where [data] could be stored.” *See* Pet. 106-107 (discussing Auradkar, ¶¶96, 130, Fig. 15), Pet. 117 (referring back). And, once again, this pattern of conflation of Auradkar’s dissimilar policies is pervasive. *See* Pet. 80-83, 87-88, 92-93, 106-109, 115-118; Ex. 2001, ¶96.

As Dr. Black explains, A POSITA would have understood that Auradkar’s “access policies” and data storage policies are wholly different. Ex. 2001, ¶97. For example, “an enterprise 620 defines its enterprise access

policy 630 applicable to enterprise staff” (Auradkar, ¶119), whereas Auradkar’s data storage policies are instead “create[d] ... [for] data.” *Id.*, ¶96. Moreover, while “access policies” govern whether different “users seek[ing] access to [protected] data ... [are] allow[ed] or den[ied] entry” (*id.*, ¶116), Auradkar’s data storage policies are used to implement data storage restrictions controlling “how data could be stored, where it could be stored, how long it could be stored, etc.” *Id.*, ¶96. Auradkar also discusses the “access policies” and data storage policies in different portions of the reference, and it never says they are related.

Despite all these clear differences, as with Ground 1, Petitioner never argues (or shows, or even alleges) that a POSITA would have considered Auradkar’s disparate “access policies” and data storage policies to be part of the same single “policy” that is clearly recited by the claims. As discussed above, a POSITA would have considered them to be completely different policies. Ex. 2001, ¶98.

Moreover, importantly, Petitioner does not argue that it would have been obvious to combine Auradkar’s disparate policies into the single claimed “policy.” For example, Petitioner never shows (or even alleges) that it would have been obvious to combine Auradkar’s “access policies” and data storage policies to achieve the single claimed “policy.” Ex. 2001, ¶99. Hence, any

such combination is unviable. *See In re Kotzab*, 217 F.3d 1365, 1370 (Fed. Cir. 2000); *Cisco Sys., Inc. v. Bockstar Tech. LLC*, IPR2014-01192 (P.T.A.B. Feb. 2, 2015), Paper 10. Notably, while Petitioner relies on Chambers for its purported disclosure of a “gateway” (claim 1) and “access control module” (claim 16), Petitioner does not rely on Chambers to address the claim requirement of a single “policy.” *See* Pet., 83-87, 115-116. And as Dr. Black explains, Chambers would not remedy this issue anyway. Ex. 2001, ¶100.

Ultimately, Petitioner once again ignores that the Petition relies on multiple disparate policies to address the requirements of what the claims recite as a single “policy.” Ex. 2001, ¶101. This is fatal to Ground 4.

B. Petitioner Fails to Establish or Sufficiently Explain the Alleged Combination of Auradkar With Chiueh

As discussed above, independent claims 1 and 16 recite a process for producing a searchable encrypted file that involves: “divid[ing] [a] file into [] chunks; creat[ing] namespaces for one or more of the [] chunks; and configur[ing] the namespaces of the one or more chunks such that content of the file is encrypted in a manner that makes it searchable.” *See supra* VI.C.; Ex. 1001, claims 1[b][i]-[iii], 16[c][ii][B]-[D]. Recognizing that Auradkar is deficient on its own, Petitioner asserts that Auradkar combined with Chiueh discloses the second and third parts of this process. *See* Pet. 94, 98, 117. In

particular, Petitioner alleges that a POSITA “would have been motivated to modify [Auradkar’s] technique by Chiueh’s mechanism that allowed searching individual document pages.” *Id.*, 79; *see also id.*, 97, 106 (more on same alleged combination). But as Dr. Black explains, Petitioner does not (and cannot) show any motivation to make this combination, or a reasonable expectation of success. Ex. 2001, ¶¶102-103.

Petitioner’s purported combination would “combine Auradkar, Chiueh, and Chambers,” “incorporating Chambers’ dedicated gateway as an interface between the client and protected data sources” (Pet. 78-79) “such that the claimed operations would be carried out by the gateway.” *Id.*, 93; *see also id.*, 85, 108, 115-116. But Chiueh teaches against such a combination. Chiueh expressly teaches it is only the *client* that “encrypts the document pages and the significant word set” (Chiueh, ¶¶8-10) so that “remote server[s] do[] not have the keys [and can] not decrypt the document pages and the significant word set. *Id.*, ¶10; *see also* ¶5. As discussed above, this is in direct contrast to the ’678 patent, which teaches that its searchable encryption processes are beneficially implemented at its intermediary Gateway, not at the client. Hence, as Dr. Black explains, a POSITA considering Chiueh would have understood that its encryption techniques should *not* be implemented on an

intermediary gateway system (as the challenged claims require), or on the remote systems disclosed in Chambers and Auradkar. Ex. 2001, ¶105.

