

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 SUR-REPLY

TABLE OF CONTENTS

	<u>Page</u>
I. Introduction	1
II. Claim Construction	2
A. “namespace”	2
1. A “namespace” defines a particular context	2
2. An “identifier” is not a “namespace”	5
3. Creating and Configuring Namespaces is Part of the Claimed Encryption Step Performed by the Gateway Device.....	7
B. The claims require at least one policy that performs all the recited functions.....	9
III. Grounds 1-3: Cidon and Shikfa.....	10
A. Cidon/Shikfa does not create namespaces (Claims 1/16)	10
B. Cidon/Shikfa does not configure namespaces (Claims 1/16)	12
C. No motivation to combine Cidon and Hermann with Shikfa (Claims 1/16).....	13
D. Cidon/Shikfa does not disclose a user-assigned encryption policy	15
E. Cidon/Shikfa’s encryption policy is not for files to be uploaded	16
F. Cidon/Shikfa does not disclose one or more policies that perform all the recited functions (Claims 1/16 and claim 14)	16
G. Cidon/Shikfa does not disclose storing a file “across two or more cloud platforms” (Claims 11/25).....	18

IV.	Grounds 4 and 5: Auradkar, Chiueh, and Chambers.....	18
A.	Auradkar/Chiueh does not create namespaces (Claims 1/16)	18
B.	Auradkar/Chiueh does not configure namespaces (Claims 1/16)	21
C.	A POSITA would not be motivated to combine Auradkar, Chambers, and Chiueh (Claims 1/16).....	23
D.	Auradkar does not disclose “cryptographic key information defined by the policy” (Claims 1/16)	24
E.	Auradkar does not disclose a policy for “processing of a file to be uploaded on a cloud” (Claims 1/16)	25
V.	Ground 5: Auradkar and Inoue (Claim 6).....	25
VI.	Conclusion.....	26

EXHIBIT LIST

No.	Exhibit
2001	Declaration of John Black Jr., Ph.D.
2002	Curriculum vitae of John Black Jr., Ph.D.
2003	Supplemental Declaration of John Black Jr., Ph.D.
2004	Transcript of the Testimony of Dr. Wenke Lee (December 13, 2023)
2005	Transcript of the Testimony of Dr. Wenke Lee (May 16, 2024)

I. INTRODUCTION

Petitioner's construction of "namespace" is unreasonably broad, is not consistent with the patent's disclosure of namespaces, and ignores the testimony of Petitioner's own expert, Dr. Lee. The parties agree that "namespace" includes "the set of names available for naming things." Petitioner, however, argues that a namespace need not name things for a ***particular context***. In so doing, Petitioner's construction does not capture the specification's repeated disclosures that creating a namespace involves creating a folder (*i.e.* a particular context) in a cloud storage service to store particularly-named file chunks. Petitioner does not point to any other embodiment where the creation of the namespace only defined the set of names available for naming things.

Turning to the prior art, Petitioner alleges that the art's ***identifiers*** qualify as "***namespace[s]***." If the Board rejects Petitioner's construction, as it should, then Petitioner's grounds fail, as Petitioner has failed to identify any contexts for the cited identifier. But even if Petitioner's proposed construction is adopted, Petitioner's grounds still fail, since an identifier is not a "set of names available for naming things," as Petitioner's contends—instead, an identifier is, at best, merely one of the names ***within*** a namespace that was previously made available.

Petitioner also fails to show that the prior art discloses at least one policy that performs all of the recited functions required in the claims. Petitioner merely suggests that the prior art's disparate policies can be considered as sub-policies within one policy, but does not provide any obviousness analysis for this new argument, and in any case, it is clear that some of these policies are directed at users whereas others are directed at files.

II. CLAIM CONSTRUCTION

A. "namespace"

1. A "namespace" defines a particular context

Petitioner argues that "namespace" should mean "the set of names available for naming things." Reply 1. Patent Owner proposes that "namespace" should mean "a set of names available *within a particular context*." POR 18. Thus, the dispute between the parties is whether a namespace defines names for a particular context, or if the names are allowed to be contextless. The intrinsic and extrinsic evidence, as well as both parties' experts, support Patent Owner's construction.

