

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

**PETITION FOR *INTER PARTES* REVIEW
UNDER 35 U.S.C. § 312 AND 37 C.F.R. § 42.104**

TABLE OF CONTENTS

I.	INTRODUCTION	1
II.	MANDATORY NOTICES	2
	A. Real Party-In-Interest	2
	B. Related Matters.....	2
	C. Counsel and Service Information.....	3
	D. 37 C.F.R. §42.8(b)(4): Service Information.....	3
III.	PAYMENT OF FEES	3
IV.	CERTIFICATION	4
V.	OVERVIEW OF CHALLENGE AND RELIEF REQUESTED.....	4
	A. Prior Art.....	4
	B. Relief Requested.....	5
VI.	NO DISCRETIONARY DENIAL	5
	A. No Prior Petition.....	5
	B. The Presented Grounds Are Dissimilar to the Previous Art and Arguments	6
	C. Efficiency, Fairness, and the Merits Support Institution	6
VII.	THE '682 PATENT	9
VIII.	PERSON OF ORDINARY SKILL IN THE ART	9
IX.	CLAIM CONSTRUCTION	9
X.	SPECIFIC GROUNDS FOR PETITION	10
	A. Ground I: Claims 1-21, 23, and 25-28 Are Rendered Obvious over <i>Handley</i> in Combination with <i>Basturk</i>	10

B.	Ground II: Claims 1-28 Are Rendered Obvious Over <i>PacketCable</i> in Combination With <i>Handley</i>	41
C.	Ground III: Claims 1-28 Are Rendered Obvious Over <i>PacketCable</i> in Combination with <i>Basturk</i>	64
D.	Ground IV: Claims 22 and 24 Are Rendered Obvious Over <i>Handley</i> in Combination with <i>Basturk</i> and <i>PacketCable</i>	72
XI.	SECONDARY CONSIDERATIONS	73
XII.	CONCLUSION.....	74

TABLE OF AUTHORITIES

	Page(s)
CASES	
<i>Apple Inc. v. Aire Technology Limited</i> , No. IPR2022-01136, Paper 11 (P.T.A.B. January 4, 2023)	7
<i>Google LLC v. EcoFactor, Inc.</i> , IPR2021-01578, Paper 9 (P.T.A.B. Mar. 18, 2022)	8
<i>KSR Int'l Co. v. Teleflex Inc.</i> , 550 U.S. 398 (2007)	47
<i>Sand Revolution II, LLC v. Continental Intermodal Group Trucking LLC</i> , No. IPR2019-01393, Paper 24 (P.T.A.B. June 16, 2020)	7, 8, 9
<i>VideoLabs Inc et al v. Amazon.com Inc. et al</i> , No. 6-22-cv-01167 (W.D. Tex.)	2
<i>VideoLabs, Inc., et al. v. Meta Platforms, Inc. et al.</i> , Civil Action No. 1-22-cv-00680 (D. Del.)	2
<i>Wyer v. Master Lock Co.</i> , 616 F.3d 1231 (Fed. Cir. 2010)	74
STATUTES	
35 U.S.C. §102	4, 5
35 U.S.C. §314	5, 7, 8, 9
35 U.S.C. §315	4
35 U.S.C. §325	6
OTHER AUTHORITIES	
37 C.F.R. §42	3, 5

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 .
-------------	--

Unless otherwise noted, all emphasis added.

TABLE OF ABBREVIATIONS

Term	Abbreviation
Person of Ordinary Skill In The Art	POSITA
Real-time Transport Protocol	RTP
Protocol Data Unit	PDU
Open Systems Interconnection Model	OSI Model
Message Authentication Code	MAC
Voice over Internet Protocol	VoIP
Internet Protocol	IP
Message-Digest Algorithm	MD5
Data Encryption Standard	DES
Secure Hash Algorithm 1	SHA-1
Public Switched Telephone Network	PSTN

I. INTRODUCTION

Meta Platforms, Inc, Instagram, Inc., WhatsApp LLC, Meta Platforms Technologies, LLC, and GIPHY, Inc. (“Petitioners”) requests *inter partes* review of claims 1-28 (“Challenged Claims”) of U.S. Patent No. 7,266,682 (EX1001, the “’682 Patent”).

The ’682 Patent is directed to data transmission and specifically, a method and system for transmitting data from a transmitter to a receiver. The Challenged Claims purport to address a technique for transmitting data from a transmitter to a receiver with the goal of ensuring the data is not unwanted. The methods and systems claimed are based on what the ’682 Patent confirms are known and conventional data transmission protocols and authentication techniques. The ’682 Patent operates at a Real Time Transport Protocol (RTP) packet level as an application protocol on an application layer. It purports to provide authentication from a transmitter to a receiver by inserting at the transmitter authentication data at the end of a whole RTP packet payload. Based on the RTP packet level authentication, a receiver allegedly accepts the whole RTP packet payload if the receiver knows the transmitter based on the RTP packet level authentication data. At base, the receiver determines whether it knows the transmitter, and if the receiver does not know the transmitter, the data is rejected. As described herein, the ’682 Patent’s techniques were known in the art (as admitted by the Patent

itself), the prior art applied in this Petition was not in front of the Examiner, and the Challenged Claims should be cancelled as obvious and unpatentable.

II. MANDATORY NOTICES

A. Real Party-In-Interest

Meta Platforms, Inc., Instagram, Inc., WhatsApp LLC, Meta Platforms Technologies, LLC, and GIPHY, Inc. are real parties-in-interest.

B. Related Matters

The '682 Patent is asserted against Petitioners in litigation in the District of Delaware, *VideoLabs, Inc., et al. v. Meta Platforms, Inc. et al.*, Civil Action No. 1-22-cv-00680 (D. Del.), filed May 24, 2022 (the “Delaware Action”). The Delaware Action may affect, or be affected by, decisions in this proceeding. Nothing herein should be taken as an endorsement of VideoLabs’ positions in the co-pending litigation. The '682 Patent was previously asserted by Patent Owner in now completed litigation: *VideoLabs Inc et al v. Amazon.com Inc. et al*, No. 6-22-cv-01167 (W.D. Tex.).

C. Counsel and Service Information

Lead Counsel	Back-Up Counsel
W. Todd Baker (No. 45,265) todd.baker@kirkland.com Postal and Hand-Delivery Address: KIRKLAND & ELLIS LLP 1301 Pennsylvania Avenue, N.W. Washington, D.C. 20004 Telephone: (202) 389-3135 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

D. 37 C.F.R. §42.8(b)(4): Service Information

Petitioners concurrently submit a Power of Attorney, 37 C.F.R. §42.10(b), and consents to electronic service to Meta_VideoLabs@kirkland.com.

III. PAYMENT OF FEES

Review of 28 claims is requested. The undersigned authorizes the Office to charge to Deposit Account No. 506092 the 37 C.F.R. §42.15(a)(1) fee and any additional fees due for this Petition.

