

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

APPLE INC.,
Petitioner,

v.

ERICSSON AB,
Patent Owner.

Case IPR2022-00618
Patent No. 9,313,178

PATENT OWNER'S PRELIMINARY RESPONSE

TABLE OF CONTENTS

I.	Introduction	1
II.	Background of the '618 Patent and the challenged claims	2
III.	Person of ordinary skill in the art	5
IV.	Claim construction.....	5
V.	The Petition fails to meet limitations 1.4/16.5 and 1.5/16.6 together.	6
	A. Limitation 1.4 corresponds to step 308 of Fig. 3 of the '178 Patent and the associated teachings, and limitation 1.5 corresponds to steps 316 and 318 of Fig. 3 and the related teachings.....	8
	B. The Petition does not demonstrate that Peterka meets limitations 1.4 and 1.5 together	17
	1..... The Petition's theory regarding limitation 1.4 cannot meet the claims as a whole.....	17
	2..... The Petition's theory regarding limitation 1.5 cannot meet the claims as a whole.....	21
VI.	Conclusion	24

EXHIBIT LIST

Exhibit No.	Description
2001	Declaration of Kayvan B. Noroozi in Support of Motion for Admission <i>Pro Hac Vice</i>

I. Introduction

The Petition fails to demonstrate a reasonable likelihood of success as to any challenged claim.

Independent claims 1 and 16 of the '178 Patent require, among other things, that the client device “detect” that a current encryption key that is being used to decrypt content will need to be replaced with a new key for reasons *other than* the natural time-based expiration of the current key, and that the client device then “*request*” a new key from the server before the current key must be replaced.

Section V.A, *infra*.

The Petition relies entirely on Peterka to meet those aspects of the challenged claims. As the Petition's own citations and Peterka's related teachings demonstrate, however, Peterka discloses an entirely different approach. As a threshold matter, no embodiment in Peterka discloses the client device “detecting” that it will need to change the current key in the future for reasons other than the time-based expiration of the current key. Moreover, in the embodiments where Peterka discloses that the server instructs the client to change keys, Peterka's server *provides* the new key to the client *with* the instruction related to the new key. The client thus does not “detect” any key rotation boundary prior to the natural time-based expiration of the current key and then “request” a new key, as

the challenged claims require. By contrast, in the only embodiments in which Peterka's client does request a new key from the server, the request is *always* based on a *time-based* expiration for the current key, and is never based on the client "detecting" any reason to change the current key for reasons *other* than the key's natural expiration time. Those embodiments thus likewise cannot meet the challenged claims.

Accordingly, the Petition fails to disclose any theory that meets the challenged claims, and institution should therefore be denied.

II. Background of the '618 Patent and the challenged claims

United States Patent 9,313,178 ("the '178 Patent"), titled "Method and System for Secure Over-The-Top Live Video Delivery," is directed to a method "for managing key rotation (use of series of keys) and secure key distribution in over-the-top content delivery." Ex. 1001 at 1, Abstract. The '178 Patent has 20 claims. The only independent claims are claim 1, which is directed to a method for handling secure distribution of content, and claim 16, which is directed to a computerized device operable as a client for handling secure distribution of content.

The '178 Patent teaches that "[a]s content delivery models move away from streaming distribution over private networks to Web-based delivery of files over

the public Internet, referred to as over-the-top (OTT) delivery, traditional content protection paradigms must be modified to support new delivery protocols, e.g., HTTP Live Streaming. For live streaming content with long or indefinite durations, use of a single encryption key for the entire duration increases the probability that the key may be compromised.” Ex. 1001 at 1:15-26. The ’178 Patent teaches that “[t]here are many protocols and methods for generating segmented content . . . each segment is independently playable, and therefore needs to be independently encrypted and decryptable. Segments are typically of a fixed duration and, in the case of video content, begin with a key-frame and contain no inter-segment references. Segmentation is performed on each of the different encoding generated by the transcoder, by parsing the resultant encoding and determining segment boundaries.” Ex. 1001, 2:46-58. “Segments are encrypted on segment boundaries using the current content encryption key and current initialization vector (IV). . . . The generation of new strongly random values for use as content encryption keys and the rotation of content encryption keys provides protection from content encryption keys being compromised in long lived streams.” *Id.*, 2:62-3:5.