The POR explains in detail why Patent Owner's construction is correct in view of the intrinsic and extrinsic evidence, including Dr. Lee's testimony. Petitioner, by contrast, does not analyze the intrinsic evidence at all, and on that basis alone, its proposed construction should be rejected. Indeed,

Petitioner's only reason for rejecting Patent Owner's construction is that "within a particular context" is allegedly "vague and encapsulates practically anything." Reply 2. Not so.

The '678 patent and both experts provide numerous examples of a namespace that defines a particular context. The embodiments explain that for each file to be encrypted, its encrypted file chunks are stored in a folder which is named with "the encrypted version of the original file name"—thus creating a namespace (*i.e.* context) for the chunks to be named. POR 20; Ex. 1001, 13:2-7; Ex. 2003, ¶71. Folders are classic examples of a namespace, because they define a context for the files to be named, as Dr. Lee extensively testified. Ex. 2004, 38:18-23, 35:19-36:8, 39:11-40:1. This further comports with other examples in the specification of namespaces—such as that "namespaces" are "[e]xamples of storage containers." Ex. 1001, 6:44-47; *see also* 9:23-26 ("data stored to be stored in the namespaces ... of one or more cloud platforms"), 10:12-15 ("one or more containers (e.g., ... namespaces) of cloud service providers 206 can store metadata and/or content associated with searchable encrypted files.").

Indeed, by removing the requirement of a "context," Petitioner's construction is not consistent with these disclosures that repeatedly describe namespaces as folders or storage containers. *Phillips v. AWH Corp.*, 415 F.3d

1303, 1315 (Fed. Cir. 2005) (“[T]he specification is always highly relevant to the claim construction analysis. Usually, it is dispositive; it is the single best guide to the meaning of a disputed term.”).

The extrinsic evidence, including Petitioner’s expert testimony, also supports Patent Owner’s construction. Indeed, Dr. Lee’s declaration—filed before Petitioner raised its current construction—was **unequivocal** that a namespace defines a particular context. Ex. 1002, ¶¶41-42. For example, Dr. Lee recognizes that “[a] namespace ensures that every object within the **context** of the namespace can be identified uniquely by its name(s).” *Id.*, ¶42. He also states that “a namespace can provide **the scope or context** of identifiers ... such that an identifier is referred to by its unique short, local name **within the namespace** and by its unique fully qualified name (e.g., that includes the name of the scope) from **outside the namespace**.” *Id.* Moreover, the dictionary Petitioner relies on states that “[i]f two parts of a program have different namespaces, the same name can be used **in both places** for different purposes without conflict,” indicating that a namespace also defines a **place** (*i.e.* context) for the names. Ex. 1014.

Dr. Lee also testified repeatedly in his deposition that a namespace defines a context for the names. *See, e.g.*, Ex. 2004, 35:10-12 (“[T]he namespace **does have a scope** of the objects that it’s identifying.”); 37:1-5

(“[W]hen you talk about namespace, first of all you talk about *what’s a scope* of this current namespace.”). For example, he frequently described a “document directory” as a “namespace” where “each file within the document directory can have a unique name within the document directory.” *Id.*, 35:11-14; *see also* 37:4-5 (“The namespace is ‘documents’. That means the scope is everything within the directory ‘documents’”). Thus, a “file Foo” can exist in a “Wei” directory but a different file also named Foo can exist in a “Wenke” directory, and that does not cause any issues because “*within each namespace*,” that is, within each directory, “this file Foo is unique”—“the name is unique within this directory.” *Id.*, 35:15-36:1.

Accordingly, all intrinsic and extrinsic evidence favors Patent Owner’s construction. Petitioner’s grounds are based on a theory that a “namespace” could include the prior art’s individual identifier. An individual identifier, however, lacks context, and therefore cannot be a “namespace” as that term is properly construed.

