

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

META PLATFORMS, INC.; INSTAGRAM, INC.; WHATSAPP LLC; META
PLATFORMS TECHNOLOGIES, LLC; AND GIPHY, INC.

Petitioners

v.

VL COLLECTIVE IP LLC

Patent Owner

U.S. Patent No. 7,266,682

Case IPR2023-00923

**PETITIONERS' REPLY TO PATENT OWNER'S RESPONSE TO PETITION
FOR INTER PARTES REVIEW OF U.S. PATENT NO. 7,266,682**

UPDATED LIST OF EXHIBITS

Ex. No.	Description
1001	U.S. Patent No. 7,266,682 (“the ’682 Patent” or “the Patent”)
1002	Certified File History of U.S. Patent No. 7,266,682
1003	Declaration of Dr. Henry Houh, Ph.D. in Support of Petition for <i>Inter Partes</i> Review of U.S. Patent No. 7,266,682
1004	Curriculum Vitae of Dr. Henry Houh
1005	M. Handley, J. Crowcroft, & C. Bormann, Very Large Conferences on the Internet: The Internet Multimedia Conferencing Architecture and the MBONE by, https://www.cl.cam.ac.uk/~jac22/otalks/cnisdn-article.pdf (“Handley”)
1006	Erol Basturk, Mihir Bellare, Chee-Seng Chow, & Roch Guérin, Secure Transport Protocols for High-Speed Networks, https://cseweb.ucsd.edu/~mihir/papers/srtp.pdf (“Basturk”)
1007	PacketCable 1.0 Architecture Framework Technical Report, https://community.cablelabs.com/wiki/plugins/servlet/cablelabs/alfresco/download?id=bddae471-01dd-43c8-8e59-d6547fcf4ffd (“PacketCable”)
1008	RTP: A Transport Protocol for Real-Time Application, IETF RFC 1889, https://datatracker.ietf.org/doc/html/rfc1889
1009	Andrew S. Tanenbaum, Computer Networks (3 rd ed. 1996) (“Tanenbaum”)
1010	Declaration of Sylvia D. Hall-Ellis, Ph.D. in Support of Petition for <i>Inter Partes</i> Review of U.S. Patent No. 7,266,682
1011	Reserved
1012	Complaint, <i>VideoLabs, Inc., et al. v. Meta Platforms, Inc. et al.</i> , Civil Action No. 1-22-cv-00680 (D. Del.)
1013	Internet Protocol – RFC 791, DARPA Internet Program (September 1981)
1014	Carrier Sense Multiple Access with Collision Detection, IEEE Standard

	802.3, IEEE Standards for Local Area Networks (1985)
1015	David D. Clark & David L. Tennenhouse, Architectural Considerations for a New Generation of Protocols, Laboratory for Computer Science, M.I.T. (1990), https://dl.acm.org/doi/10.1145/99517.99553 .
1016	Declaration of Dr. Jon Crowcroft
1017	Supplemental Declaration of Dr. Henry Houh
1018	Supplemental Declaration of Dr. Sylvia Hall-Ellis relating to Handley (EX 1005)
1019	Supplemental Declaration of Dr. Sylvia Hall-Ellis relating to PacketCable (EX 1007)
1020	Declaration of Nathaniel E. Frank-White
1021	Web Archive of UCL Web Servers Search (August 16, 2000), available at https://web.archive.org/web/20000816095229/http://www.ucl.ac.uk/search/compass/
1022	Web Archive of Mark Handley's List of Papers, Publications, and Specifications (February 10, 1999) available at https://web.archive.org/web/19990210102149/http://north.east.isi.edu/~mjh/papers.html
1023	Web Archive of Mark Handley's University of Southern California Information Sciences Institute Website (January 30, 1998), available at https://web.archive.org/web/19980130215743/http://north.east.isi.edu/~mjh/
1024	Web Archive of Dr. Jon Crowcroft's List of Papers (August 28, 1999), available at https://web.archive.org/web/19990828053320/http://www.cs.ucl.ac.uk/staff/jon/jons-papers.html
1025	Web Archive of UCL's DARPA FTP Server (February 17, 2001), available at https://web.archive.org/web/20010217183731/http://www.cs.ucl.ac.uk/darpa
1026	Web Archive of Dr. Jon Crowcroft's UCL Website (December 3, 1998), available at

	https://web.archive.org/web/19981203112130/http://www.cs.ucl.ac.uk:80/staff/jon/
1027	M. Handley, J. Crowcroft, C. Bormann, & J. Ott, <i>Very Large conferences on the Internet: the Internet multimedia conferencing architecture</i> , 31 COMPUT. NETWORKS 191 (1999), https://www.sciencedirect.com/science/article/abs/pii/S0169755298002621#preview-section-abstract (last visited Sep 25, 2023).
1028	M. Handley, J. Crowcroft, C. Bormann, & J. Ott, <i>Very Large conferences on the Internet: the Internet multimedia conferencing architecture</i> , 31 COMPUT. NETWORKS, 191 (1999).
1029	Cross-Examination Transcript of Dr. Henry Houh dated February 16, 2024
1030	Declaration of Ethan Perbohner
1031	Cross-Examination Transcript of Dr. Stuart Stubblebine dated May 15, 2024
1032	IETF RFC 1936
1033	Supplemental Declaration Of Dr. Henry Houh, Ph.D. In Support Of Petitioners' Reply
1034	U.S. Patent No. 6,526,056

Unless otherwise noted, all emphasis added.