IV. CERTIFICATION

Petitioners certify under Rule 42.104(a) that the '682 Patent is available for IPR and that the Petitioners are not barred or estopped from requesting IPR of the Challenged Claims on the grounds herein. Petitioners certify: (1) Petitioners do not own the '682 Patent; (2) Petitioners (or any real party-in-interest) have not filed a civil action challenging the validity of any the '682 Patent claims; (3) this Petition is being filed within one year of service of a complaint against Petitioners asserting the '682 Patent in the Delaware Action; (4) estoppel provisions of 35 U.S.C. §315(e)(1) do not prohibit this IPR; and (5) this Petition is filed after the '682 Patent was granted.

V. OVERVIEW OF CHALLENGE AND RELIEF REQUESTED

A. Prior Art

The earliest potential effective filing date of the '682 Patent's claims is January 18, 2000.¹ Petitioners' challenge is based on the following prior art, for which a detailed overview is provided in the Declaration of Dr. Houh, which is incorporated by reference. EX1003, Section VI.

1. **Handley (EX1005)** – *Very Large Conferences on the Internet: The Internet Multimedia Conferencing Architecture and the MBONE*, published February 11, 1999, in the Computer Networks journal. Handley is at least pre-AIA 35 U.S.C. §§102(a) and 102(b) prior art.

¹ Petitioners reserve the right to challenge the January 18, 2000, priority date or any evidence of prior invention.

2. **Basturk (EX1006)** – *Secure Transport Protocols for High-Speed Networks* was publicly available by at least September 1, 1995. See EX1010. Basturk is at least pre-AIA 35 U.S.C. §§102(a) and 102(b) prior art.
3. **PacketCable (EX1007)** – *PacketCable 1.0 Architecture Framework Technical Report*, published December 1, 1999. PacketCable is at least pre-AIA 35 U.S.C. §§102(a) and 102(b) prior art.

No prior-art reference identified was substantively considered by the Examiner during prosecution or cumulative of any art considered by the Examiner.

B. Relief Requested

Petitioners request cancellation of the Challenged Claims as follows:

Ground	Challenged Claims	Proposed Statutory Rejection
I	1-21, 23, 25-28	Obvious under § 103 in view of Handley in combination with Basturk
II	1-28	Obvious under § 103 in view of PacketCable in combination with Handley
III	1-28	Obvious under § 103 in view of PacketCable in combination with Basturk
IV	22, 24	Obvious under § 103 in view of Handley in combination with Basturk and PacketCable

VI. NO DISCRETIONARY DENIAL

A. No Prior Petition

Neither Petitioners nor anyone else has previously filed any IPR or PGR petitions against the '682 Patent. Thus, there is no basis for the Board to exercise its discretion under 35 U.S.C. §314(a) and 37 C.F.R. §42.108(a).

B. The Presented Grounds Are Dissimilar to the Previous Art and Arguments

The prior art identified in the four Grounds of this Petition have not been previously applied against the Challenged Claims, and all factors considered under 35 U.S.C. §325(d) favor institution. Grounds I-IV in this Petition apply various combinations of *Handley*, *Basturk*, and *PacketCable*, which are not cumulative and are substantially different than any previously considered prior art. Further *Handley*, *Basturk*, and *PacketCable* were not applied against the Challenged Claims or discussed during prosecution of the '682 Patent. *See generally* EX1002. None of the references relied upon herein are included in any other IPR or PGR proceeding related to the Challenged Claims.

Additionally, for Grounds I-IV, this Petition cites RFC 1889 (EX1008), which was previously cited in a related IDS, and Tanenbaum (EX1009), a previous version of which was cited in a related IDS. The Petition's citation to RFC 1889 simply demonstrates the technical intricacies of the RTP protocol upon which the '682 Patent relates. Similarly, Tanenbaum is utilized as a background/teaching reference to demonstrate the knowledge of a POSITA at the time of the '682 Patent.

C. Efficiency, Fairness, and the Merits Support Institution

Pursuant to the Interim Procedure for Discretionary Denials in AIA Post-Grant Proceedings with Parallel District Court Litigation (June 21, 2022) ("Vidal

Memo”), the Board should not exercise its discretion under §314(a) considering the Delaware Action, since the *Fintiv* factors support institution. Further, the petition presents compelling evidence of patentability.

Factor 1: Institution will enable the Board to resolve patentability and if the Challenged Claims are deemed unpatentable, it would relieve the Court of the need to continue with that issue.

Factor 2: The Board will very likely issue its Final Written Decision before a trial occurs in the Delaware Action, where no answer has been filed, no discovery has taken place, no claim construction proceedings have occurred, and no scheduling order or trial date have been set.

Factor 3: All major case milestones in the Delaware Action are yet to occur, and deadlines for them have not been set, such that substantial work remains in the Delaware Action. Indeed, the Delaware Action remains in its infancy at the pleading stage.

Factor 4: Petitioners stipulate in the Delaware Action that, if this IPR is instituted, Petitioners will not pursue invalidity of the Challenged Claims on the actual grounds raised in this IPR. Although not as broad as a *Sotera* type stipulation, the Board has concluded—subsequent to the Vidal Memo—that a *Sand Revolution* type stipulation effectively addresses the risk of duplicative efforts. *Apple Inc. v. Aire Technology Limited*, No. IPR2022-01136, Paper 11 (P.T.A.B.

January 4, 2023) (citing *Sand Revolution II, LLC v. Continental Intermodal Group Trucking LLC*, No. IPR2019-01393, Paper 24 at 11–12 (P.T.A.B. June 16, 2020) (informative)).

In addition, claims 1-28, are challenged in this Petition, but only claim 1 is asserted in the Delaware Action.² The Board should address the validity of those additional claims for the public’s benefit. *See Google LLC v. EcoFactor, Inc.*, IPR2021-01578, Paper 9 (P.T.A.B. Mar. 18, 2022).

Factor 5: The parties have no similarity, but in view of the other *Fintiv* factors—which heavily weigh against the Board’s exercise of §314(a) discretion—the parties’ similarity is of limited weight.

Factor 6: This Petition’s merits are strong, which weighs against the Board exercising its discretion under §314(a). *Sand Revolution II, LLC*, IPR2019-01393, Paper 24 at 13 (P.T.A.B. June 16, 2020). Further, the petition presents compelling evidence of unpatentability such that “the evidence, if unrebutted in trial, would plainly lead to a conclusion that one or more claims are unpatentable by a preponderance of the evidence.” Vidal Memo at 4. Accordingly, the Board should not deny institution under *Fintiv*.

² The Complaint alleges the “Accused Products” infringe “by using methods and/or taking steps that practice at least claim 1 of the ’682 Patent.” EX1012, ¶231.

“Considering the *Fintiv* factors as part of a holistic analysis,” it would run counter to “the interests of the efficiency and integrity of the system” if this Board were “to deny institution of a potentially meritorious Petition.” *Sand Revolution II, LLC*, IPR2019-01393, Paper 24 at 13. Thus, the Board should decline to exercise its discretion under § 314(a).

VII. THE '682 PATENT

An overview of the '682 Patent, including a summary of the technology, claims, and prosecution history is provided in the Declaration of Dr. Houh. EX1003, Section IV.