2. An “identifier” is not a “namespace”

Even under Petitioner’s construction, an identifier is not “the *set of names* available for naming *things*.” A “namespace” is not created whenever an identifier is generated. Creating an identifier does not create a set of names nor does it make those names available for naming things.

Indeed, during Dr. Lee’s second deposition, he frequently distinguished an identifier from a namespace. On the one hand, “a namespace normally tells you the naming *convention* and *the range of ... values*.” Ex. 2005, 32:7-9. And from that range of values, an “identifier [is] chosen.” *Id.*, 32:20-21. Dr. Lee also testified that “with a namespace, basically each object has a unique identifier ... *within the scope of the context of the namespace*.” *Id.*, 38:7-9. Thus, an identifier is a name *within the range of values defined by a namespace*—it is not the namespace itself. POR 19, 23-24; Ex. 2003, ¶69, ¶¶77-78; see also Pet. 10 (“[A] namespace ensure that every object within it can be identified uniquely by its name(s).”).

In response, Petitioner argues that generating multiple identifiers creates a namespace because it creates a set of names. However, both parties’ constructions contemplates that the set of available names is created *before* any objects are named—because they are the set of names *available for naming things*. Generating names for objects does not create *available names*, because those names have already been used. In addition, this is consistent with the evidence and the experts’ testimony. Again, Dr. Lee testified that the namespace is the “naming convention” and defines “a range of ... values” from which an “identifier [is] chosen”—*i.e.* the namespace must be created before identifiers are generated for objects. Ex. 2005, 32:7-21.

Similarly, Exhibit 1023 explains that “a flat identifier might be **a number chosen** uniformly at random from $[0, 2^{128} - 1]$,” *i.e.* the namespace is defined as the range of values between 0 and $2^{128} - 1$, and an identifier is “chose[n]” from that namespace. Ex. 1023, 344. Accordingly, even under Petitioner’s construction, the cited “identifiers” are not the claimed “namespace[s].”

3. **Creating and Configuring Namespaces is Part of the Claimed Encryption Step Performed by the Gateway Device**

Petitioner argues in its Reply that “‘creating’ and ‘configuring’ namespaces are **not** part of encrypting content of the file.” Reply 2 (emphasis in original). This construction is not only contrived, it is directly contrary to the claim language. Indeed, Petitioner proposes this construction because the prior art does not create a namespace during encryption of the file chunks. Petitioner’s construction is necessary to sweep in an alleged namespace that is, at best, defined by the prior art reference (which specifies SHA-2 identifiers) and not created as part of a method for encrypting a file. *See, e.g.*, Reply 9.

Yet the language of the claims is clear: “creating namespaces” and “configuring the namespaces” are steps that belong to the “encrypting” step of claim 1 or are performed by the “encryption module” of claim 16. The encryption process, including the creation and configuration of namespaces,

must be performed “by a gateway device” (claim 1) or a “cloud storage system” (claim 16), not pre-defined by the references or some specification sheet. Moreover, the claims are clear that these steps are intertwined with the encryption of content, as the last step must “configur[e] 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, cl. 1. Thus, the creation and configuration of namespaces must occur as part of the process of encrypting a particular file.

Petitioner instead argues that the creating and configuring steps “describe how *the encrypted searchable file* is produced.” Reply 3 (emphasis in original). This, of course, is true, because the claim language makes clear that it is the “encrypting” step that “produce[s] a searchable encrypted file.” And Dr. Lee agreed that in the claim language, in order to produce a searchable encrypted file, “encrypting ... content of the file [is] one element that needs to be done,” and “[t]hen in order to produce a searchable encrypted file, meaning that, *in addition* to encrypting the content ... you need to ... divid[e] the file into chunks, creat[e] namespaces, and configur[e] the namespaces.” Ex. 2005, 19:5-14. Thus, Dr. Lee agrees that all four steps are part of a process for encrypting a particular file that must be performed by a gateway device or cloud storage system, which is how Patent Owner characterized it in the POR. *See, e.g.*, POR 19 (“The Challenged Claims recite

‘creating’ and ‘configuring’ namespaces for file chunks as part of an encryption process.”).