Exemplary claim 1 correspondingly recites:

Element	Claim 1
[1.P]	A method for handling secure distribution of content comprising:

[1.1]	initiating a media playback request and receiving a playback request response;
[1.2]	parsing content information from the playback request response, the content information including content encryption keys, content encryption key identifiers, and content encryption key expiration times;
[1.3]	retrieving content and manifest files from a content delivery server;
[1.4]	detecting content encryption key rotation boundaries between periods of use of different content encryption keys in decrypting retrieved content;
[1.5]	issuing requests to a license server ahead of a key rotation boundary to retrieve a second content encryption key to be used after a content encryption key rotation boundary is reached; and
[1.6]	applying the second key for content decryption after the key rotation boundary is reached.

The Petition challenges claims 1-20. The Petition's Grounds are summarized below.

Ground	References	Basis	Challenged
--------	------------	-------	------------

			Claims
1A	Peterka, Bocharov	§ 103	1-4, 6-7, 12-13, 16-20
1B	Peterka, Bocharov, Peterka308, Chen	§ 103	7-9, 14-15, 19
1C	Peterka, Bocharov, Balraj	§ 103	5
1D	Peterka, Bocharov, Kelly	§ 103	10
1E	Peterka, Bocharov, Kelly, Eisen	§ 103	11

III. Person of ordinary skill in the art

For purposes of this Preliminary Response, Patent Owner does not dispute the Petition's proposed level of skill for a person of skill in the art ("POSA"), because the level of skill in the art is not necessary for addressing any disputes between the parties.

IV. Claim construction

The Board does not construe claim terms that are not in dispute. *Shenzhen Liown Electronics Co. v. Disney Enterprises, Inc.*, IPR2015-01656, Paper 7 at 10 (Feb. 8, 2016) (citing *Vivid Techs., Inc. v. Am. Sci. & Eng'g, Inc.*, 200 F.3d 795, 803 (Fed. Cir. 1999)). The Petition does not raise a dispute as to any claim construction issues. Pet. at 6 ("the Board does not need to construe any terms to

resolve the arguments presented herein"). Patent Owner does not believe the Board need resolve any claim construction issues at this stage. Nonetheless, to the extent the Board believes Patent Owner's arguments herein raise claim construction issues, Patent Owner provides the evidentiary bases and arguments in support of its positions below.

V. The Petition fails to meet limitations 1.4/16.5 and 1.5/16.6 together.

Under the Board's regulations, "the petition *must* set forth," "*must* identify," and "*must* specify where *each element* of the claim is found in the prior art patents or printed publications relied upon." 37 C.F.R. § 42.104 (b)(4) (emphasis added).

The Board may not reach a conclusion of unpatentability where the Petition fails to provide evidence that the prior art meets *every* element of a challenged claim. *See, e.g., Wireless Protocol Innovations, Inc. v. TCT Mobile, Inc.*, 771 F. App'x. 1012, 1016-18 (Fed. Cir. 2019) (reversing because there was "no reasonable support" for the conclusion that the prior art met the "only after" limitation of the claim); *Nidec Motor Corp. v. Zhonghan Broad Ocean Motor Co.*, 851 F.3d 1270, 1273-74 (Fed. Cir. 2017) (reversing because the challenged claim required the recited signal to be "in the rotating frame of reference," whereas the evidence did not show that the prior art taught a signal in a rotating frame of reference); *IBM v. Iancu*, 759 F. App'x. 1002 (Fed. Cir. 2019) (reversing because

there was “no substantial evidence . . . that Mellmer teaches” the “single-sign-on limitation” of the claims); *Innovative Memory Sys., Inc. v. Micron Tech., Inc.*, 781 F. App'x 1013, 1017-18 (Fed. Cir. 2019) (vacating where the record failed to show that the prior art actually met every limitation).