Petitioner asserts that a POSITA would have been motivated to make the purported modification “because, in addition to Auradkar’s existing whole-document search, the modified system would have allowed retrieval of encrypted document portions stored in the cloud.” Pet. 97; *see also* Pet. 79, 106 (identifying same alleged benefit). However, Auradkar’s comprehensive “trustworthy platform” already purportedly allows for search and retrieval of encrypted document portions stored in the cloud. Petitioner alleges Auradkar already provides for “searchable encryption” of documents. And Auradkar’s search techniques allow data to be “selectively queried to obtain access to XML fragments” constituting “sliver[s] of a larger document ... dispersed across different [cloud storage] containers.” Auradkar, ¶¶91-92; *see also id.*, ¶¶93-95 (Auradkar’s platform “enable[s] users to search, extract and decrypt segments, subsets or parts of encrypted data,” such as individual “paragraphs” of an “XML document”). Hence, this purported benefit is something that Auradkar allegedly already offers. As Dr. Black explains, it would not have provided a motivation to combine. Ex. 2001, ¶106; *see also id.*, ¶104.

Petitioner also generally asserts that “a POSITA would have been motivated to combine Auradkar, Chiueh, and Chambers” because “[a]ll three

references concern improving computer networking and security for accessing files stored remotely from client devices” Pet. 78-79. But as Dr. Black explains, this is true of virtually any client-server system in existence, and it does not provide a motivation to combine these specific references in the specific manner of the challenged claims. Ex. 2001, ¶104.

Overall, Petitioner fails to articulate any legitimate reason why a POSITA would have been motivated to modify the searchable encryption techniques already included in Auradkar’s comprehensive platform with Petitioner’s cherry-picked pieces of Chiueh’s encryption technique in the specific manner of the challenged claims. Ex. 2001, ¶107. In fact, the only motivation for Petitioner’s purported combination is impermissible hindsight.

Petitioner also fails to show a reasonable expectation of success. Petitioner asserts that a POSITA would have reasonably expected success “because Auradkar already disclosed generating tags, which could include filenames for payloads, and using these tags to search the encrypted payloads.” Pet. 97; *see also id.*, 106 (referring back). But Auradkar and Chiueh each teach their own different—and complex—searchable encryption techniques. The fact that Auradkar disclosed techniques using tags containing metadata (*see, e.g.*, Auradkar, ¶90) does not mean that those techniques could have been modified with Chieuh’s very different page-based “significant-

word-set” technique (*see, e.g.*, Chiueh, ¶¶8-9), let alone in the specific manner of the challenged claims. Petitioner also baldly asserts, without any explanation or support, that “creating identifiers for portions of a document had been well-known, would have been compatible with Auradkar’s system, and the combination would have yielded expected results.” Pet. 97. Not true. But even if creating identifiers was known at general level, that would not show that a POSITA would have reasonably expected success in modifying the specific searchable encryption techniques in Auradkar’s comprehensive platform with Chiueh’s own specific and very different technique. In fact, as Dr. Black explains, they are incompatible. Ex. 2001, ¶¶108-109.

Moreover, Petitioner does not explain (or even directly allege) that a POSITA would have had a motivation or reasonable expectation of success in implementing the portions of Chiueh’s searchable encryption technique on Chambers’ gateway system, as Petitioner’s purported combination would require. This, alone, is fatal to the combination, given that Chiueh expressly teaches its technique should only be performed by the client and not on remote systems like Chambers’ gateway. *See* Chiueh, ¶¶5, 8-10; Ex. 2001, ¶110.

Hence, Petitioner fails to show a motivation to combine and reasonable expectation of success in combining Chiueh with Aradkar and Chambers to achieve the claim limitations requiring “creat[ing] namespaces for one or

more of the [] chunks” and “configur[ing] the namespaces of the one or more chunks such that content of the file is encrypted in a manner that makes it searchable.” Ex. 2001, ¶111. For this reason as well, Ground 4 fails.

C. Petitioner Fails to Establish or Sufficiently Explain the Alleged Combination of Auradkar/Chiueh With Chambers

As discussed above, independent claim 1 recites that multiple method steps—including “assigning,” “encrypting,” and “storing”—are performed “by a gateway device.” *See supra* VI.D.; Ex. 1001, claim 1. Petitioner concedes that, at a minimum, Auradkar fails to disclose that the claimed “assigning” step is performed “by a gateway device.” Pet. 83 (“Auradkar ... did not explicitly disclose a *gateway device* assigning the policy”). Relatedly, independent claim 16 recites multiple stored modules—including “a policy assignment module,” “an encryption module,” “a storage module,” and “a management module”—that each perform claimed processes. Ex. 1001, claim 16. And Petitioner implicitly concedes that, at a minimum, Auradkar fails to disclose the claimed “policy assignment module.” *See* Pet. 115-116.