B. The claims require at least one policy that performs all the recited functions

Petitioner recasts Patent Owner’s argument, stating that Patent Owner “attempt[s] to limit ‘policy’ to a ‘single policy.’” Reply 4. That is not Patent Owner’s position. The Response plainly stated that “there must be **at least one policy**” that performs all of the recited functions. POR 36. Patent Owner’s position is consistent with the well-known axiom that, generally, the claim term “a” means “one or more.” The claims recite one policy that (1) manages access to a file, (2) processes a file, (3) define access rights of users, (4) defines cryptographic key information, and (5) defines how the searchable encrypted file is stored. And Patent Owner is in full agreement that there can be two of such policies that **each** meet those five functions. Or three or four. Or other policies that do not perform all five functions. There just needs to be at least one “policy” (or more) that does perform all functions.

By contrast, Petitioner’s construction is at odds with Federal Circuit precedent. POR 38-39; *Salazar v. AT&T Mobility LLC*, 64 F.4th 1311, 1318 (Fed. Cir. 2023) (“[I]t does not suffice to have multiple microprocessors able to perform just one of the recited functions; the claim language requires at

least one microprocessor capable of performing each of the recited functions.”). Petitioner attempts to distinguish this precedent by arguing the ’678 patent discloses a policy that “comprises multiple policies or sub-policies, addressing *different* functions.” Reply 5 (emphasis in original). But the ’678 patent makes clear that it is still **one policy** that may have a “key management sub-polic[y]” (the only disclosure of a sub-policy in the patent). Ex. 1001, 9:40-41. Whether or not the one policy is implemented as multiple sub-policies is irrelevant—it is still one policy that performs all recited functions as the claims require.

III. GROUNDS 1-3: CIDON AND SHIKFA

A. Cidon/Shikfa does not create namespaces (Claims 1/16)

Petitioner’s Cidon/Shikfa combination does not create a namespace when it generates an identifier for a file chunk because, as discussed above, an “identifier” is not a namespace under either party’s construction. POR 23-25. Creating a single identifier does not create a set of names available for naming the file chunks, nor does it create a set of names for a particular context. At best, an identifier is merely one name within a namespace. *Id.*; see also Ex. 2005, 38:7-9 (“[W]ith a namespace, basically, each objects has a unique identifier ... ***within the scope of the context of the namespace.***”).

Petitioner's Reply arguments do not mention Petitioner's own construction of namespace, much less address how the prior art meets it. Reply 6-8. Indeed, Petitioner's argument that "[n]amespace creation does not require defining a range of names" cannot be reconciled with Petitioner's proposed construction or its expert's testimony. Reply 6; Ex. 2005, 32:7-9 ("[A] namespace normally tells you the naming **convention** and **the range of ... values.**").

Petitioner argues that the claim's recitation of "creating namespaces for one or more of the plurality of chunks" encompasses generating a single identifier. Reply 6-7. This is incorrect under either Petitioner or Patent Owner's construction of namespace, both of which require defining the **set of available** names. Even if only one object is to be named, the claims require that the set of available names for that single, to-be-named object must still be defined. Generating a single SHA-2 hash does not define a set of **available** names to be given to future objects.

In addition, generating a single SHA-2 hash does not **create** a particular context. In the '678 patent, creating a folder creates a particular context for one or more file chunks to be named (since they are stored in the folder)—this is part of the embodiments that allow for searching of encrypted files using the cloud service's search functionalities. Cidon does not disclose anything

similar. Petitioner argues that “Cidon’s namespaces were created when identifiers 1452 were assigned to file segments 1451, 1450, thus creating a unique set of names within the context of encrypted file 1460.” Reply 8. But the encrypted file 1460 was not created as a scope/context for naming the file segments—instead, it is the concatenation of the file chunks *after* they have been assigned identifiers. Thus, the encrypted file 1460 could not possibly be a context created for naming objects when its creation occurs after said objects are already named.