Thus, when a Petition side-steps an element of the challenged claims, or otherwise fails to “specify where *each element* of the claim is found in the prior art . . . relied upon,” institution must be denied.

The Petition's Ground 1A relies on U.S. Pub. 2002/0172368 (“Peterka”) in view of U.S. Pub. 2010/0235528 (“Bocharov”). As shown below, Ground 1A fails to establish that this combination teaches or suggests limitation 1.4 and 1.5 together, *i.e.*, “detecting content encryption key rotation boundaries between periods of use of different content encryption keys in decrypting retrieved content” and “issuing requests to a license server ahead of a key rotation boundary to retrieve a second content encryption key to be used after a content encryption key rotation boundary is reached.”¹

¹ The Petition treats limitations 16.5 and 16.6 identically with limitations 1.4 and 1.5. Pet. at 44. Accordingly, the arguments herein as to limitations 1.4 and 1.5 apply equally to limitations 16.5 and 16.6 and claim 16 as a whole.

A. Limitation 1.4 corresponds to step 308 of Fig. 3 of the '178 Patent and the associated teachings, and limitation 1.5 corresponds to steps 316 and 318 of Fig. 3 and the related teachings

According to the '178 Patent, a client must obtain content encryption keys from the license server before it can decrypt and render encrypted content. *Id.*, 6:41-44. The license server stores the content encryption keys, content encryption key identifiers, content encryption key lifespans (or expirations), and the locations of the encrypted content. *Id.*, 6:54-57.

Crucially, the '178 Patent teaches that the workflow manager may issue a new *unsolicited* content encryption key. *Id.* at 7:3-4. The new *unsolicited* content encryption key may be pushed under three different scenarios. “In one embodiment, the WFM 102 pushes the new content encryption key to the packager 104 when the current content encryption key is nearing the end of its lifespan.” *Id.* at 7:4-7. In a second embodiment, “the new content encryption key is pushed ahead of the current content key expiration, and the packager 104 waits until the current content encryption key has expired before applying the new content encryption key.” *Id.* at 7:7-11. However, in a third embodiment, “the new content encryption key is pushed ahead of the current content encryption key with *explicit instructions* to apply the key as soon as possible, in which case the packager 104

does not wait until the current content encryption key has expired before applying the new content encryption key.” *Id.* at 7:11-16 (emphasis added).

The '178 Patent further explains that the WFM 102 may, for instance, “push[] the new content encryption key to the packager 104 when the current content encryption key is deemed to be no longer secure (e.g., if the content encryption key has been compromised).” *Id.* at 7:17-20.

Consistent with the above teachings, Fig. 3 of the Patent (shown below) depicts a flow chart describing a process for detecting the rotation of content encryption keys and retrieving updated content encryption keys.