VIII. Person of Ordinary Skill in the Art

A POSITA at the time of the '682 Patent would have had a Bachelor's degree in computer science, electrical engineering, or a related field, and two years of experience; or by obtaining a Master's degree in electrical engineering or a related field, but having only one to two years of experience; or by having no formal education but experience in network data transmission and computer networking of at least eight years. Additional education might compensate for less experience and vice-versa. EX1003, ¶45.

IX. CLAIM CONSTRUCTION

For purposes of this petition only, Petitioners present no terms for construction and further submits that the Board does not need to construe any claim term for purposes of evaluating the prior art in this Petition.

X. SPECIFIC GROUNDS FOR PETITION

When addressing a dependent claim, the analysis from any independent claim that it depends on is incorporated by reference.

A. Ground I: Claims 1-21, 23, and 25-28 Are Rendered Obvious over *Handley* in Combination with *Basturk*

1. Motivation to Combine

Basturk's disclosure of secure transport protocols and its known communication benefits supplements *Handley's* disclosure of internet conferencing architecture. EX1003, ¶63. *Basturk*, which shares the same field of internet data transmission as *Handley*, further informs a POSITA regarding implementing authentication over the internet. EX1003, ¶63 As described below, and as Dr. Houh explains, a POSITA would have been motivated to modify *Handley* with *Basturk* for many reasons. See EX1003, ¶¶63-70.

Although *Handley* does not expressly disclose “inserting, at the transmitter, authentication data at end of a whole RTP packet payload,” combining *Handley's* disclosure of internet conferencing architecture in combination with *Basturk's* teaching of secure transport protocols would have been obvious to a POSITA. EX1001, cl. 1; EX1003, ¶64. A POSITA would have been motivated to use *Basturk's* “on-the-fly” signature computation with *Handley's* internet conferencing architecture due to the known communication benefits, which include faster data transmission, reduced jitter, and reductions in computing power. EX1003, ¶64.

For example, *Handley* recognized that receivers need to efficiently and securely process data and discloses that “[r]eal-time applications such as audio and video need to be able to buffer real-time data at the receiver for sufficient time to remove the jitter added by the network.” EX1005, 8. Similarly, *Basturk* discloses, “[i]deally, we want signatures to be computed and verified on-the-fly; i.e., MAC generation should be carried out as data transmissions proceeds rather than before these operations have completed” because “[t]he goal is to avoid slowing down the regular flow of data.” EX1006, 12 (cleaned up).

A POSITA would have appreciated that although *Basturk* does not explicitly disclose RTP, RTP’s secure transport protocol is compatible with *Handley*’s internet conferencing architecture. EX1003, ¶66. For example, *Basturk* discloses that its proposed protocol “should be able to operate in both environments,” including “software encryption and decryption for secure video conferences” at both low- and high-resolution environments. EX1006, 5. A POSITA would have understood that RTP is commonly used for video conferences. EX1003, ¶66. Further, *Basturk* teaches its security features “reflect[] the need to add security functionality in a way which is consistent with, and best matches, the existing communication structure” and so that “[a]pplications using the regular transport protocol [will] not have to be modified at all.” EX1006, 3, 5.

A POSITA would have understood that the *Handley* and *Basturk* systems are compatible and related because *Handley* describes a scenario in which authentication is needed in a multicast environment.

This is nicely illustrated by the multicast of the Rolling Stones concert in 1994 which was billed as being the first live concert on the Mbone. In fact, this honour goes to a little known band called Severe Tire Damage who had multicast an impromptu concert a year previously. To make their point, just before the Stones were due to go on stage, Severe Tire Damage suddenly (sic) started broadcasting one of their songs live to the same multicast group. ***Clearly commercial broadcasters want to avoid occurrences like this one.***

See EX1005, 13.

Basturk describes a similar scenario in which “[t]wo parties, denoted *A* and *B*, communicate over a network. We will naturally view the parties as applications who use a common transport protocol. An adversary *E* monitors the parties’ communications, and is even capable of injecting or removing messages from the lines.” EX1006, 3. Both *Handley* and *Basturk* aim to remove the possibility of a third-party interfering with communication between a defined set of parties. Thus, a POSITA would have understood and found it obvious to implement an authentication solution for methods of secured data transmission presented in *Basturk* to the same problem presented in *Handley*. EX1003, ¶68. Specifically, a POSITA would have understood and found obvious that *Basturk*’s authentication

methods would have prevented unauthenticated interruption of video transmissions. EX1003, ¶68.

Using *Handley's* internet conferencing architecture and *Basturk's* secure transport protocol, a POSITA would have understood that hardware to implement multi-media transport may be necessary to keep up with the high volume of packets generated in video applications and could include dedicated cryptographic hardware as taught by *Basturk*. EX1003, ¶69.

Thus, use of *Handley's* internet conferencing architecture in conjunction with *Basturk's* secure transport protocol would have made use of known multi-media data transport architecture and a known authentication method to improve data transmission security. EX1003, ¶70. The combination also applies a known technique to a known device or method ready for improvement to yield predictable results. EX1003, ¶70. Using an internet conferencing architecture in combination with a secure transport protocol involves the use of routine computers, equipment, and techniques, and a POSITA would have had a reasonable expectation of success in using “on-the-fly” data authentication with *Handley*. A POSITA would have found the use and implementation of such hardware to implement *Basturk's* authentication methods with the internet conferencing architecture obvious. EX1003, ¶70.

2. Independent Claim 1

a) **Preamble [1]**

As discussed in Section VII, the '682 Patent concedes that data transmission (sending of data) was known in the art. Regardless, and to the extent it is a limitation, *Handley* disclosed the preamble of Claim 1 or at least rendered it obvious. EX1003, ¶¶71-73.

Handley teaches that “multimedia information must be transmitted in packets” and that traditionally, “the internet has provided so-called *best-effort* delivery of datagram traffic from senders to receivers.” EX1005, 2, 4. Specifically, *Handley* discloses synchronous or “real-time” conferencing “including audio, video, shared whiteboards and other applications.” *Id.*, 1. As Dr. Houh explains, a POSITA would have understood that the multimedia described in *Handley* can be transmitted over the internet in the form of packets and that transmitting data over the internet requires a transmitter and receiver. EX1003, ¶72. Further, a POSITA would have understood that a “sender” in the context of *Handley* is equivalent to a transmitter because those terms are used interchangeably when describing multimedia conferencing networks. *Id.*, ¶72. Thus, *Handley* discloses “a method for transmitting data from a transmitter to a receiver” because multimedia information, *i.e.*, data, is transmitted by “senders to receivers” in packets over the internet. EX1001, cl. 1; EX1005, 4; EX1003, ¶72.

The preamble is further obvious in view of this combination as *Basturk* discloses that “[t]wo parties, denoted *A* and *B*, communicate over a network...who use a common transport protocol.” EX1006, 3; EX1003, ¶73. A POSITA would have understood that networks support protocol hierarchies that are used to transmit data and that networks contain transmitters and receivers. EX1003, ¶73. Further, a POSITA would have understood that parties *A* and *B* in *Handley* may be transmitters or receivers because they both “communicate over a network” with one another EX1003, ¶73; EX1006, 3. Thus, *Handley* in combination with *Basturk* discloses “a method for transmitting data from a transmitter to a receiver.” EX1003, ¶73.

b) **Element [1.1]**

See Ground I, Preamble [1]; EX1003, ¶74.