B. Cidon/Shikfa does not configure namespaces (Claims 1/16)

Petitioner reiterates its argument that the prior art’s identifiers are namespaces. But Petitioner ignores a critical argument on this limitation raised in the POR. Specifically, the POR pointed out that the assignment of identifiers to keywords has no bearing on the manner in which a file’s content is encrypted, thus failing to meet the claim’s requirement that the configuration of namespaces is done “such that content of the file is encrypted in a manner that makes it searchable.” POR 27-28; Ex. 2003, ¶¶85-87; Ex. 2004, 136:13-137:7 (Petitioner’s expert confirming that the use of identifiers in the index is “independent” and “orthogonal” to the manner of encryption of the document). This is because, in the prior art, the keywords are mapped to identifiers in a local index, after which the document is encrypted—how

the document's content is encrypted has nothing to do with how the keywords are mapped to identifiers. *Id.*

Petitioner's Reply ignores this vital language from the claim, instead arguing that the claims do not require "configuring" to be part of encrypting content of the file—which is at odds with the plain language of the claim: "configuring the namespaces ... such that ***content of the file is encrypted in a manner that makes it searchable.***" The configuring step and the encryption of content are intertwined and cannot be "independent" or "orthogonal" of each other. Ex. 2004, 136:13-137:7. Petitioner also argues that mapping a keyword to an identifier in the index is part of the encryption process, yet does not dispute that this has no effect on the ***manner*** in which the file is encrypted. Reply 12.

C. **No motivation to combine Cidon and Hermann with Shikfa (Claims 1/16)**

Petitioner argues that in the POR, Patent Owner "overlooks the benefits of adding Herrmann's gateway beyond Shikfa" (Reply 13), but does not identify or explain any teachings in Hermann that would benefit Shikfa's system.

Petitioner also argues that Shikfa "did not preclude incorporating a gateway with a similar encryption functionality in the Cidon-Herrmann

combination.” Reply 13. Shikfa does, in fact, preclude such a combination—because Shikfa emphasizes that the server should not “access and/or discover the content of the client’s data” and “does not know the encryption used on the ... documents” encrypted by the client device. Ex. 1006, ¶48. In Shikfa, the document is encrypted on the client device and therefore cannot (and should not) be accessed by the Cidon/Herrmann gateway. Shikfa’s teachings are also a clear criticism and discrediting of encryption performed by the Cidon/Herrmann gateway. *Meiresonne v. Google, Inc.*, 849 F.3d 1379, 1382 (Fed. Cir. 2017). Petitioner is also wrong that Shikfa taught “broader principles of searchable encryption ..., including those where a gateway device performs encryption” (Reply 44) because Shikfa expressly states that the gateway device should not perform encryption. Ex. 1006, ¶48 (“[T]he 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 14. Thus, the broker system 10 provides searchable encryption *while maintaining the confidentiality of data external to the client 16.*”).

D. Cidon/Shikfa does not disclose a user-assigned encryption policy

The claims require a *user-assigned* policy that performs processing of a file, which Petitioner maps to Cidon’s encryption policy. As the POR explained, Cidon’s encryption policy is file-specific, not user-assigned. POR 31-34.

Petitioner argues that Cidon’s encryption file was user-assigned. Reply 15-16. Specifically, Petitioner argues that “Cidon’s encryption policy ... was part of a comprehensive policy that governed file processing and access relating to users.” Reply 16. Yet Cidon does not disclose anywhere that its encryption policy is part of a “comprehensive policy,” nor could it because, as disclosed in Cidon, explained in the POR, and not disputed by Dr. Lee, Cidon’s encryption policy is *file-specific*. POR 31-34; Ex. 1004, ¶207 (“Policies can be *defined on documents* which are ancestors of other documents.”); Ex. 2004, 57:6-9 (“[E]ncryption [i]s one of the policies you can define *on a document*.”); 48:14-17 (“B is a child of A *regardless of who owns B or who owns A*.”).