TABLE OF CONTENTS

I.	INTRODUCTION	1
II.	PERSON OF ORDINARY SKILL IN THE ART	1
III.	CLAIM CONSTRUCTION	1
IV.	PRIOR ART REFERENCES	2
	A. Handley Qualifies as Prior Art	2
	B. Basturk Qualifies as Prior Art	2
	C. PacketCable Qualifies as Prior Art	2
V.	ANALYSIS.....	3
	A. Ground I	3
	1. Motivation to Combine	3
	2. Independent Claim 1	14
	3. Independent Claims 25-28	18
	4. Dependent Claims 2-21, and 23.....	18
	B. Ground II	19
	1. Independent Claim 1	19
	2. Independent Claims 25-28	23
	3. Dependent Claims 22 and 24	23
	4. Dependent Claims 2-21 and 23	24
	C. Ground III.....	24
	1. Motivation to Combine	24
	2. Independent Claim 1	25
	3. Independent Claims 25-28	25
	4. Dependent Claims 22 and 24	26
	5. Dependent 2-21 and 23	26
	D. Ground IV.....	26

TABLE OF AUTHORITIES

Page(s)

Cases

<i>Comcast Cable Comm., LLC. v. Rovi Guides, Inc.</i> IPR2019-00281, Paper 39 (PTAB June 26, 2020)	8
---	---

Statutes

35 U.S.C. §§102(a) and 102(b)	2, 3
-------------------------------------	------

I. INTRODUCTION

Throughout the analysis in Patent Owner's Response (Paper 26; "POR"), Patent Owner ("PO") builds its arguments on the foundation of false equivalencies and fundamental misunderstandings. PO relies entirely on (1) an incorrect understanding of *Basturk's* "on-the-fly" authentication technique, (2) a warped view that *Basturk's* secure transport protocol is a reliable delivery protocol, (3) an overly strict and improper interpretation of the Open Systems Interconnection ("OSI") reference model, and (4) cherry-picked, and often out of context, reference teachings. As a result, PO's arguments are distractions that amount to no more than unsubstantiated conclusions. As demonstrated below and in the Petition (Paper 3), a POSITA would indeed have been motivated to combine the asserted references because of their highly complementary nature and these references either disclose and/or render obvious all Challenged Claims.

II. PERSON OF ORDINARY SKILL IN THE ART

PO does not dispute the level of ordinary skill Petitioners set forth in the Petition. Petition, 9; POR, 11. PO does not set forth its own level of ordinary skill.

III. CLAIM CONSTRUCTION

PO sets forth a claim construction for "application layer," and "application protocol." POR, 14. Even under PO's proposed claim constructions, it does not render any asserted reference inapplicable, nor does it change that those references either disclose and/or render obvious all Challenged Claims. EX1033, ¶10.

IV. PRIOR ART REFERENCES

A. Handley Qualifies as Prior Art

PO contends the Petition does not show *Handley* (EX1005) qualifies as prior art. POR, 18-23. Since the filing of the POR, the Board granted Petitioners' Motion to Submit Supplemental Information. *See* Paper 28. There, the Board confirmed that *Handley* was publicly accessible by February 11, 1999, thus, confirming *Handley* is at least pre-AIA 35 U.S.C. §§102(a) and 102(b) prior art. Paper 28, 3.

B. Basturk Qualifies as Prior Art

PO does not challenge *Basturk's* prior art status in its POR. To the extent PO maintains its challenge set forth in its Patent Owner's Preliminary Reply ("POPR") (POPR, 36-40), the Board found a reasonable likelihood that *Basturk* qualifies as a printed publication based on the Declaration of Dr. Hall-Ellis (Paper 16, 18). Dr. Hall-Ellis makes clear *Basturk* was publicly available by at least September 1, 1995. *See e.g.*, EX1010, ¶¶64-65. *Basturk* is at least pre-AIA 35 U.S.C. §§102(a) and 102(b) prior art.

C. PacketCable Qualifies as Prior Art

PO's contention that the Petition does not show *PacketCable* (EX1007) qualifies as prior art defies *PacketCable's* indicia, Dr. Hall-Ellis' Declaration, and the Board's findings. POR, 40-42. *PacketCable* includes reliable indicia of publication on its face, including a 1999 copyright date, a December 1, 1999 release date, and consecutive "12/01/99" footer date stamps. Regardless of whether

PacketCable is a technical report or standard, it states “[t]his PacketCable technical report is a cooperative effort undertaken at the direction of Cable Television Laboratories, Inc. (CableLabs®) *for the benefit of the cable industry.*” EX1007, 1. As described in Petitioners’ Reply to POPR, a standard’s purpose is to be adopted and infers public accessibility. *See* Paper 14, 3. The Board found *PacketCable*’s indicia “further corroborate Petitioner’s sufficient showing for institution that CableLabs intended to disseminate the document at least to the listed member companies of the project and artisans of ordinary skill in those companies seeking to design and/or sell applicable networks.” Paper 16, 27. Dr. Hall-Ellis, “a librarian with over 50 years of experience,” confirmed *PacketCable* was publicly available no later than December 1, 1999. *See, e.g.,* EX1019; Paper 16, 18. *PacketCable* is at least pre-AIA 35 U.S.C. §§102(a) and 102(b) prior art.