First, as discussed in Section VII, the '682 Patent admits that each claimed technology including RTP, authentication, and the OSI Model (which includes an application layer and application protocol), were known in the art. EX1003, ¶75. Second, despite said admissions, *Handley* disclosed this element or at least rendered it obvious. *Id.*, ¶¶74-82.

Beginning with RTP, *Handley* teaches RTP as a “transport protocol for real-time flows” that “provides a standard format packet header which gives media specific timestamp data, as well as payload format information and sequence

numbering amongst other things.” EX1005, 8. According to the original RTP specification, RFC 1889, “RTP is intended to be malleable to provide the information required by a particular application and will often be integrated into the application processing rather than being implemented as a separate layer.” EX1008, 4. As Dr. Houh explains, a POSITA would have understood RFC 1889 to mean RTP packets are created at the application layer and use an application protocol because RTP is integrated into the application processing and the media application generating the audio or video stream is creating the RTP packets. EX1003, ¶76. As such, a POSITA would have understood that the media application generating the audio or video stream is creating the RTP packets at the application layer. *Id.*, ¶76.

Next, RTP utilizes the known OSI Model and as described in Section VII, RFC 1889 describes RTP as “represent[ing] a new style of protocol following the principles of application level framing and integrated layer processing.” EX1003, ¶77; EX1008, 4. A POSITA would have understood and found obvious that because RTP follows the principles of application-level framing and integrated layer processing, that the RTP packets are created at the application layer and RTP is an application protocol. EX1003, ¶77. Thus, *Handley* teaches “Real Time Transport Protocol (RTP) packet level as an application protocol on an application layer” due to its disclosure of RTP. EX1003, ¶77; EX1001, cl. 1.

Handley discloses “transmitter-to-receiver authentication.” EX1003, ¶78. *Handley* discloses that “security measures may be required to actually enforce the conference policy, *e.g.*, to control who is listening and to *authenticate* contributions as actually originating from a specific person.” EX1005, 2; EX1003, ¶78. Further, *Handley* discloses that if an application requirement, *e.g.*, conference policy, “is to communicate between some defined set of users” then “strict privacy can [] be enforced...through adequate end-to-end encryption.” EX1005, 12. *Handley* also discloses multiple types of transmitter-to-receiver authentication techniques that take place at the RTP packet level. EX1003, ¶78. As described in Ground I, Element [1.1], *Handley* teaches RTP encryption using DES³ and MD5⁴. EX1005, 12; EX1003, ¶78. A POSITA would have understood that *Handley* discloses the implementation of authentication at the RTP packet level because using encryption schemes like DES or MD5 to manipulate the RTP packet directly occurs at the RTP packet level. EX1003, ¶78.

To the extent *Handley* does not expressly disclose Element [1.1], it would have been obvious in view of *Handley* in combination with *Basturk. Id.*, ¶79.

³ DES is a symmetric-key block cipher encryption algorithm used to secure electronic data by converting plaintext to ciphertext. EX1003, ¶78.

⁴ MD5 is a cryptographic hash function that generates a fixed-size output from an input message of any size, used for verifying the integrity of the data and detecting any changes or tampering with the original message. EX1003, ¶78.

As described in Section X.A.1, a POSITA would have found it obvious to implement *Basturk*'s secure transport protocol and its authentication techniques to *Handley*'s internet conferencing architecture. *Id.*, ¶80.

Regarding said authentication, *Basturk* discloses:

[S]ecure transport protocol has two phases, the initialization phase and the steady state. In the former, an entity authentication and key exchange protocol is executed. This authenticates the parties and yields them a session key α . In the steady state, if the integrity option is selected, both data integrity and secure delivery are provided by attaching to **each packet a signature**, computed under key α , and taken over some portions of the header and the data payload.

EX1006, 5.

Further, *Basturk* teaches that when machine A (“M_A”) and machine B (“M_B”) wish to send and receive data with the integrity option selected, then “data is **signed** by M_A under 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” and “[o]n receiving a signed packet, M_B uses α to verify the signature.” EX1006, 9.

Additionally, *Basturk* discloses that “[a]pplication level security features are easily (and usually) provided by **application-level security**. Here the sender applies appropriate cryptographic functions to his data **before** passing it to the transport for conveyance” and that “[i]ntegrity is usually guaranteed by **signing** the data before

passing it to the transport level.” *Id.*, 3. A POSITA would have understood that key elements of *Basturk*’s authentication technique take place as an application protocol on an application layer because even in “transport level security” described in *Basturk*, e.g., cryptographic functions such as key exchange and selection of a key to use for a communication, occur at the application layer, since such keys are passed from the application layer to the transport layer. EX1006, 4-5; EX1003, ¶82.

c) **Element [1.2]**

See Ground I, Preamble [1], Element [1.1]; EX1003, ¶83.

Element [1.2] would have been obvious in view of *Handley* in combination with *Basturk*. *Id.*, ¶¶83-85. *Basturk* discloses that, for performance, it is “essential that the steady state overhead involved in computing and checking signatures and counters be kept to a minimum.” EX1006, 12. To keep computing to a minimum, which is a necessity in multi-media transport for delay and buffering purposes, *Basturk* discloses “on-the-fly” signature verification as described in X.A.1. EX1006, 12; EX1003, ¶84. *Basturk* further explains that

[c]onfidentiality means user data should not be readable by the adversary. The secure transport protocol will encrypt user data, under the session key, and use the ciphertexts as the actual payload. As encryption is done prior to packetization, the signature (if present) of an outgoing packet is computed after data encryption and the signature of an incoming packet is computed before decryption. This

complies with the requirement of a signature computed ‘on the fly’. If the signature checking at reception fails, it means the the [sic] packet is corrupted, and the packet is simply dropped: decryption is avoided.”

EX1006, 12.

To illustrate “on-the-fly” signature implementation, *Basturk* discloses that “[a] security segment has been added to include information needed for authentication and confidentiality” and that “[f]ollowing data, a trailer has been added that contains the packet signature.” *Id.*, 16. A POSITA would have understood and found obvious that a “trailer” is data that can be added at the end of the packet, much like the header is data added at the beginning of a packet, and that a trailer may be defined to contain authentication data, such as the packet signature. EX1003, ¶85. Although *Basturk* does not explicitly disclose RTP, a POSITA would have understood and found obvious that when combined with *Handley’s* RTP internet conferencing architecture, the trailer, *i.e.*, authentication data or MAC from *Basturk*, would be placed at the end of the whole RTP packet payload to enable “on-the-fly” signature computation and verification, allowing the transmitter to transmit data while simultaneously computing the authentication data. EX1005, 12; EX1006, 12; EX1003, ¶85.

d) **Element [1.3]**

See Ground I, Preamble [1], Elements [1.1], [1.2]; EX1003, ¶86.