Petitioner also cites to Cidon’s “access and sharing permissions,” “copy control,” and “alert[ing of] users” for placement policy violations. Reply 16 (citing Ex. 1004, ¶¶207-213, 217). These are expressly *not* the encryption

policy, which is listed separately. Ex. 1004, ¶212. Moreover, none of these indicate that a file-based policy is user-assigned.

E. Cidon/Shikfa’s encryption policy is not for files to be uploaded

The POR argued that Cidon encrypts files that were already uploaded to the cloud, and not files “to be uploaded on a cloud,” as claimed. POR 35; Ex. 2004, 87:4-7 (Dr. Lee confirming that encryption does not occur until after “the user uploads an unencrypted file”). Petitioner does not dispute that Cidon only encrypts files after upload to their final cloud location.

Petitioner states that “Cidon described encrypting specific files as per policy and its server managing these encrypted files’ cloud upload.” Reply 17 (citing Ex. 1004, ¶¶224, 227, 57, 221). But all of those citations confirm that encryption (*i.e.* what Petitioner maps to “processing a file”) occurs *after* upload to their final cloud location. Petitioner does not cite to a single disclosure where encryption occurs before the files arrive at that location.

F. Cidon/Shikfa does not disclose one or more policies that perform all the recited functions (Claims 1/16 and claim 14)

As discussed above, the claims require one or more policies that performs all recited functions, including managing access to a file, processing a file, and defining how the file is stored. Claim 14 further requires that the policy defines storage locations for the file. As the POR explained, Cidon’s

access policy, encryption policy, and placement policies are all different policies and therefore do not meet the claims. POR 35-41, 42-43.

With respect to Cidon's encryption policy, Petitioner does not explain how it is the same policy as the access policy (or sub-policies within the same policy), nor cite to anything in Cidon disclosing that it is the same policy. Reply 17-18.

With respect to Cidon's placement policies, Petitioner does not explain how it is the same policy as the encryption policy or access policy (or at least sub-policies within the same policy). Petitioner merely states that "Cidon outlined a comprehensive policy framework that encompassed both placement and access management functions," which is not an explanation for how Cidon discloses a single policy. Reply 18-19. An overarching "policy framework" is not a particular "policy," as claimed.

With respect to claim 14, Petitioner's only argument was that it was "obvious that the file locations would be part of Cidon's security policy that could also define (and manage) such location information." Pet. 64. The POR explained that the Petition did not identify what "Cidon's security policy" referred to, and Cidon did not disclose any policy that manages location information. Petitioner states that this was an obviousness argument (Reply 20), but merely stating something is obvious does not make it so—Petitioner's

obviousness argument does not explain how Cidon would be modified because Cidon does not explain Cidon's security policy. Petitioner also does not explain how such a policy would be the same policy as the other policies. For those reasons, Petitioner's obviousness argument fails.

G. Cidon/Shikfa does not disclose storing a file “across two or more cloud platforms” (Claims 11/25)

Petitioner argues that “archiving” and “backup” limitations allow for storing singular copies of each file across platforms. Reply 19. However, only one of Petitioner's citations to the '678 patent discusses storage “across multiple cloud platforms,” and it does not mention singular copies of files. Although the quotation refers to “redundancy,” that does not necessarily mean a copy of a file. A POSITA would understand that storing chunks across multiple cloud services could also involve providing redundancy. For example, RAID 5 was a known method of storing data across multiple disks that also provided for redundancy.

IV. GROUND 4 AND 5: AURADKAR, CHIUEH, AND CHAMBERS

A. Auradkar/Chiueh does not create namespaces (Claims 1/16)

As with Cidon/Shikfa, the POR explained how creating identifiers in Auradkar (XML tags) and Chiueh (document ID) does not create a namespace. Reply 45-47. As discussed above, under either party's

construction, creating a namespace requires creating the set of available names. Generating an identifier is, at best, assigning a name from within a namespace—it does not make any set of names **available** for naming future objects. *Id.* The set of available XML tags in Auradkar or document IDs in Chiueh was created and defined by the references, not the gateway device as claimed.