V. ANALYSIS

A. Ground I

1. Motivation to Combine

a. Basturk’s “On-The-Fly” Means MAC Generation Occurs as Data Transmission Proceeds

PO misinterprets *Basturk*’s “on-the-fly” teaching. In doing so, PO contends “Basturk is referring to starting the process of computing the style of MAC during (or in parallel with) the process of *retrieving context data* for performing cryptographic operations.” POR, 24. This interpretation directly conflicts with

Basturk's plain teachings. For example, as Dr. Houh explains, *Basturk* is clear about what “on-the-fly” means. EX1033, ¶¶11-14. In full, *Basturk* states, “[i]deally, we want signatures to be computed and verified on-the-fly; i.e., MAC generation (verification) should be carried out as data transmissions (receptions) proceeds rather than before (after) these operations have completed.” EX1006, 12; EX1033, ¶12. Dr. Houh explains a POSITA would have understood *Basturk's* statement as a shorthand for two separate statements: (1) “[i]deally, we want signatures to be **computed on-the-fly**; i.e., MAC **generation** should be carried out as data **transmissions** proceeds rather than **before** these operations have completed” and (2) “[i]deally, we want signatures to be **verified on-the-fly**; i.e., MAC **verification** should be carried out as data **receptions** proceeds rather than **after** these operations have completed.” EX1006, 12; EX1033, ¶12. A POSITA would have understood these statements to mean “on-the-fly” is the generation of the MAC which is carried out as data transmissions proceeds, i.e., in parallel, and MAC verification is carried out as data receptions proceed. EX1033, ¶12.

It is not clear PO or Dr. Stubblebine appreciate or even understand *Basturk's* teachings. When asked during his cross-examination what “these operations” refer to in *Basturk's* sentence on page 12, Dr. Stubblebine responded “Yeah, it appears he’s referring to MAC generation and verification, in a sense, when he’s referring to ‘these operations.’” EX1031, 127:18-129:8. In fact, Dr. Stubblebine testified, “I

would just have to say, it's a confusing sentence.” *Id.*, 129:9-10. A POSITA at the time would not have understood “these operations” to refer to the MAC generation/verification, but rather to the data transmission/reception. EX1033, ¶14. When read correctly, a POSITA would have understood *Basturk’s* “on-the-fly” teachings means, in part, the process of computing the MAC in parallel to the transmission of the packet. EX1033, ¶¶11-14. Notably, Dr. Stubblebine failed to remember seeing the Petition, which clearly explained *Basturk’s* teachings. EX1031, 34:20-22.

PO’s alternative interpretation of “on-the-fly,” i.e., its retrieval context interpretation, does not make sense for data transmission. *Basturk* teaches that context information is immediately available during data transmission. *See* EX1006, 6. This means there is no latency from retrieving context information during data transmission. EX1033, ¶14; *see also* EX1031, 148:9-149:13. But the goal of *Basturk’s* “on-the-fly” technique is to “minimize[] latency,” a benefit that is not realized during data transmission, as there is no latency for context retrieval. EX1033, ¶14; EX1006, 7.

b. PO Misinterprets the OSI Layer Stack

PO incorrectly asserts “[t]he POSITA would have understood that if a packet requires information to be computed (e.g., a signature), that information will always have to be computed before passing the PDU to the next layer.” POR, 25.

As Dr. Houh explains, these strict boundary requirements between layers do not exist. EX1033, ¶¶22-28. The '682 Patent itself states, “[i]n practice, many applications *do not always permit exact nomination of strict boundaries* specifically between the upper protocol layers, particularly layers 5 to 7.” EX1001, 1:37-45. This discussion of the merging of layers 5-7 belies PO’s strict boundary requirements. POR, 25; EX1033, ¶22.

PO itself acknowledges that OSI layers may operate “in conjunction” with one another. POR, 25. For example, PO states “Data *transmission* does not occur until the data is passed down to the physical layer where it is transmitted (in conjunction with the link layer) as bits.” *Id.* (emphasis in original); *see also* EX1031, 167:3-10. As Dr. Houh explains, if all functions at one layer had to be completed prior to passing data down to the next layer, the lower layer would not need to work in conjunction with the upper layer and instead could perform its own function independently. EX1033, ¶24.

Dr. Houh further explains that PO’s strict interpretation of boundary requirements is further contradicted by *Basturk’s* “dedicated hardware” teachings. EX1006, 7; EX1033, ¶¶25-26. As of the '682 Patent’s priority date, a POSITA would have been familiar with hardware protocol accelerators. EX1033, ¶25. Hardware protocol accelerators allow incomplete packets, such as uncomputed header checksum values, to be passed from upper layers down to network interface

hardware that compute and fill in these values at a lower layer in the protocol stack. EX1033, ¶26. *Basturk* specifically teaches using this kind of accelerator hardware to off-load “on-the-fly” processing. EX1006, 7; EX1033, ¶26. *Basturk* applies this idea of using “dedicated hardware” to signature computations to “off-load [] signature computations and verifications to dedicated hardware.” EX1006, 12; EX1033, ¶26.

That said, PO is wrong that “Basturk’s secure transport protocol is a *transport level* protocol that specifically operates at the transport layer, not at the application layer.” POR, 26. As Dr. Houh explains, this is not how a POSITA would have understood the combination of *Basturk* and *Handley*. EX1033, ¶¶43-44. The ’682 Patent describes RTP as “the real time transport protocol” (EX1001, 6:39-40) and is simultaneously “directly above the layer 4” (*id.*), which is above the transport layer 4 (*id.*, 1:32-33). EX1033, ¶44. Thus, even though RTP is a “transport protocol,” the ’682 Patent teaches that the layers above the layer 4 transport layer are effectively application layer protocols. EX1001, 1:37-40; EX1033, ¶¶43-44. Dr. Houh also explains that a POSITA would have applied the authentication of *Basturk* to the RTP packet level in *Handley*’s internet conferencing architecture. EX1033, ¶44. As such, attaching a signature to the RTP packet in the combination of *Handley* and *Basturk* is “above the transport layer 4,”

i.e., at the application layer, and is an application protocol as disclosed by the '682 Patent. EX1033, ¶44.

c. Basturk's Secure Transport Protocol Is Not A Reliable Delivery Protocol

PO's incompatibility arguments towards reliable delivery improperly requires bodily incorporation. *Comcast Cable Comm., LLC. v. Rovi Guides, Inc.* IPR2019-00281, Paper 39 at 50 (PTAB June 26, 2020) ("Providing persuasive rationale does not require specifying how to incorporate bodily teachings of one reference into another reference or group of references."). As stated in the Petition, a POSITA would have been motivated to use *Basturk's* "on-the-fly" signature computation with *Handley's* internet conferencing architecture. Petition, 10; EX1033, §IV. Although not required to support obviousness, the Petition and this Reply meet PO's heightened standard and demonstrate how to bodily incorporate *Basturk's* "on-the-fly" signature computation with *Handley's* internet conferencing architecture. See Petition, 10-13.