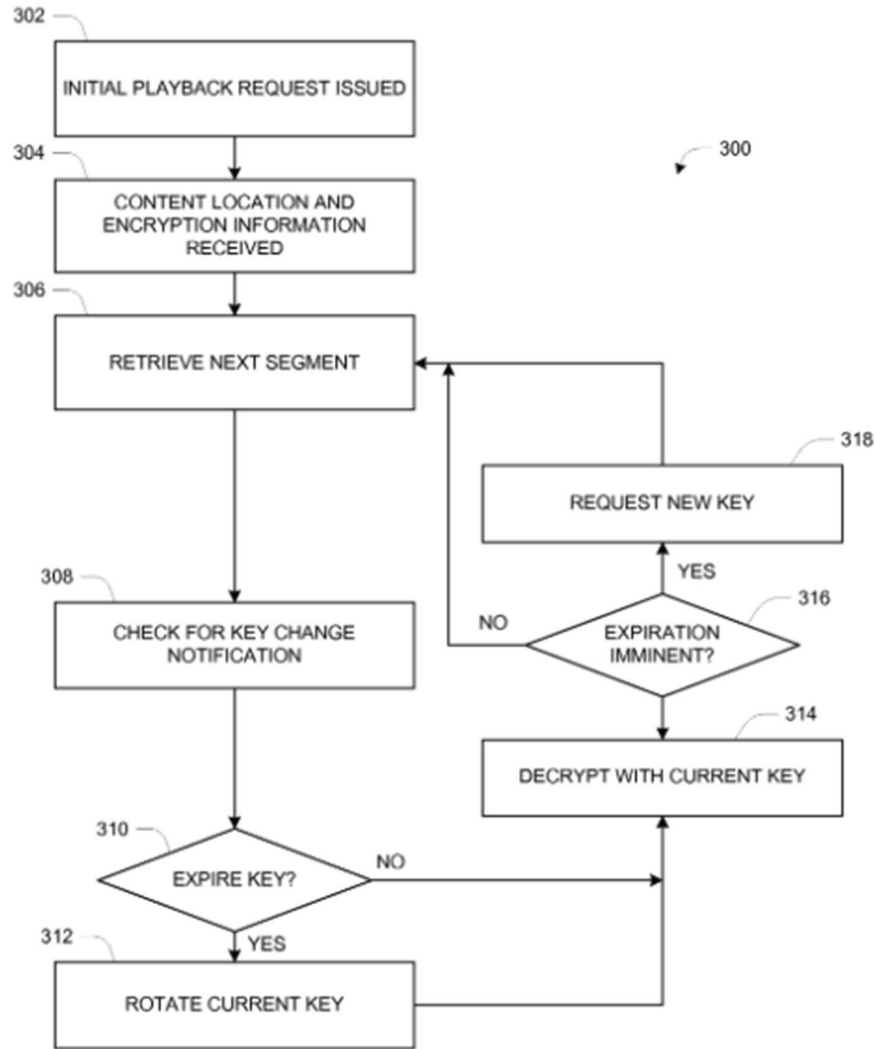


Fig. 3

At step 308, “the client 110 checks to see if a key change is signaled, i.e., *a key change was requested outside of any fixed duration lifespan.*” *Id.* at 10:39-43 (emphasis added). “If a content encryption change request is *detected*, the client notes the need to expire the current content encryption key and the content

encryption key identifier of the new key to be used. The client 110 then proceeds to step 310.” *Id.* at 10:65-11:2 (emphasis added).

“In step 310, the client 110 checks the expiration of current content key to see if a new key should be applied. ***There are two cases—normal period-based expiration and an explicit key change notification received in step 308 that applies to the current segment being decrypted.*** If the current content encryption key has not yet expired, processing proceeds to step 314. If the current content encryption key needs to be expired, processing proceeds to step 312 where the key is ‘rotated’, i.e., the current content encryption key is expired and a new one is put into use.” *Id.*, 11:3-12 (emphasis added).

The client 110 “first checks whether the content encryption key corresponding to a key change notification in step 308 has already been retrieved by prefetching. . . .” *Id.* at 11:12-15. “If the new content encryption key has not been retrieved yet (e.g., due to early expiration of the current content encryption key for security reasons, or content encryption key retrieval failures in step 318), the client 110 must issue a new content encryption key request to the license server 106 to retrieve the new content encryption key, then replace the current content encryption key with the newly retrieved content encryption key.” *Id.*, 11:18-25.

Claim 1 limitation 1.4, and claim 16 limitation 16.5, are directed precisely at Fig. 3 step 308 of the '178 Patent. Limitation 1.4 requires “**detecting** content encryption key rotation boundaries **between periods of use** of different content encryption keys in decrypting retrieved content.” The terms “detecting” and “between periods of use of different content encryption keys” refer to the scenario in which the client receives a notification instructing it to rotate a key **before** the normal period-based expiration (i.e., “between periods of use”) associated with that key. In other words, limitation 1.4 is directed to the scenario in which the client detects an instruction to change keys (i.e., detects a key rotation boundary) before the time-based expiration of the current key (i.e., “between periods of use of different content encryption keys”).