As discussed above, a namespace also requires a particular context for those names. Petitioner argues that “namespaces came into existence as these identifiers were being created and assigned to particular file segments, as these identifiers now became a set of unique names within the context of the encrypted file(s).” Reply 23-24. But the XML segments and document IDs are not names within the context of an encrypted file—they are, at best, identifiers outside of any context, as the identifier is not **particular** to a context.

In addition, Petitioner also argues that identifiers can be namespaces because of the “broader functional aspect of namespaces.” Reply 21. But this ignores the undisputed evidence that identifiers are not namespaces from Dr. Lee and Dr. Black. Dr. Black’s testimony that a “name can be almost anything” clearly does not refer to a namespace but rather a name within a

namespace—a namespace can use anything to define the set of available names.

Petitioner continues to argue that by creating the XML tags or document IDs, that creates a namespace, because the XML tags and documents IDs are a “set of names available for naming.” Reply 22-23; 24 (“namespaces came into existence as these identifiers were being created and assigned to particular file segments, as these identifiers now became a set of unique names”). Yet, as explained above, this is not consistent with either party’s construction where it must create “a set of names ***available for naming things***.” Generating identifiers does not create a set of names ***available for naming things***, because those identifiers have already been used as names and are therefore not ***available for naming things***. Both party’s construction (and all the evidence) contemplates that the creation of the namespace occurs ***before*** the objects are named. Ex. 2005, 32:7-21 (“a namespace normally tells you the naming ***convention*** and ***the range of ... values***” from which “an identifier [is] chosen”); Ex. 1023, 344 (“a flat identifier might be ***a number chosen*** uniformly at random from $[0, 2^{128} - 1]$ ”).

B. Auradkar/Chiueh does not configure namespaces (Claims 1/16)

The Reply's arguments for this limitation largely rehash that identifiers are namespaces (Reply 26-29), which as explained above, is wrong under either party's construction.

As with Cidon/Shikfa, the POR argued that the mapping of identifiers to keywords in an index has no bearing on the manner in which a file's content is encrypted, as required by the claims. POR 49-50; Ex. 184:24-185:8 (confirming that creation of the index is "orthogonal" to the encryption of the document). This is because the keywords are mapped to identifiers in a local index, after which the document undergoes an encryption process—how the document's content is encrypted thus has nothing to do with how the keywords are mapped to identifiers. POR 49-50; Ex. 1011, ¶21.

The Reply does not address the POR's arguments and instead rehashes the Petition's arguments. Reply 29. The Reply does not explain how Auradkar-Chiueh's index encrypts "content of the file ... in a manner that makes it searchable." The Reply also does not address Patent Owner's argument that the index is created before a document undergoes an encryption process and therefore does not have any bearing on how the document is encrypted. Reply 49.

In the POR, Patent Owner pointed out that the preferred embodiment '678 patent practiced this step by encrypting keywords and using them as filenames of the file chunks. Reply 49-50. This allowed for searching of the encrypted file using the cloud service's built-in searching function. *Id.* As a result, the configuration of the namespace in this exemplary embodiment encrypts content of the file in a manner that makes it searchable—*i.e.* it encrypts the content of the file by using encrypted keywords as the file chunk filenames. *Id.*

The purpose of Patent Owner's argument is to provide an example disclosed in the '678 patent as to how this step is performed, to contrast it with how Auradkar or Chiueh's index has no bearing on the encryption process for the file content, since in those references, the index for the file is generated ***before*** the encryption process. POR 49; Ex. 1011, ¶21 ("the client 110 removes a part of words from the document pages that ***are still not encrypted***"). Petitioner does not respond beyond saying the limitations are not limited to the embodiment—while true, that does not satisfy Petitioner's burden in establishing that the prior art meets the limitations.

C. A POSITA would not be motivated to combine Auradkar, Chambers, and Chiueh (Claims 1/16)

The POR explains that Chiueh emphasizes the importance of performing encryption on the client device rather than a server. POR 50-51. Thus, a POSITA would not be motivated to incorporate Chiueh’s encryption processes into Auradkar’s server, because Chiueh counsels against and teaches away from the combination. POR 50-53.