Handley disclosed this element or at least rendered it obvious. *Id.*, ¶¶86-91. As Dr. Houh explains, a POSITA would have understood and found obvious that a receiver would ascertain whether it “knows” if the transmitter is using either symmetric or asymmetric key encryption and, further, for symmetric key encryption, the transmitter and receiver exchange a secret key prior to data transmission for data encryption and decryption. *Id.*, ¶87. For symmetric encryption, *Handley* discloses that “RTP specifies a standard way to encrypt RTP and RTCP packets using private key encryption schemes such as DES.” EX1005, 12. As described in Ground I, Element [1.1], a POSITA would have understood that RTP encryption occurs at the packet level because DES and other encryption schemes encrypt the RTP packets directly. EX1003, ¶87. A POSITA would have further understood that DES is a symmetric-key algorithm, meaning both the transmitter and the receiver must know and use the same secret key to both encrypt and decrypt the data. *Id.*, ¶87. A POSITA would have understood and found obvious that if the receiver knows the transmitter’s secret key for the sessions, then the receiver knows the transmitter’s identity because the transmitter and receiver would have previously established a secret key before they could encrypt and decrypt the data. *Id.*, ¶87.

A POSITA would have understood asymmetric encryption to use public and private keys, whereby the transmitter encrypts the data using the public key of the

recipient, such that only the recipient can decrypt the data using its private key. *Id.*, ¶88. *Handley* discloses that “secured broadcasts can be performed by encrypting a hash (digitally signing) of each packet with the senders private key of a public-private key pair. The public key is then given to the receivers, and they discard (and prune if possible) any packets that are unsigned.” EX1005, 13. As Dr. Houh explains, in this asymmetric encryption digital signature scenario, a POSITA would have understood and found obvious that if the receiver already has the transmitter’s public key stored, e.g., in its key cache, then the receiver knows who the transmitter is because the receiver would have stored the transmitter’s public key to avoid the overhead of retrieving the public key every time a secure communication is initiated. EX1003, ¶88.

In both symmetric and asymmetric encryption implementations, a POSITA would have understood and found obvious that the receiver is ascertaining, based on RTP-packet-level authentication data, whether it knows the transmitter because it either has a shared key to verify the RTP level authentication data in the symmetric implementation or, in the asymmetric implementation, has stored the transmitter’s public key in its key cache to check the authentication data. *Id.*, ¶89.

To the extent *Handley* does not expressly disclose this element, this element would have been obvious in view of *Handley* in combination with *Basturk*. *Id.*, ¶90. *Basturk* discloses a symmetric key implementation, whereby “parties wish to

be able to communicate ‘securely’” and use a “shared secret key a (e.g., a DES key)” to do so. EX1006, 3. Further, *Basturk* discloses that for a message authentication code under session key “ α ”, denoted as “ MAC_α ,” a transmitter “attaches to each packet p the value of $\text{MAC}_\alpha(\bar{p})$ where $\bar{p}$ is the subset of p which includes the transport header, the full data payload, and the trailer of the packet p ” and that “the transport header contains information which identifies the sender so that the identity of the sender is covered by the MAC as well.” *Id.*, 12. A POSITA would have understood and found obvious that the MAC described in *Basturk*, when applied to *Handley’s* RTP internet conferencing architecture, would be generated from the RTP packet because the MAC is generated from $\bar{p}$ “which includes the transport header, the full data payload, and the trailer of the packet p ” and, therefore, it would have been obvious that packet p would be an RTP packet. EX1003, ¶90. Further, a POSITA would have understood and found obvious the MAC would be based on RTP packet level authentication data because, as described in Ground I, Elements [1.1], [1.2], the MAC is generated from the RTP packet data and is inserted at the end of the whole RTP packet payload. EX1003, ¶90.

As described, a receiver knows a transmitter if the two share a secret key, or if the receiver has the transmitter’s public key stored. *Id.*, ¶91 *Basturk* discloses that “data is *signed* by M_A under the session key α (a signature is also called a

MAC)” and “on receiving a signed packet, M_B uses α to verify the signature.” EX1006, 9. *Basturk* describes this as “[o]n receiving p , τ the receiver checks that $\tau = \text{MAC}_\alpha(\bar{p})$ ” whereby p is the packet and τ is the MAC signature. *Id.*, 12. As such, a POSITA would have understood and found obvious said teaching to mean that the receiver uses the session key α (e.g., a symmetric DES key) to verify the MAC signature (i.e., compute a MAC based off the data received and use the same session key α to see if the computed MAC matches the received MAC) which, as described, is generated based of RTP packets and the session key α , making it RTP packet level authentication data. *Basturk* clarifies that said teaching is not limited to symmetric encryption but states that “[f]or simplicity we stick to this symmetric case. Our methods can be modified to work in the public key scenario as well.” EX1006, 3 n.2.

e) **Element [1.4]**

See Ground I, Preamble [1], Elements [1.1], [1.2], [1.3]; EX1003, ¶92.

Handley disclosed this element or at least rendered it obvious. *Id.*, ¶¶92-94. As discussed in Element [1.3], a POSITA would have understood and found obvious that, in using symmetric encryption, a receiver knows a transmitter based on the shared secret key and, in using asymmetric encryption, a receiver knows the transmitter if the receiver has stored the transmitter’s public key. *Id.*, ¶93. *Handley* discloses that, in the asymmetric encryption use case, “[t]he public key is then

given to the receivers, and they **discard** (and prune if possible) any packets that are unsigned.” EX1005, 13. A POSITA would have understood and found obvious that if the receiver discards unsigned packets, or if the signature does not match, then the receiver does not know the transmitter and the whole RTP packet payload is rejected. EX1003, ¶93. Further, a POSITA would have understood and found obvious that if the packets are not “discard[ed],” then the receiver is “accepting” the whole RTP packet payload because the packets are then decrypted and processed. EX1003, ¶93; EX1005, 13.

To the extent *Handley* does not explicitly disclose this element, this element would have been obvious in view of *Handley* in combination with *Basturk*. EX1003, ¶94. As described in Ground I, Elements [1.1], [1.3], *Basturk* discloses the MAC is computed using a DES session key α in addition to the packet’s contents. EX1006, 12. *Basturk* further discloses that “[o]n receiving p , τ the receiver checks that $\tau = \text{MAC}_{\alpha}(p)$ and **rejects** the packet unless this is true,” EX1006, 12, where p is the packet and τ is the MAC signature. EX1003, ¶94. This signature “prevents the adversary from making the receiver accept packets which were not in fact produced by the sender.” EX1006, 12. Further, *Basturk* discloses “[i]f the signature checking at reception fails, it means the the [sic] packet is corrupted, and the packet is simply dropped: decryption is avoided.” *Id.*, 14. A POSITA would have understood and found obvious that when combining

Basturk's authentication techniques with *Handley's* RTP internet conferencing architecture, that the receiver “rejects” the whole RTP packet payload because it would not accept a packet from a transmitter it does not know, or a packet that failed the authentication. EX1003, ¶¶94; EX1005, 11-12.