In an attempt to address these deficiencies, Petitioner alleges that a POSITA would have “incorporat[ed] Chambers’ dedicated gateway as an interface between the client and protected data sources” (Pet. 79) and “use[d] Chambers’ gateway to implement Auradkar’s ‘assigning’ function (and other

claimed functions), and to automatically assign the pertinent security policies.” *Id.*, 85; *see also id.*, 93, 108, 115-116 (referring to same combination in connection with multiple claim limitations). But as Dr. Black explains, Petitioner does not show a motivation to combine and reasonable expectation of success. Ex. 2001, ¶¶112-114.

As an initial matter, Petitioner does not even attempt to articulate a specific motivation to combine the references (or reasonable expectation of success) with respect to multiple claim elements. Ex. 2001, ¶115. As an example, for the claim step requiring “storing, by the gateway device, the searchable encrypted file,” Petitioner baldly asserts that “a POSITA would have found it obvious to use Chambers’ gateway device to perform the functions described in Auradkar, such as storing the searchable encryption file on one or more cloud platforms based on the policy.” Pet. 108. Then, for support, Petitioner merely refers back to a prior section discussing the very different “assigning” claim step, and Petitioner does not offer any explanation or rationale for combining the references in the specific manner required by the “storing” step. *Id.*; *see also id.*, 93 (another example). Moreover, this complete lack of analysis also applies to claim 16. *See, e.g., id.*, 117 (“A POSITA would have found it obvious that Auradkar’s system would have included various modules, including the claimed ‘encryption module,’ that

would be executed by its processing unit (processor)”) (referring back); *see also id.*, 115-118 (other examples). For this reason, Petitioner cannot meet its burden with respect to the purported combination.

Petitioner also does not (and cannot) show any legitimate motivation to combine Auradkar (and Chiueh) with Chambers. Auradkar discloses a comprehensive “trustworthy platform” using “mathematical transformation techniques, such as searchable encryption, deassembling/reassembling or distribution techniques,” to “distribute[] trust across multiple entities ... and decouple[] data protection requirements ... from the containers that hold the data (e.g., file systems, databases, etc.) enabling the data to act as its own custodian through imposition of a shroud of mathematical complexity.” Auradkar, ¶¶9-13. Petitioner fails to explain how or why a POSITA would look to modify this comprehensive platform to implement Auradkar’s complex techniques on Chambers’ simple “network access device” (Petitioner’s identified “gateway” device) that cannot even perform its own existing functions without a separate “management server.” Chambers, ¶¶24-28. In fact, Chambers’ simple network access device, on its own, would be incapable of implementing Auradkar’s complex techniques, making the purported combination unworkable. As Dr. Black explains, given Auradkar

already provides a comprehensive platform, a POSITA would have had no reason to attempt the combination. Ex. 2001, ¶116.

A POSITA also would not have been motivated to make the purported combination because, as Dr. Black explains, Auradkar and Chambers are non-analogous references in different fields of art. Ex. 2001, ¶117. Auradkar teaches a comprehensive “trustworthy platform” that uses complex mathematical techniques to provide “containerless” protection of data stored by cloud services in a system of “distribute[d] trust” (Auradkar, Abstract, ¶¶9-13), whereas Chambers teaches using “network access devices” to “automatically assign[] [network] access control policies (ACPs) based on [client] device types,” where the policies specify network parameters such as “bandwidth limits and traffic shaping rules, VLAN assignment, firewall rules, whether a captive portal should be applied to that device, etc.” Chambers, ¶¶21, 24-28, 32-40. Chambers does **not** disclose using its network access control techniques in connection with encrypting, storing, or controlling access to files stored in the cloud. In fact, Chambers does not address cloud storage at all. Hence, for all these reasons, a POSITA would not have looked to Chambers’ network access device to modify Auradkar’s comprehensive platform.

Petitioner asserts that a POSITA would have been motivated to “incorporat[e] Chambers’ dedicated gateway as an interface between the client and protected data sources [because this] would have enhanced network security by, e.g., shielding these sources from the client.” Pet. 79; *see also* Pet. 87 (identifying same alleged benefit). But Petitioner ignores that one of Auradkar’s primary goals is to “[e]liminat[e] [] external entity controlled active custodianships” of data and, instead, “provide a system where data ... acts as a custodian for itself regardless of the third party machine(s), mechanism(s), device(s) or container(s) holding the data,” and “enable containerless data along with guarantees of security, confidentiality, tamper-proof, etc., which are made transparent to the user.” Auradkar, ¶¶68-70. Hence, as Dr. Black explains, a POSITA would have had no reason (or desire) to incorporate and implement Auradkar’s techniques on Chambers’ network access device as a layer of further complexity. Ex. 2001, ¶118.