Petitioner argues that the modification is to Auradkar, not Chiueh. However, that does not change the analysis—because Chiueh expressly counsels away from incorporating its encryption mechanism into a server like Auradkar’s. Reply 30. Petitioner also argues that Auradkar’s gateway “is controlled by the data-protecting entity” and is “owned by the same entity.” Reply 30-31. That is not true. In both Auradkar and Chiueh, the data that is encrypted is user data, which is not the same entity as Auradkar’s server. Thus, all of the concerns mentioned in Chiueh about how “the remote server is probably not reliable” and “the document stored in the remote server could be peeked/leaked” apply to Auradkar as well. Ex. 1004, ¶5.

D. Auradkar does not disclose “cryptographic key information defined by the policy” (Claims 1/16)

As explained above, the claims require at least one policy that performs the recited functions, including managing access to a file, processing a file, and defining cryptographic key information.

As explained in the POR, Petitioner’s reliance on Auradkar’s AES 256 encryption for this limitation fails because Auradkar is clear that AES 256 is used for secrecy classification, such as medical data. POR 53-54. As a result, this policy is different than the Enterprise Access Policy that Petitioner relied on for “a policy for managing access to and processing a file to be stored” and therefore does not meet the claimed policy. POR 54.

Petitioner’s only response is that “Auradkar taught a policy framework that is not limited to classifications of paragraphs within a document but extends to the policy for encryption mechanism, key, access, data storage.” Reply 31. But Auradkar’s disclosure of AES 256 is limited to the secrecy classification of a paragraph within a document. Ex. 1009, ¶95. Nowhere does Auradkar state that it uses the AES 256 policy for anything else, or that it has anything to do with the Enterprise Access Policy. And Petitioner does not argue obviousness for this limitation. Thus, Petitioner failed to meet their burden.

E. Auradkar does not disclose a policy for “processing of a file to be uploaded on a cloud” (Claims 1/16)

The POR explains that Auradkar’s policy for processing a document is limited to processing of documents that are already uploaded to their final location in the cloud service, instead of a file *to be uploaded*. POR 54-55. This is especially because Auradkar’s policy uses crypto access controls which only reside on the same cloud server used to store the files. *Id.*; Ex. 1009, ¶117.

Petitioner does not address these express disclosures and instead appears to pivot to Auradkar’s encryption of files. Reply 32. Yet Petitioner does not explain how Auradkar’s encryption is based on any policy. Notably, Petitioner does not cite to Auradkar in this section of the Reply.

V. GROUND 5: AURADKAR AND INOUE (CLAIM 6)

Claim 6 requires “configuring the namespaces comprises storing content of the file as filenames. In the POR, Patent Owner explained that Inoue’s keywords that are used to name a file are not *content of the file* but are instead arbitrary labels given to a photograph by the user. POR 55-56.

Petitioner does not dispute this infirmity in Inoue. Instead, Petitioner argues that Auradkar teaches extraction of keywords and that a POSITA would “incorporate Inoue’s functionality of assigning a keyword extracted from file content.” Reply 32-33. Specifically, “Auradkar’s keywords are then

used using Inoue's functionality to create a searchable filename in the combined system." Reply 33.

But Petitioner's combination rests on a false assumption. Inoue's functionality does **not** name files based on file content, but rather based on arbitrary descriptions by a user (*e.g.* the user speaking into a microphone "Paris" when observing a photo of Paris). Thus, a POSITA would not use Inoue's teachings for Auradkar's extracted keywords, since Inoue's keywords are completely different from Auradkar's keywords. POR 56-57.

VI. CONCLUSION

For the foregoing reasons, Patent Owner requests that the Board find that Petitioner has not established that the '825 patent's claims 1-3, 6, 8-19, and 21-27 are unpatentable.

Date: May 22, 2024

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 5,425 words and thus complies with the applicable word limit.

Date: May 22, 2024

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 May 22, 2024 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: May 22, 2024

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.*