PO's assumption that *Basturk's* secure transport protocol is a reliable delivery protocol and not compatible with *Handley* is based on multiple unfounded conclusions. PO takes *Basturk's* single use of the word "reliability" to analogize *Basturk's* secure transport protocol to reliable transport protocols like Transmission Control Protocol ("TCP"). See POR, 26-27; EX1033, ¶30. *Basturk* states that "[s]ecure delivery, roughly speaking, is a guarantee to the sender A that

his message was received by B and not intercepted and destroyed by the adversary. It is essentially an extension of the standard transport level reliability feature to the more adversarial setting we are discussing.” EX1006, 4. As Dr. Houh explains, a POSITA would have understood *Basturk*, and its use of “reliability,” is not teaching reliable transport features of TCP, but something entirely different. EX1033, ¶30. For example, and as PO admits, “[r]eliability is the ability to ‘recover from data that is damaged, lost, duplicated, or delivered out of order.’” POR, 27 (quoting EX2006, 10); EX1033, ¶30. In contrast, TCP does not guarantee to sender A that his message was (1) “received by B,” (2) “not intercepted” by the adversary, or (3) “destroyed by the adversary.” EX1006, 4; EX1033, ¶30. Dr. Stubblebine so confirmed: when asked “[d]oes TCP guarantee protocol messages are not destroyed by an adversary?” Dr. Stubblebine answered “[n]o . . . there is no guarantees.” EX1031, 197:6-9.

Dr. Houh further explains *Basturk*, titled “**Secure Transport Protocols** for High-Speed Networks,” is focused on *secure* delivery, not *reliable* delivery. EX1033, ¶¶31-33. The reliability feature described in *Basturk* serves the purpose of authenticating the sender of the data, which is not a feature of protocols such as TCP. See EX1006, 4. EX1033, ¶31. This reliability feature of *Basturk*, i.e., to sign the packets (authenticating the source of data) is orthogonal to the reliability features in protocols such as TCP, which deliver the data reliably. *Id.* Dr. Houh

highlights *Basturk's* teachings that an adversary can forge a reply. *See* EX1006 4; EX1033, ¶32. Namely, an adversary acknowledges the receipt of data by the recipient and can intercept and destroy packets while sending these forged acknowledgements. *Id.* Consequently, forged messages can be erroneously accepted. *Id.* As seen in *Basturk*, forged data can be reliably delivered, but the delivery of forged data can be prevented by adding a signature to the data. In other words, verifying the data came from the correct sender is orthogonal to the correct delivery of that data. *Id.* Thus, the “reliability” features in *Basturk* (guaranteeing to the sender that its message was received, not intercepted, and not destroyed by an adversary) do not overlap with the reliability features of TCP (“recover[ing] from data that is damaged, lost, duplicated, or delivered out of order”), which a POSITA would have known and understood. *Id.*; EX2006, 10; EX1033, ¶32.

Dr. Houh explains that a POSITA would have understood and found obvious that *Basturk's* authentication would have prevented unauthenticated interruption of video transmissions, like that in the Rolling Stones concert described in *Handley*. EX1033, ¶33; *See* EX1005, 13. As described above, the reliability feature in *Basturk* is to authenticate the sender of the data, which is not a feature of protocols such as TCP, whose role is to provide reliable delivery of data, i.e., deliver data accurately and in order over an unreliable network. *Id.* Incorporating reliability features like those of TCP would not have prevented Severe Tire Damage from

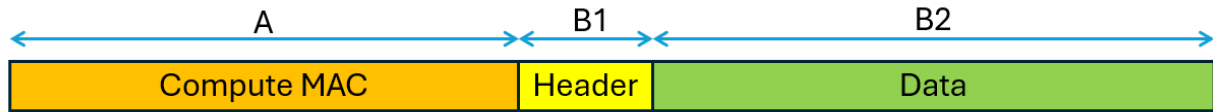
broadcasting to the same Rolling Stones multicast, though *Basturk's* authentication methods would have, which a POSITA would have known and understood. *Id.*

d. Basturk's Secure Transport Protocol and Authentication Techniques are Compatible with Handley's Internet Conferencing Architecture

As Dr. Houh explains, *Basturk's* secure transport protocol is compatible with *Handley's* internet conferencing architecture in part because *Basturk's* “on-the-fly” teaching does not add any latency to the data transmission. EX1033, ¶¶34-40. As described above, PO incorrectly equates *Basturk's* “reliability” feature to reliable protocol features like those found in TCP. POR, 29; EX1033, ¶34. PO's string of unsupported conclusions include that “because Basturk's protocol is a reliable delivery protocol like TCP, it would be unsuitable for transporting Handley's real-time RTP traffic for the same reasons that Handley teaches TCP is unsuitable.” POR, 28-29. As Dr. Houh explains, *Basturk's* secure transport protocol is not a reliable delivery protocol like TCP, nor does it introduce hundreds of milliseconds of delay, as PO implies. POR, 28; EX1033, ¶34. Performing the MAC computation “on-the-fly” does not add latency to the transmission because the MAC computation occurs in parallel to the transmission. EX1033, ¶34.