3. **Dependent Claim 2**

See Ground I, Claim 1. EX1003, ¶¶95-98.

Handley disclosed this Claim or at least rendered it obvious. *Id.*, ¶96 As Dr. Houh explains, a POSITA would have understood and found obvious that a PDU is a single unit of information at a particular OSI layer (*e.g.*, an RTP packet at the application layer). *Id.*, ¶96. As described in Ground I, Element [1.3] *Handley* discloses “encrypting a hash” and “digitally signing” each packet. EX1005, 13; EX1003, ¶96. A POSITA would appreciate that the PDU (an RTP packet) is linked to a secret in two possible ways. EX1003, ¶96. First, the RTP packet may be encrypted, and is linked to the secret via the keys used to encrypt and decrypt the RTP packet. *Id.*, ¶96. Second, the RTP packet may be used as the basis of an authentication code, *i.e.*, digital signature, and is linked to the secret when the hash is encrypted. *Id.*, ¶96. *Handley*, for example, states, “a conference policy that restricts conference membership can be implemented using encryption and restricted distribution of encryption keys.” EX1005, 9 n.2. Further, *Handley* discloses, if “the application requirement (conference policy) is to communicate

between some defined set of users, then strict privacy can only be enforced in any case through adequate end-to-end encryption.” *Id.*, 12. A POSITA would have understood and found obvious that because the distribution of keys is handled as a conference policy, it is handled at the application layer since the layers below the application layer (transport layer and below) do not track the specifics of any particular application’s policies; the lower layers are simply to transport and deliver the data provided by the applications. EX1003, ¶96.

To the extent *Handley* does not expressly disclose this element, it would have been obvious in view of *Handley* in combination with *Basturk*. *Id.*, ¶97.

Basturk discloses:

[T]he application passes its key *a* to the transport module, and leaves to the latter the task of applying any cryptographic functions or taking other security steps. The advantage is that the transport module can compute a signature which covers not only the data but also (relevant parts of) the transport header which it creates, particularly the byte sequence numbers. It can also sign the acknowledgements messages which exist only at the transport level.

EX1006, 4-5. A POSITA would have understood the key is a “protocol data unit available at the application layer” because the key is passed from the application to the transport module. EX1001, cl. 2; EX1003, ¶98. Further, as discussed in Ground I, Claim 1, a POSITA would have understood and found obvious that because the key is a secret key used by a transmitter to sign each packet in the communication,

it discloses “the application protocol being linked to a secret by the transmitter.” EX1003, ¶98.

4. Dependent Claim 3

See Ground I, Elements [1.1], [1.3], Claim 2. EX1003, ¶99.

5. Dependent Claim 4

See Ground I, Claim 2. *Id.*, ¶¶100-102.

Handley disclosed this Claim or at least rendered it obvious. *Id.*, ¶101. As described in Ground I, Element [1.1], *Handley* discloses the use of RTP. As Dr. Houh describes, a POSITA would have understood that RTP inherently includes sequence numbers and timestamps. *Id.*, ¶101. For example, RFC 1889 discloses that “the RTP header contains timing information and a sequence number that allow the receivers to reconstruct the timing produced by the source” and “RTP Fixed Header Fields” including “sequence number” and “timestamp.” EX1008, 5, §2.1, 10, §5.1. From this, a POSITA would have understood and found obvious that the sequence and timestamp are included in the RTP packet header and that the whole RTP packet payload is a PDU on the application layer. EX1003, ¶101. As discussed in Ground I, Element [2.1], a POSITA would have understood that a protocol data unit would be any part of a protocol exchange and, because RTP is available at the application layer, *Handley* discloses this element. *Id.*, ¶101.

To the extent *Handley* does not disclose this element, it would have been obvious in view of *Handley* in combination with *Basturk*. *Id.*, ¶102. *Basturk* discloses that “[w]e will see that of particular importance is the fact that the transport header [of the packet] includes the sequence number of the packet which is thereby covered by the MAC.” EX1006, 12. A POSITA would have understood and found obvious a packet on the application layer is a PDU, and the MAC that is included with the packet includes a sequence number. EX1003, ¶102.

6. Dependent Claim 5

See Ground I, Claim 2. *Id.*, ¶¶103-107.

Handley disclosed this Claim or at least rendered it obvious. *Id.*, ¶104. A POSITA would have understood and found obvious “a key known only to the receiver and the transmitter to be authenticated” is implementing symmetric encryption where the transmitter and receiver share a secret key. *Id.*, ¶101. As described in Ground I, Element [1.3] *Handley* discloses “encrypting a hash” and “digitally signing” each packet. EX1005, 13; EX1003, ¶101. A POSITA would have understood this to mean the secret is the use of both one-way hash function, as well as encryption of the resulting hash with the session key. EX1003, ¶104.

As described in Ground I, Element [1.1], *Handley* teaches RTP encryption using DES and MD5. EX1005, 13; EX1003, ¶105. A POSITA would have understood the secret would be the combination of MD5, a known one-way hash

function, and DES, a known symmetric encryption algorithm, using keys known only to the parties involved and keeping the underlying data encrypted in this manner secret from all parties who do not possess the secret key. EX1003, ¶105.

To the extent *Handley* does not disclose this element, it would have been obvious in view of *Handley* in combination with *Basturk*. EX1003, ¶106. *Basturk* discloses that “[b]oth signature computation and encryption/decryption rely on the use of secret information (key) shared by the two communicating parties.” EX1006, 8. For one-way hash functions, *Basturk* discloses “...MD5, SHA, DES, Double DES are the functions the upper layer can choose from” for encryption and signature schemes. EX1006, 16; EX1003, ¶106. As Dr. Houh explains, a POSITA would have understood that the secret is the encryption operation of DES or Double DES which utilize a secret key known only to transmitter and receiver, and the one-way hash function is MD5 or SHA. EX1003, ¶106.

Basturk also discloses the use of message authentication codes as described above in Ground I, Element [1.1]; EX1003, ¶107; EX1006, 9. A POSITA would have understood *Basturk* to mean α is a secret key known only to transmitter and receiver which is used to generate a MAC. EX1003, ¶107. Further, *Basturk* discloses that “MAC computations are kept simple and can start prior to context retrieval. In particular, we rely on MAC implementations of the form $\text{MAC}_\alpha(m) = f_\alpha(h(m))$, where h is a (hash) function depending neither on [the key] nor on the

cryptographic primitive chosen in the initial options.” EX1006, 13. A POSITA would have understood that “h is a (hash) function,” m is “a message m” and accordingly, the message is placed through a one-way hash function like MD5 or SHA, and the resulting hash is then encrypted using f_α , which may be a number of “cryptographic functions.” EX1006, 13, 18; EX1003, ¶107.

7. **Dependent Claim 6**

See Ground I, Claim 2. EX1003, ¶108.