Petitioner further asserts that a POSITA would have been motivated to “[i]ncorporat[e] Chambers’ teachings into Auradkar’s system [because this] would have allowed automatic assignment of a policy based on, e.g., a user’s role in the organization, or the user’s devices.” Pet. 86-87. But Chambers’ network access device only assigns policies that specify network parameters such as “bandwidth limits and traffic shaping rules, VLAN assignment,

firewall rules, whether a captive portal should be applied to that device, etc.,” and it can only assign those policies under the control of a separate “management server.” Chambers, ¶¶24-28, 32-40. Hence, a POSITA would not have looked to Chambers’ network access device’s automatic policy assignment to improve Auradkar’s system, which has no such separate “management server,” and which implements access policies to control users’ “access to data.” Auradkar, ¶¶116-117. For this reason, as well as those discussed above, there would have been no motivation to combine Auradkar (and Chiueh) with Chambers’ network access device. Ex. 2001, ¶119.

Petitioner also fails to show a reasonable expectation of success. Petitioner merely asserts, without any explanation, that a POSITA “would have reasonably expected success in making this combination because gateway devices were well-known for network security-related functionality, including controlling network traffic and managing access policies” and “adding Chambers’ gateway into Auradkar’s system would have yielded predictable results.” Pet. 87. Petitioner’s conclusory statements are untrue and insufficient to establish a reasonable expectation of success. Ex. 2001, ¶120. First, the fact that gateway devices may have been known for conventional network security functions is irrelevant because, as discussed above, conventional network security is not the subject of Auradkar’s system,

and Chambers’ network access control policies are very different from the user access policies at issue in Auradkar. Ex. 2001, ¶121. Second, as discussed above, Auradkar already taught a comprehensive platform that would have been incompatible with Chambers’ network access device (which requires a separate “management server” to function). Chambers’ network access device could not have accommodated Auradkar’s complicated searchable encryption, storage, and access control techniques, nor the complex searchable encryption technique taught in Chiueh. As Dr. Black explains, Petitioner fails to show how all these teachings could be successfully combined. Ex. 2001, ¶122.

Hence, Petitioner fails to show a motivation to combine and reasonable expectation of success in combining Auradkar and Chiueh with Chambers to meet the “assigning,” “encrypting,” and “storing” steps of claim 1 or the recited “module[s]” in claim 16. Ex. 2001, ¶123. And Ground 4 fails for this reason as well.

IX. GROUND 5 FAILS ON THE MERITS

As discussed above, Petitioner fails to establish that Auradkar in view of Chiueh and Chambers renders independent claims 1 and 16 obvious. Because Ground 5 relies on Ground 4 with regard to the limitations of the independent claims, Ground 5 also fails for the same reasons. Ex. 2001, ¶124.

X. CONCLUSION

For the foregoing reasons, Patent Owner respectfully requests that the
Petition be denied.

Date: June 20, 2023

By: /s/ James Glass
James M. Glass (Reg. No. 46,729)
QUINN EMANUEL URQUHART &
SULLIVAN LLP
51 Madison Avenue, 22nd Floor
New York, NY 10010
Email: jimglass@quinnemanuel.com
Phone: 212-849-7142
Fax: 212-849-7100

Counsel for Patent Owner
Fortinet, Inc.

CERTIFICATE OF LENGTH (37 C.F.R. §§ 42.24)

The undersigned hereby certifies that, according to the word-processing system used to prepare the foregoing document, this document has 9,466 words and thus complies with the applicable word limit.

Date: June 20, 2023

By: /s/ James Glass

James M. Glass (Reg. No. 46,729)
QUINN EMANUEL URQUHART &
SULLIVAN LLP

51 Madison Avenue, 22nd Floor
New York, NY 10010

Email: jimglass@quinnemanuel.com

Phone: 212-849-7142

Fax: 212-849-7100

*Counsel for Patent Owner
Fortinet, Inc.*

CERTIFICATE OF SERVICE (37 C.F.R. §§ 42.6(E), 42.105(A))

The undersigned hereby certifies that the foregoing document was served in its entirety on June 20, 2023 upon the following parties via Electronic Mail:

Thomas N. Millikan (millikan-ptab@perkinscoie.com)

Babak Tehrani (tehranchi-ptab@perkinscoie.com)

Andrew N. Klein (klein-ptab@perkinscoie.com)

Wei Yuan (yuan-ptab@perkinscoie.com)

Netskope-FortinetIPRTeam@perkinscoie.com

Date: June 20, 2023

By: /s/ James Glass

James M. Glass (Reg. No. 46,729)

QUINN EMANUEL URQUHART &
SULLIVAN LLP

51 Madison Avenue, 22nd Floor

New York, NY 10010

Email: jimglass@quinnemanuel.com

Phone: 212-849-7142

Fax: 212-849-7100

*Counsel for Patent Owner
Fortinet, Inc.*