The language of limitation 1.4 is thus closely aligned with the specification's description of step 308. For instance, the specification teaches: “If a content encryption key change request is **detected**, the client notes the need to expire the current content encryption key and the content encryption key identifier of the new key to be used.” *Id.* at 10:65-11:2. The specification likewise teaches: “Once the new segment is retrieved, processing proceeds to step 308 where the client 110 checks to see if a key change is signaled, i.e., a key change was requested outside of any fixed duration lifespan.” *Id.* at 10:39-42. As noted, the '178 Specification

teaches that for key rotation boundaries, “[t]here are two cases—normal period-based expiration, and *an explicit key change notification received in step 308 that applies to the current segment being decrypted.*” *Id.*, 11:3-7 (emphasis added).

Limitation 1.4 is directed to the latter scenario, *i.e.*, detecting an explicit key change notification that applies to the current segment being decrypted, and that creates a key rotation boundary between normal periods of use, *i.e.*, before normal period-based expiration.

Dependent claims 12-15 further demonstrate that limitation 1.4 is directed to step 308 of Fig. 3 and the associated teachings. Claims 12-15 are the only dependent claims that recite narrower embodiments of the “detecting” step of limitation 1.4. Those claims further define specific embodiments related to how the notification for a *pre-period-based expiration* key change is received so that the client can detect the notification. Like limitation 1.4, these dependent claims likewise closely track various embodiments taught with respect to step 308 of Fig. 3. For instance, claim 13 recites “wherein content encryption key rotation boundaries are detected based on notifications in an unencrypted header portion of the encrypted content files,” which claims the specification’s teaching that “the packager 104 notifies the client 110 of the key change by prepending a header to the encrypted content which contains a flag that describes the expiration of the

previous content encryption key and the content encryption key identifier of the new content encryption key to be applied.” *Id.* at 10:43-48. Claim 14 recites “wherein content encryption key rotation boundaries are detected based on notifications present in the content manifest file, the content manifest file being selected from an m3u8 file, an IIS Smooth Streaming manifest file, a DASH MPD file and a proprietary manifest file.” That claim aligns directly with the specification’s teachings regarding an embodiment of step 308 of Fig. 3, in which “the packager 104 notifies the client 110 of the key change by updating a manifest file that describes the encrypted content with a flag that describes the expiration of the previous content encryption key and the content encryption key identifier of the new content encryption key to be applied. In one embodiment, the manifest may be an m3u8 file. In another embodiment, the manifest may be a Smooth Streaming manifest file. In another embodiment, the manifest may be a DASH MPD file.” *Id.* at 10:52-60. To the same effect, Claim 15 recites that “content encryption key rotation boundaries are detected based on a notification embedded in the file name of the encrypted content file.” Once again, that dependent claim—which is directed to the “detecting” step recited in limitation 1.4—aligns directly with the specification’s teachings regarding step 308 of Fig. 3. The specification thus teaches: “In another embodiment, the packager 104 notifies the client 110 of the

key change *in the file name* of the encrypted content by appending a flag that describes the expiration of the previous content encryption key and the content encryption key identifier of the new content encryption key to be applied.” *Id.* at 10:60-65 (emphasis added).

Accordingly, limitation 1.4 requires the client to detect an instruction from the server to change a current key before the normal period-based expiration of the current key while decrypting content, *i.e.*, “detecting content encryption key rotation boundaries between periods of use of different content encryption keys in decrypting retrieved content.”

Limitation 1.5 then requires “issuing requests to a license server ahead of a key rotation boundary to retrieve a second content encryption key to be used after a content encryption key rotation boundary is reached.” As demonstrated above, the ’178 Patent teaches that a “key rotation boundary” can be reached in “two cases—normal period-based expiration, and *an explicit key change notification received in step 308 that applies to the current segment being decrypted.*” *Id.*, 11:3-7 (emphasis added).