Handley disclosed this Claim or at least rendered it obvious. *Id.*, ¶¶108-110. As discussed in Ground I, Elements [1.1], [1.3], [5.1], *Handley* discloses the use of symmetric and asymmetric encryption/decryption as well as the use of MD5. *Id.*, ¶109. A POSITA would have understood that the ascertaining in a pure encryption/decryption scenario utilizing only keys and no cryptographic checksum would stem from whether the receiver knows the transmitter’s secret key or has cached the receiver’s public key. *Id.*, ¶109. Dr. Houh details that in a scenario using both an encryption/decryption operation and a cryptographic checksum, a POSITA would have understood that (1) the transmitter runs data through a hash algorithm to create a message digest; (2) the digest is encrypted to create a digital signature using either a secret key or private key; (3) the data and digital signature are sent to the receiver; (4) the receiver decrypts the digital signature using a secret key or the transmitter’s public key to obtain the message digest; (5)

the data is run through the same hash algorithm as the transmitter did; and (6) the output of the hash algorithm is compared to the decrypted message digest. *Id.*, ¶109.

To the extent *Handley* does not expressly disclose this element it would have been obvious over *Handley* in combination with *Basturk*. As discussed in Ground I, Elements [1.1], [1.3], Claim 5, *Basturk* discloses the implementation of MAC signatures, hash functions, cryptographic checksums, as well as asymmetric or symmetric encryption and decryption that are used to authenticate and ascertain whether the receiver knows the transmitter. *Id.*, ¶110. A POSITA would have understood that the signature must also be decrypted because the signature was originally encrypted and, in the symmetric scenario, utilize the same key it was encrypted with. *Id.*, ¶110.

8. Dependent Claim 7

See Ground I, Claims 6, 2. *Id.*, ¶111.

9. Dependent Claim 8

See Ground I, Claim 2. *Id.*, ¶112.

Handley disclosed this Claim or at least rendered it obvious. *Id.*, ¶¶112-114. As discussed in Ground I, Elements [1.1], [1.3], Claim 5, *Handley* discloses the use of MD5 and asymmetric and symmetric encryption, *e.g.*, DES. As described in Ground I, Claim 6, a POSITA would have understood the linking to the secret is

the use of MD5 to create a hash or cryptographic checksum and DES with secret key to encrypt the data because only the parties that have the secret key can properly decrypt the data, keeping the data secret from other non-parties. *Id.*, ¶113.

To the extent *Handley* does not expressly disclose this element, this element would have been obvious in view of *Handley* in combination with *Basturk*. *Id.*, ¶114. As described in Ground I, Elements [1.1], [1.3], Claim 6, *Handley* discloses the use of MAC signatures, MD5, SHA, and asymmetric and symmetric encryption. EX1005, 13, 16. A POSITA would have understood and found obvious the linking to the secret is the use of MD5/SHA to create a hash or cryptographic checksum and a MAC with secret keys to encrypt the data because only the parties that have the secret key can properly decrypt the data, keeping the data secret from other non-parties. EX1003, ¶114.

10. Dependent Claim 9

See Ground I, Elements [1.1], [1.3], Claim 8. EX1003, ¶115.

11. Dependent Claim 10

See Ground I, Claims 4, 8. EX1003, ¶116.

12. Dependent Claim 11

See Ground I, Claims 5, 8. EX1003, ¶117.

13. Dependent Claim 12

See Ground I, Element [1.3], Claim 8. EX1003, ¶118.

14. Dependent Claim 13

See Ground I, Elements [1.1], [1.3], Claim 12. EX1003, ¶119.

15. Dependent Claim 14

See Ground I, Claims 4, 12. EX1003, ¶120.

16. Dependent Claim 15

See Ground I, Claims 5, 12. EX1003, ¶121.

17. Dependent Claim 16

See Ground I, Claims 6, 12. EX1003, ¶122.

18. Dependent Claim 17

See Ground I, Elements [1.1], [1.3], Claim 16. EX1003, ¶123.

19. Dependent Claim 18

See Ground I, Claims 4, 17. EX1003, ¶124.

20. Dependent Claim 19

See Ground I, Claims 5, 18. EX1003, ¶125.

21. Dependent Claim 20

See Ground I, Claim 19. EX1003, ¶126.

Handley disclosed this Claim or at least rendered it obvious. *Id.*, ¶¶126-128

As discussed in Ground I, Elements [1.1], [1.3], *Handley* discloses the use of DES, a symmetric key encryption algorithm. A POSITA would have understood that authentication is carried out between the transmitter and the receiver before the data is transmitted from the transmitter to the receiver because exchanging of a

secret key needs to occur prior to data transmission for a transmitter to encrypt the data with a secret key that the receiver can use to decrypt the data. *Id.*, ¶127.

To the extent *Handley* does not expressly disclose this element, this element would have been obvious in view of *Handley* in combination with *Basturk*. *Id.*, ¶128 As described in Ground I, Elements [1.1], [1.3], *Basturk* discloses DES, a symmetric key encryption algorithm. A POSITA would have understood that authentication is carried out between the transmitter and the receiver before the data is transmitted from the transmitter to the receiver because any exchanging of a secret key needs to occur prior to data transmission for a transmitter to encrypt the data with a secret key that the receiver can use to decrypt the data. EX1003, ¶128.

22. Dependent Claim 21

See Ground I, Claim 20. EX1003, ¶129.

Handley disclosed this Claim or at least rendered it obvious. *Id.*, ¶¶129-130. *Handley* discloses that “the real interest in using the internet for audio and video should come from the prospect for a single ubiquitous communications network that not only allows traditional telephony services, but also video, shared collaboration tools, and through IP Multicast, multi-party conferences and multimedia sessions that scale from small group meetings through to television sized audiences.” EX1005, 1. A POSITA would have known that the internet is a packet-switched network and any services, such as “traditional telephony services”

provided over the internet are internet telephony. EX1001 cl. 21; EX1003, ¶130.

As such, a POSITA would have known and understood that the receiver and transmitter are used in Internet telephony. EX1003, ¶130.

23. Dependent Claim 23

See Ground I, Claims 20, 21. EX1003, ¶131.

24. Independent Claims 25

a) **Preamble [25]**

See Ground I, Preamble [1]; EX1003, ¶132.

b) **Element [25.1]**

See Ground I, Elements [1.1], [1.2], [1.3], Preamble [25]; EX1003, ¶133.

c) **Element [25.2]**

See Ground I, Elements [1.1], [1.3], [1.4], [25.1]; EX1003, ¶134

d) **Element [25.3]**

See Ground I, Elements [1.1], [25.2], Claim 4; EX1003, ¶135.

A POSITA would have understood and found obvious that the whole RTP packet payload comprises a whole RTP packet payload header comprising at least one of a sequence number and a time stamp because RFC 1889 discloses that “the RTP header contains timing information and a sequence number that allow the receivers to reconstruct the timing produced by the source” and “RTP Fixed Header Fields” including a “sequence number” and “timestamp.” EX1008, 5, §2.1, 10, §5.1; EX1003, ¶136.

e) **Element [25.4]**

See Ground I, Elements [1.2], [25.3]; EX1003, ¶137.

f) **Element [25.5]**

See Ground I, Elements [1.1], [1.3], [25.4], Claim 4; EX1003, ¶138.