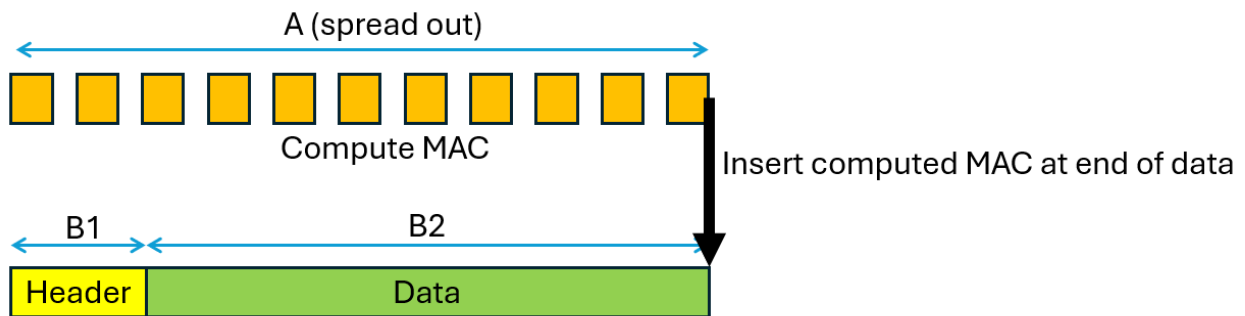
By way of illustration, Dr. Houh compares a serial computation when “on-the-fly” is not used (Houh Figure 4) to a parallel computation when “on-the-fly” is used (Houh Figure 5). EX1033, ¶35. In Houh Figure 4, the MAC is first computed, placed in the appropriate place in the packet, and then the packet is transmitted. *Id.*

For this serial computation (not on-the fly), **A** is the time required to compute the MAC, **B1** is the time required to transmit the packet header, and **B2** is the time required to transmit the packet's data contents. *Id.*



Houh Figure 4

Measured from the time the MAC computation is started, the packet is fully transmitted in $(A+B1+B2)$ seconds. *Id.* Compare this serialized computation in Houh Figure 4 to the parallel computation of the MAC as shown below in Houh Figure 5 when “on-the-fly” is used. *Id.*



Houh Figure 5

Basturk teaches “on-the-fly” computation as illustrated in Houh Figure 5. *Id.*; EX1006, 7-8, 12. In this example of “on-the-fly” computation, while the packet is being transmitted, the computation of the MAC begins and continues in parallel until the computation is finished. EX1033, ¶35. The computed MAC value is inserted at the end of the packet as transmission completes. *Id.* Here, the time

from when the MAC computation is started to when the packet transmission is finished is **(B1+B2)** seconds. *Id.* As seen, *Basturk's* “on-the-fly” MAC computation would not add latency to *Handley's* internet conferencing architecture and would meet *Handley's* delay bounds. EX1005, 4-5. *Id.*

PO further reasons that “because *Basturk's* protocol is a reliable delivery protocol like TCP, it would be unsuitable for use with *Handley* which relies on a multicast network for delivery of packets.” POR, 29. As described above, *Basturk's* secure transport protocol is not a reliable protocol like TCP and PO's arguments are without merit. EX1033, ¶¶29-33.

Finally, PO misinterprets *Basturk's* secure transport protocol, claiming it “similarly uses acknowledgements, and thus would be incompatible with *Handley* due to delays caused by acknowledgment and retransmission.” POR, 30 (citing EX2004, ¶116). But Ground I is based on *Basturk's* secure transport protocol and its “on-the-fly” authentication techniques and *Handley's* internet conferencing architecture. EX1033, ¶38. *Basturk* teaches:

Integrity via signing. If the integrity option is selected then data is *signed* by M_A under the session key α (a signature is also called a MAC, or message authentication code). The signature covers not only the data payload but also the transport header (as explained later). On receiving a signed packet, M_B uses α to verify the signature.

EX1006, 9. A POSITA would not have understood this passage to mean that “[i]ntegrity via signing” only applies to acknowledgment packets. EX1033, ¶38. Instead, a POSITA would have understood that *Basturk*’s plain teaching of “[i]ntegrity via signing” means any data sent is signed under the integrity option, even data that does not require acknowledgment. *Id.*

2. **Independent Claim 1**

i. **[Element 1.1]**

PO relies on *Basturk*’s footnote three to conclude *Basturk* discourages from modifying an existing application layer communication structure to include the features of *Basturk*’s protocol. POR, 31; EX1006, 4, n. 3. PO takes footnote three out of context. EX1033, ¶¶41-42. Footnote three states that “asking the application to duplicate these transport functions is usually poor design in that it belies the layered communication structure, and would also adversely affect performance.” EX1006, 4 n. 3. But the combination Petitioners put forth does not duplicate functions because *Handley* does not disclose the use of a MAC or signatures and any checksums in different protocol layers used in *Handley* are not for the purpose of authentication. EX1033, ¶39.