Thus, limitation 1.5 requires that the client have the ability to address *both* key rotation boundary scenarios, *i.e.*, that the client request a new key from the server before the rotation boundary of the current key *either* (1) because the

current key is naturally set to expire *or* (2) because the client detected an instruction to change the current key ahead of its normal expiration, as stated in limitation 1.4 and taught in step 308 of Fig. 3. Crucially, limitation 1.5 requires that it is the client that *requests* a new key from the server, even where the client detects a key rotation boundary instruction that requires a key rotation ahead of the natural expiration of the current key, as recited in limitation 1.4. *See* limitation 1.5 (requiring client to “issu[e] requests to a license server” for a second key “ahead of a key rotation boundary” for the current key).

Limitation 1.5 thus captures steps 316 and 318 of Fig. 3. As the specification explains: “In step 316, the client 110 checks to see if the current content encryption key is going to expire in the near future *or* if a content encryption key *change request* for a future segment was *detected in step 308*.” *Id.* at 11:38-41 (emphasis added). “If the current content encryption key is within a fixed threshold of expiring or a content encryption key change request was detected in step 308, processing proceeds to step 318 *where the client 110 requests a new content encryption key from the license server 106*.” *Id.* at 11:48-53 (emphasis added).

B. The Petition does not demonstrate that Peterka meets limitations 1.4 and 1.5 together²

1. The Petition's theory regarding limitation 1.4 cannot meet the claims as a whole

With respect to limitation 1.4, the Petition's first paragraph cites to Peterka at paragraphs 116-121 for the teachings that "when a new content key is implemented, the implementation will be signaled to the clients so that the clients can begin utilizing *the new content key with which they have been provided*," Ex. 1004 ¶ 116 (emphasis added), and that "a client can check the predetermined bit in a packet and determine the proper content key to use." *Id.* ¶ 117; *see* Pet. at 34.

Neither teaching, however, meets limitation 1.4, and certainly does not meet limitations 1.4 and 1.5 together.

First, neither of the Petition's quoted passages nor any portions of Peterka's paragraphs 116-121 teaches a client detecting a key rotation boundary ahead of the natural expiration of the current key. Peterka's paragraph 116 simply teaches that the server signals to the client that it may begin using a new key that has already been provided to it. Ex. 1004 ¶ 116 ("when a new content key is implemented, the implementation will be signaled to the clients so that the clients can begin utilizing

² The Petition solely relies on Peterka as teaching or suggesting limitations 1.4 and 1.5. Pet. at 34-36.

the new content key with which they have been provided.”). That approach neither requires nor entails any detection of a key rotation boundary ahead of a key’s natural time-based rotation boundary. And paragraph 116 does not teach that the client detects a key rotation boundary ahead of the normal expiration period that was already set for the current key, as limitation 1.4 requires.

Peterka’s paragraph 117 similarly teaches that “a predetermined bit can be used to indicate if an old or current content key should be used as opposed to a new content key which has recently been distributed to the client. Thus, a client can check the predetermined bit in a packet and determine the proper content key to use.” Ex. 1004 ¶ 117. As that passage plainly states, paragraph 117 teaches an embodiment in which a client can be instructed to use one of three keys: an old key, a current key, or a new key. The client receives that instruction in a “predetermined bit in a packet,” and follows that instruction to use the appropriate key. Under the embodiment of paragraph 117, there is no consideration of the natural time-based expiration period for encryption keys. That is demonstrated by the fact that the client can be instructed to use an “old” key, or to continue using a “current” key. By contrast, limitation 1.4 requires the client to “detect” a key change instruction or other non-time-based key rotation boundary to rotate keys prior to the natural time-based expiration of the current key. *See* Section V.A,

supra. Peterka's paragraph 117 discloses no such instruction being received by the client, nor any such detection by the client.