To contend *Handley* or *Basturk* do not disclose Element [1.1], PO incorrectly argues that “the POSITA would not have understood *Handley*’s mentioning of RTP’s encryption schemes to disclose providing authentication at

the RTP packet level.” POR, 31. As Dr. Houh explains, *Handley* discloses the implementation of authentication at the RTP packet level because using encryption schemes like DES or MD5 to manipulate the RTP packet occurs directly at the RTP packet level. EX1033, ¶41; EX1005, 12. *Handley* teaches the encryption is specified by RTP to encrypt RTP packets and not at any other layer in the stack. *Id.* Thus, *Handley* discloses authentication at the application layer because *Handley* discloses encryption at the application layer, for example, using an encryption scheme such as DES. *Id.* DES is a symmetric key algorithm and if the data is successfully decrypted, then the receiver knows the data was sent by the sender and the receiver authenticated the sender. *Id.*

PO incorrectly argues *Basturk*’s protocol is a transport level protocol below the application layer and does not provide authentication as an application protocol on an application layer. POR, 33-34. First, the combination set forth is not based on the wholesale combination of both *Handley* and *Basturk*. It is the combination of *Basturk*’s “on-the-fly” authentication technique with *Handley*’s internet conferencing architecture. EX1033, ¶38. Second, PO’s own expert testified that in *Basturk*, data is signed at the application level. EX1031, 172:15-16 (“Q What level is the data signed at? A At the application level.”).

ii. Element [1.2]

PO's arguments for Element [1.2] again rely on its fundamental misunderstanding of *Basturk's* "on-the-fly" teaching. As described above in §V.A.1.a, *Basturk* does not refer to computing the style of MAC during the process of retrieving context data, but rather computing the MAC in parallel to data transmission. EX1033, ¶¶11-14, 45-49. As Dr. Houh explains, "on-the-fly" processing requires or motivates placing the signature at the end of the packet in the combination of *Handley* and *Basturk*. EX1033, ¶¶15-21. *Basturk* states "[t]he signature covers not only the data payload but also the transport header." EX1006, 9. A POSITA would have understood that the MAC can only be computed "on-the-fly" over the data payload and transport header as data transmission proceeds when the MAC is placed at the end of a packet. EX1033, ¶15. If the MAC were placed somewhere other than at the end of the packet, the "on-the-fly" MAC computation during data transmission cannot be computed for the data in the packet that is transmitted **after** the fully computed MAC is inserted into the packet. *Id.* Were the MAC located at the beginning or middle of the packet, the "on-the-fly" computed MAC would not be computed over the data transmitted after the MAC is inserted and, thus, cannot accurately reflect the correct MAC computed over the whole packet. *Id.*

Further, the “Implementation Overview” Section of *Basturk* describes placing the MAC at the end of the whole packet. EX1033, ¶16; EX1006, 16. (“A security segment has been added to include information needed for authentication and confidentiality. ***Following data, a trailer has been added that contains the packet signature.***”). A POSITA would have understood and found obvious that the “trailer,” including a signature or MAC, is data added to the end of the packet. EX1033, ¶16.

PO continues its overly strict interpretation of boundaries between layers when it states “data is passed between the layers of a protocol stack in PDUs only their [*sic*] formation is completed.” POR, 36. As discussed, this interpretation is not supported by the ’682 Patent or Dr. Stubblebine’s testimony. See §V.A.1.b, *supra*.

Attempting to disprove motivation to add an RTP packet trailer, PO argues “RTP provides a method for extending the RTP *header* to support additional information.” POR, 38. As Dr. Houh explains, PO ignores that the RTP RFC 1889 specifically states “[n]ote that this header extension is intended only for limited use. Most potential uses of this mechanism would be better done another way, using the methods described in the previous section.” EX1008, 15; EX1033, ¶50. Even if there were a place to add the MAC in the header, a POSITA would have

still been motivated to place the MAC at the end of the packet, not in the header. EX1033, ¶51.

iii. Element [1.3]

PO provides little analysis for Element [1.3] and merely states “Petitioners have not shown that Element [1.3] was disclosed or rendered obvious for the same reasons [as Elements [1.1] and [1.2]]. POR, 39. Petitioners already explained the reasons PO provides incorrect analysis for Elements [1.1] and [1.2] in Sections V.A.2.i-ii and incorporate that analysis here. *See* EX1003, ¶¶52-56.

iv. Element [1.4]

PO does not dispute that *Handley* and/or *Basturk* disclose and/or renders obvious Element [1.4] in its POR. As such, Petitioners’ stand by the Petition. *See* Petition, 24-26; EX1003, ¶54.

3. Independent Claims 25-28

PO points to its “reasons explained above with respect to Elements [1.1]-[1.3]” as analysis for independent claims 25-28. POR, 60. PO is wrong and for the reasons explained above and *Handley* and/or *Basturk* disclose and/or render obvious claims 25-28. *See* §§V.A.2.i-iv, *supra.*; Petition, 36-41; EX1003, ¶55.

4. Dependent Claims 2-21, and 23

PO merely states that “by their dependency on claim 1, claims 2-24 are patentable over *Handley* in combination with *Basturk*.” POR, 40. As described above, and for the reasons laid out in the Petition, *Handley* and/or *Basturk* disclose

and/or render obvious claim 1 and dependent claims 2-21, and 23. *See* §§V.A.2.i-iv, *supra*; Petition, 26-36; EX1003, ¶56.

B. Ground II

1. Independent Claim 1

i. Element [1.1]

PO erroneously asserts that “Petitioners have not shown that PacketCable’s alleged ‘implementing RC4 or adding a MAC’ is provided ‘as an application protocol on an application layer.’” POR, 43. PO’s arguments are without merit. EX1033, ¶57-59. *PacketCable* explicitly discloses that “RTP packets are encrypted directly with RC4, without any additional security layers. An MMH-based MAC...provides message integrity.” EX1007, 45. Thus, as the Board has recognized, “implementing RC4 or adding a MAC as taught by *PacketCable* results in ‘providing . . . authentication at [an RTP] packet level as an application protocol on the application layer.’” Paper 16, 30.