Second, the embodiments of Peterka's paragraphs 116-121 certainly cannot meet limitations 1.4 and 1.5 together. As explained above, limitation 1.5 requires that the ***client request*** a new key even when it detects a key change notification or other non-time-based key rotation boundary ahead of the normal expiration period for the current key. Section V.A, *supra*. By contrast, Peterka's paragraph 116 teaches the opposite, *i.e.*, the server sending the new key to the client without any client request. Ex. 1004 ¶ 116 ("when a new content key is implemented, the implementation will be signaled to the clients so that the clients can begin utilizing ***the new content key with which they have been provided.***") (emphasis added). Paragraph 117 likewise discloses that at the time the client receives the "predetermined bit" informing it which key to use, "a new key [] has recently ***been distributed*** to the client." Ex. 1004 ¶ 117 (emphasis added). In the embodiments of paragraphs 116-121, the client thus *already has* the new key, and does not issue any request for a new key, as limitation 1.5 of the '178 Patent requires. Accordingly, Peterka's paragraphs 116, 117, and 118-121 cannot meet limitations 1.4 and 1.5 ***together***, as the Petition's theory must.

The Petition's second paragraph regarding limitation 1.4 is likewise inapposite. *See* Pet. 34-35. There, the Petition relies on Peterka's paragraphs 105, 106, 114, and its Fig. 8. *Id.* As the Petition itself admits, those passages relate to an embodiment in which the client keeps track of "**known key expiry times** that allow the client to request new keys **ahead of the expiration** or key rotation boundary." Pet. at 35 (emphasis added). Peterka's paragraph 105 likewise teaches that in this particular embodiment, which it calls the "**pull model**," "each client keeps track of the keys and their expiration times and actively requests new keys before the current keys expire so as to avoid service interruptions." Ex. 1004 ¶ 105. The other passages the Petition cites—Peterka's paragraph 106, 114, and Fig. 8—all teach the same thing.

But Peterka's "pull model," and the associated teachings the Petition relies upon, cannot meet limitation 1.4. As the Petition itself states, Peterka's "pull model" changes keys entirely and exclusively based on "known key expiry times." Ex. 1004 ¶¶ 105-106. There is no key change notification received by the client ahead of the natural key expiry time of a current key, nor any other non-time-based key rotation boundary, and there is thus no such non-time-based key rotation boundary for the client to "detect," as required in limitation 1.4. To the contrary, Peterka teaches that in its "pull model," the **client** makes all key change

determinations, entirely based on the expiration time of the current key. Ex. 1004 ¶ 105 (“Under the pull model, each client keeps track of the keys and their expiration times and actively requests new keys before current keys expire so as to avoid service interruptions. *Alternatively*, the push model migrates responsibility to the server which keeps track of active clients and distributes new keys to them before the current keys expire.”); *id.* ¶ 106 (explain that under the pull model, “the server need not monitor the status of the client’s keys; rather, *the responsibility of determining when a new key is required can be passed to the client.*”) (emphasis added).

Accordingly, none of the passages the Petition relies upon from Peterka meet limitation 1.4, much less disclose an embodiment that can meet limitations 1.4 and 1.5 together, and thus the challenged claims as a whole. The Petition fails on that basis alone.

2. The Petition’s theory regarding limitation 1.5 cannot meet the claims as a whole

The Petition’s theory regarding limitation 1.5 is similarly self-defeating. The Petition relies entirely on Peterka’s “pull model” to meet limitation 1.5, arguing that Peterka discloses limitation 1.5 because it teaches that, under the pull model, the client requests a new key from the server ahead of the expiration time of the current key. Pet. at 35.

But as shown above, the pull model cannot meet limitations 1.4 and 1.5 together because, by its very design, the pull model does not entail key change notifications to the client ahead of the known expiry time of the current key, nor any other non-time-based key rotation boundaries, and the pull model thus cannot meet the “detecting” requirements of limitation 1.4. Section V.B.1, *supra*. Accordingly, the Petition’s reliance on Peterka’s pull model cannot demonstrate unpatentability of the challenged claims.