As discussed above in §V.A.1.b, the ’682 Patent itself teaches that RTP is above layer 4 and shows protocols above layer 4 as an “Application protocol” used by an “Application.” EX1001, 6:39-40, FIG. 1; EX1033, ¶57. PO states that “PacketCable indicates to the POSITA that MTA devices generally are responsible for providing secure transport services, including ‘Authentication, confidentiality and integrity of some messages between the MTA and other PacketCable network elements.’” POR, 43-44. Dr. Houh explains that to the extent Dr. Stubblebine

opines that a device performing authentication is different than an application performing authentication, that is incorrect. EX1033, ¶58. The authentication may be for different uses, and the device performing authentication does not preclude applications on the device from performing authentication. *Id.*

ii. Element [1.2]

PO admits that “PacketCable mentions that ‘[e]ach RTP packet may include an optional message authentication code (MAC)’ and that ‘MAC computation spans the packet’s unencrypted header and encrypted payload,’” but argues that “the POSITA would not have understood this to disclose (explicitly or inherently) inserting the optional MAC ‘at end of a whole RTP packet payload.’” POR, 46 (quoting EX1007, 47). As Dr. Houh explains, a POSITA would have only had a finite number of places to place the MAC—the beginning, middle, or end. EX1033, ¶60-75; *see also* Petition, 46-47. PO’s substantial reliance on RTP’s “mechanism for placing additional information in the header” is misguided. POR, 49. As stated above, PO ignores that the RTP RFC 1889 specifically states the header extension is of limited use and most potential uses would be better done another way. EX1008, 15; §V.A.2.ii, *supra*. PO cannot fathom better ways of storing the MAC than in the RTP header extension even when the RTP RFC explicitly urges the reader to do so. EX1033, ¶67; EX1008, 14 (“Additional information that is required for a particular payload format, such as a video

encoding, should be carried in the payload section of the packet.”). Dr. Houh explains there can be only one header extension in an RTP packet header; PO’s technique cannot be used where an RTP header extension already exists. EX1033, ¶68.

PO falsely asserts that “Petitioners identify no negative impacts to ‘performance’” and “do not identify any ‘packet fragmentation issues’” that favor putting data at the end of an RTP packet payload. POR, 47-48. For performance impacts, Dr. Houh explains that by placing a MAC in the middle of the data, the data in the packet would be interrupted and extra data fields would be needed to identify the exact location of the authentication data within the packet. EX1033, ¶61. PO misses or fails to acknowledge that extra data in a packet means more storage is required. *Id.* Extra data in a packet also means the transmission of a packet takes longer because for a fixed transmission rate, transmitting more data requires more time. *Id.* For fragmentation impacts, Dr. Houh explains that placing authentication data in the middle of the packet data causes the data to be separated into two data segments: the segment before the authentication data and the segment after the authentication data. EX1033, ¶62. A POSITA would have known and understood to avoid both this fragmentation and unnecessary data fields. *Id.* Instead, placing the MAC at the end of the payload would allow the transmitter to transmit data while simultaneously computing the checksum. *Id.* This prevents any

unnecessary transmission delay by having to first compute the checksum and then transmit the payload. *Id.*

PO incorrectly states that “RTP can rely on the underlying UDP protocol for providing checksum services, thus the POSITA would not have been motivated to include a trailer at the end of an RTP packet based on the use of checksums in lower layer protocols.” POR, 49 (internal citations omitted). As Dr. Houh explains, the checksum field in a UDP header is a checksum of the data. EX1033, ¶71. While it can be used to verify there are no errors in the received data, it is not used for authentication. *Id.* This issue has no bearing on *PacketCable’s* explicit statement that a MAC can be used in conjunction with RTP packets (EX1007, 45) and that it would have been obvious to put at the end of a packet. *Id.*

PO then puts forth unavailing arguments against putting the MAC at the end of the packet. As described, a POSITA would have understood that placing the MAC at the end of the packet enables transmitting data in parallel to computing the checksum, preventing unnecessary transmission delay from first computing the checksum before transmitting the payload. EX1033, ¶62.

PO states “to the extent Petitioners are suggesting that ‘placing authentication at the end of the whole RTP packet payload’ somehow results in bit-by-bit processing benefits occurring across all layers going down to the physical layer where transmission occurs bit-by-bit; they have not shown this.” POR, 51.

PO either misinterprets or misunderstands this issue. Petitioners describe the benefits of putting the MAC at the end of a packet, but do not describe it as “bit-by-bit processing benefits occurring across all layers going down to the physical layer.” *Id.* Instead, Petitioners describe performance benefits that include reducing the latency to transmit packets. EX1033, ¶74.

iii. Element [1.3]

PO provides no substantive analysis for Element [1.3]. PO is wrong for the reasons explained. *See* §§V.B.1.i-ii, *supra*; EX1033, ¶76.

iv. Element [1.4]

PO does not dispute that *Handley* and/or *Basturk* disclose and/or render obvious Element [1.4] in its POR. As such, Petitioners’ stand by the Petition. *See* Petition, 50; EX1033, ¶77.

2. Independent Claims 25-28

PO cites back to claim 1 and provides no additional analysis for independent claims 25-28. PO is wrong for the reasons explained. *See* §§V.B.1.i-iv, *supra*; EX1033, ¶78.

3. Dependent Claims 22 and 24

PO argues “Petitioners do not explain whether or how the act of switching data ‘between a particular IP flow and a particular PSTN port’ involves ‘inserting... authentication data at end of a whole RTP packet payload.’” POR, 54. Dr. Houh explains that “a gateway terminates the IP data side into the PSTN and

ultimately acts as a switch between a particular IP flow and a particular PSTN port.” EX1033, ¶79. Since the gateway terminates the IP flow, it terminates the RTP streams, and this both generates RTP packets containing data from the PSTN side of the call and receives RTP packets to convert to the PSTN format. *Id.* Thus, the gateway both (1) generating generates RTP packets with the MAC at the end of the packet when converting call data from the PSTN into IP packets, and (2) authenticates received RTP packets by computing the MAC and comparing it to the MAC received at the end of a packet when converting call data from IP packets to the PSTN. *Id.*

4. Dependent Claims 2-21 and 23

PO provides no substantive analysis for dependent claims, 2-21, and 23. PO is wrong for the reasons explained above. *See* §§V.B.1.i-iv, *supra*; EX1033, ¶80.

C. Ground III

1. Motivation to Combine

PO’s arguments regarding the combination of *PacketCable* and *Basturk* are the same as its arguments against the combination of *Handley* and *Basturk*. As stated in the Petition, a POSITA would have understood the systems of *PacketCable* and *Basturk* are compatible or related because both implement MACs for authentication. Petition, 65; EX1033, ¶¶81-82. PO’s argument that a “POSITA would have understood that a MAC based on MMH is incompatible with *Basturk*” is unavailing. POR, 58. PO reasons that “a context retrieval would be required

before processing any portion of the message using the secure MAC based on MMH. Consequently, the delay associated with context retrieval would be unavoidable in *PacketCable*, meaning that ‘on-the-fly’ cannot be achieved.” EX2004, ¶177. But PO’s expert undermines its position. For example, during his cross-examination, Dr. Stubblebine testified for context retrieval “the key is immediately available” so “for the same reasons there isn’t any latency for transmission, there is also no latency for reception.” EX1033, ¶83; EX1031, 148:9-149:13; *see also id.*, 146:11-147:7. Thus, Dr. Stubblebine admits that context is immediately available on both transmission and reception, which directly contradicts PO’s statements that “the delay associated with context retrieval would be unavoidable in *PacketCable*, meaning that ‘on-the-fly’ cannot be achieved.” POR, 58.

2. Independent Claim 1

i. Elements [1.1]-[1.4]

PO provides no substantive analysis for Element [1.1]-[1.4] and cites back to Grounds I-II. POR, 59. PO is wrong for the reasons explained above. See §§V.A-B, *supra*; EX1033, ¶83.

3. Independent Claims 25-28

PO provides no substantive analysis for independent claims 25-28 and cites back to Grounds I-II. POR, 59. PO is wrong for the reasons explained above. *See* §§V.A-B, *supra*; EX1033, ¶84.

4. Dependent Claims 22 and 24

PO provides no substantive analysis for dependent claims 22 and 24 and cites back to Ground II. POR, 60. PO is wrong for the reasons explained above. *See* §V.B.3, *supra*; EX1033, ¶85.

5. Dependent 2-21 and 23

PO provides no substantive analysis for dependent claims 2-21 and 23 and cites back to claim 1. *See* §§V.A-B, *supra*; EX1033, ¶86.

D. Ground IV

PO provides no analysis other than citing back to claim 22 of Grounds II and III. For the same reasons explained above, PO is incorrect and claims 22 and 24 are rendered obvious over *Handley* in combination with *Basturk* and *PacketCable*. *See* §V.B.3, *supra*; EX1033, ¶87.

Date: June 17, 2024

Respectfully submitted,

/s/ W. Todd Baker

W. Todd Baker (No. 45,265)

todd.baker@kirkland.com

Postal and Hand-Delivery Address:

KIRKLAND & ELLIS LLP

1301 Pennsylvania Ave., N.W.

Washington, D.C. 20004

Telephone: (202) 389-5000

Facsimile: (202) 389-5200

Ellisen Shelton Turner (No. 54,503)

ellisen.turner@kirkland.com

Postal and Hand-Delivery Address:

KIRKLAND & ELLIS LLP

2049 Century Park East

Los Angeles, CA 90067

Telephone: (310) 552-4200

Facsimile: (310) 552-5900

Jonathan D. Brit (No. 75,646)

jonathan.brit@kirkland.com

Postal and Hand-Delivery Address:

KIRKLAND & ELLIS LLP

601 Lexington Avenue

New York, NY 10022

Telephone: (212) 390-4209

Facsimile: (212) 446-4900

Ethan Perbohner (No. 78,356)

ethan.perbohner@kirkland.com

Postal and Hand-Delivery Address:

KIRKLAND & ELLIS LLP

333 West Wolf Point Plaza

Chicago, IL 60654

Telephone: (312) 862-2000

Facsimile: (312) 862-2200

Attorneys for Petitioners

CERTIFICATE OF COMPLIANCE

This Petition complies with the type-volume limitations as mandated in 37 C.F.R. § 42.24, totaling 5,594 words. Counsel has relied upon the word count feature provided by Microsoft Word.

/s/ *W. Todd Baker*

(No. 45,265)

CERTIFICATE OF SERVICE

The undersigned hereby certifies that a copy of the foregoing document with accompanying Exhibits 1031 through 1034 as listed in Petitioner's Exhibit List on page iii of this document, was served on June 17, 2024, via electronic mail to counsel of record for the Patent Owner at the following:

Christine E. Lehman (Reg. No. 38,535)
clehman@reichmanjorgensen.com
Philip J. Eklem (Reg. No. 76,490)
peklem@reichmanjorgensen.com
REICHMAN JORGENSEN LEHMAN
& FELDBERG LLP
1909 K Street, NW, Suite 800
Washington, DC 20006

Jaime F. Cardenas-Navia
jcardenas-navia@reichmanjorgensen.com
Michael Matulewicz-Crowley (Reg. No. 72,220)
mmatulewicz-crowley@reichmanjorgensen.com
REICHMAN JORGENSEN LEHMAN & FELDBERG LLP
400 Madison Avenue, Suite 14D
New York, NY 10017

RJ_VideoLabs-Facebook@reichmanjorgensen.com

/s/ W. Todd Baker
(No. 45,265)