Nor can the Petition be read to meet limitations 1.4 and 1.5 based on a combination of the push and pull models together. As a threshold matter, the Petition never presents such a theory, and never proposes to modify the push and pull model aspects of Peterka’s own teachings to meet the challenged claims.

Moreover, Peterka itself teaches a “combination” push-pull model, which also cannot meet the challenged claims. Ex. 1004 ¶¶ 113-117. Specifically, Peterka states: “The push key model and pull key model can be combined in a combination model for distributing keys to clients.” *Id.* ¶ 113. Under this combination model, “a key is distributed to a client for use in decrypting program contents. The server which distributes the key *awaits confirmation* that the client received the key as shown in block 1220.” *Id.* (emphasis added). If a client confirms receipt of the new key, the server places the client on a list of active participants and continues to

push new keys out to those active clients. *Id.* ¶ 114 (“The server will obtain a list of active participants and *distribute new segment keys to them* via a multicast UPD message with the new key encrypted under the various individual unique keys for each client.”) (emphasis added). If a client *does not* confirm receipt of a new key, it is removed from the active client list. *Id.* (“If a server does not see a ‘keep alive’ message for the duration of a segment, it will remove the client from the active list.”). When a client is removed from the active client list, the server no longer pushes new keys to that client; instead, if such an inactive client needs a new key, it must *request* a new key based on the expiration time of the current key. *Id.* (“If for some reason the client does not send a ‘keep alive’ message but wants to continue receiving the contents, it can monitor the expiration time of the program segment key and send an individual key update *request* before the key expires. Again, this is a way of implementing the ‘pull’ aspect to the combination model.”); *see also* Ex. 1004 Figs. 12-13.

Thus, Peterka’s “combination” push-pull model is an *either-or* model: either the server pushes new keys to the clients in all instances (which cannot meet limitations 1.4 or 1.5), or the client requests or pulls new keys from the client *based only on the known expiration time* of the current key (which cannot meet

limitation 1.4). The Petition thus identifies no teaching in Peterka that meets limitations 1.4 and 1.5 together, and the challenged claims as a whole.

VI. Conclusion

For the reasons above, the Board should decline to institute the Petition.

Respectfully submitted,

/s/ Peter C. Knops

Peter C. Knops, Reg. No. 37,659
Kayvan B. Noroozi, *Pro Hac Vice* Pending
NOROOZI PC
11601 Wilshire Blvd., Suite 2170
Los Angeles, CA 90025
Attorneys for the Patent Owner

Date: July 8, 2022

CERTIFICATION OF WORD COUNT

The undersigned hereby certifies that the portions of the above-captioned **PATENT OWNER'S PRELIMINARY RESPONSE** specified in 37 C.F.R. § 42.24 has 4,833 words, in compliance with the 14,000 word limit set forth in 37 C.F.R. § 42.24(b)(1). This word count was prepared using Microsoft Word 365.

/Peter Knops/

Peter Knops

Counsel for Patent Owner

Date: July 8, 2022

CERTIFICATION OF SERVICE

The undersigned hereby certifies that the foregoing **PATENT OWNER'S PRELIMINARY RESPONSE** was served electronically via e-mail on July 8, 2022, on the following counsel of record for Petitioner:

LEAD COUNSEL	BACK-UP COUNSEL
W. Karl Renner, Reg. No. 41,265 Fish & Richardson P.C. 3200 RBC Plaza 60 South Sixth Street Minneapolis, MN 55402 Tel: 202-783-5070 Fax: 877-769-7945 Email: IPR50095-0082IP1@fr.com	Thomas A. Rozylowicz, Reg. No. 50,620 Nicholas W. Stephens, Reg. No. 74,320 3200 RBC Plaza 60 South Sixth Street Minneapolis, MN 55402 Tel: 202-783-5070 Fax: 877-769-7945 PTABInbound@fr.com

/Peter Knops/

Peter Knops
NOROOZI PC
11601 Wilshire Blvd., Suite 2170
Los Angeles, CA 90025
Tel.: 310-975-7074

Counsel for Patent Owner

Date: July 8, 2022