This element would have been obvious in view of *Handley* in combination with *Basturk*. *Id.*, ¶¶138-140. As described in Ground I, Elements [1.1], [1.3], *Basturk* discloses a MAC that “includes the transport header, the full data payload, and the trailer of the packet” as part of the MAC computation. EX1006, 12. As Dr. Houh describes, the common secret between the transmitter and the receiver is a session key α that is used to generate MAC_{α} . EX1006, 13; EX1003, ¶139.

Further, as described in Ground I, Element [4.1], *Handley* discloses RTP and, pursuant to RFC 1889, “the RTP header contains timing information and a sequence number that allow the receivers to reconstruct the timing produced by the source” and “RTP Fixed Header Fields” including “sequence number” and “timestamp.” EX1008, 5, §2.1, 10, §5.1. A POSITA would have understood that the RTP packet header includes a sequence number and timestamp. EX1003, ¶140. A POSITA would have appreciated when combining *Basturk*’s secure transport protocol to *Handley*’s RTP internet conferencing architecture, that to generate a MAC, the transmitter would use for computation, the contents of the RTP which

“includes the transport header, the full data payload, and the trailer of the packet,” as well as the sequence number and time stamp in the RTP header.

g) **[Element 25.6]**

See Ground I, Elements [1.1], [1.2], [1.3], [25.5]; EX1003, ¶141.

h) **Element [25.7]**

See Ground I, Elements [1.1], [1.3], [1.4], [25.6]; EX1003, ¶142.

25. Independent Claim 26

a) **Preamble [26]**

See Ground I, Preamble [1]; EX1003, ¶143.

b) **Element [26.1]**

See Ground I, Element [1.2]; EX1003, ¶144. A POSITA would have understood and found obvious that a transmitter contains a processor for transmitting data and computing the authentication data because in networking a transmitter is a type of computer that contains a processor for functions like encoding and managing the transmission protocol. EX1003, ¶144.

c) **Element [26.2]**

See Ground I, Preamble [1], Elements [1.1], [1.2], [26.1]; EX1003, ¶145.

d) **Element [26.3]**

See Ground I, Element [26.3], Claim 4; EX1003, ¶146.

e) **Element [26.4]**

See Ground I, Elements, [25.3], [25.4], [25.5], [25.6], [26.1], Claim 4; EX1003, ¶147.

26. Independent Claim 27

a) **Preamble [27]**

See Ground I, Preamble [1]; EX1003, ¶148.

b) **Element [27.1]**

See Ground I, Preamble [1], Preamble [27], Element [26.1]; EX1003, ¶149.

A POSITA would have understood and found obvious that a receiver, as with a transmitter, would also have a processing unit to decrypt data received from the transmitter and manage the protocol of the packets received from the transmitter. *Id.*, ¶150. Further a POSITA would have understood and found obvious that the processing unit of the receiver is communicably connecting to the transmitter through the use of device driver software that instructs the processor how to utilize the packets received from the network. *Id.*, ¶150.

c) **Element [27.2]**

See Ground I, Elements [1.1], [1.2], [1.3], [27.1]; EX1003, ¶151.

d) **Element [27.3]**

See Ground I, Elements [1.4], [27.2]; EX1003, ¶152.

e) **Element [27.4]**

See Ground I, Elements [25.3], [27.3], Claim 4; EX1003, ¶153.

f) **Element [27.5]**

See Ground I, Elements [1.1], [1.2], [1.3], [25.5], [27.4], Claim 4; EX1003, ¶154.

A POSITA would have understood and found obvious that the receiver, obtains the authentication data after the transmitter transmits data because it was added to the end of the whole RTP packet payload as a MAC which travels with the whole RTP packet payload. EX1003, ¶¶154-155.

27. Independent Claim 28

a) **Preamble [28]**

See Ground I, Preamble [1]; EX1003, ¶156.

b) **Element [28.1]**

See Ground I, Elements [1.1], [1.2], [28.1]; EX1003, ¶157.

c) **Element [28.2]**

See Ground I, Elements [1.3], [28.1]; EX1003, ¶158.

d) **Element [28.3]**

See Ground I, Elements [1.4] [28.2]; EX1003, ¶159.

e) **Element [28.4]**

See Ground I, Element [28.3], Claim 4; EX1003, ¶160.

f) **Element [28.5]**

See Ground I, Elements [1.2], [28.4]; EX1003, ¶161.

g) **Element [28.6]**

See Ground I, Elements [1.1], [1.2], [1.3], [25.5], [28.5], Claim 4; EX1003, ¶162

h) **Element [28.7]**

See Ground I, Elements [1.1], [1.2], [1.3], [28.6], Claim 4; EX1003, ¶163.

i) **Element [28.8]**

See Ground I, Elements [1.1], [1.3], [1.4], [25.7]; EX1003, ¶164.

B. Ground II: Claims 1-28 Are Rendered Obvious Over *PacketCable* in Combination With *Handley*

1. Motivation to Combine

A POSITA would have been motivated to modify *PacketCable* in view of *Handley* for a number of reasons. EX1003, ¶¶165-170. Both references are in the field of data transport over the internet and, more specifically, directed to standardizing the transport of multimedia. *Id.*, ¶165. For example, *PacketCable* discloses that the “PacketCable project is aimed at defining interface specification that can be used to develop interoperable equipment capable of providing packet-based voice, video and other high-speed multimedia services over hybrid fiber coax (HFC) cable systems utilizing the DOCSIS protocol.” EX1007, 9. As Dr. Houh describes, and as discussed in Ground I, Claim 22, *Handley* discloses the interest of a “single ubiquitous communications network” for, but not limited to, audio and video. EX1007, 9; EX1003, ¶165.

Although *PacketCable* does not expressly disclose accepting/rejection of the whole RTP packet payload, the use of *PacketCable*'s disclosure of its PacketCable project which provides high-speed multimedia data transport using RTP in combination with *Handley*'s RTP internet conferencing architecture and its teaching of discarding of packets would have been obvious to a POSITA in view of *PacketCable*. EX1003, ¶166.

A POSITA would have also been motivated to modify *PacketCable* in view of *Handley* because both references understood and describe the importance of authentication in networks. *Id.*, ¶167. For example, *PacketCable* discloses that “[d]esign considerations include confidentiality, authentication, integrity, non-repudiation and access control.” EX1007, 17. As described in Ground I, Element [1.1], *Handley* discloses similar security measures. EX1003, ¶167. *Handley* was published in February 1999, is prior art, relates the field of computer networking, and a POSITA would have known of and looked to its teachings and motivations. *Id.*, ¶167.

The similarities between the references are substantial and a POSITA would have had a reasonable expectation of success in combining *PacketCable* with *Handley*. *Id.*, ¶168. *PacketCable* discloses that “PacketCable core security services include a mechanism for providing end-to-end encryption of RTP media streams, thus substantially reducing the threat to privacy.” EX1007, 42-43. *Handley*

similarly discloses end-to-end encryption with RTP as described in Ground I, Claim 2. EX1006, 12; EX1003, ¶168.

A POSITA would have understood that the hardware and systems to implement authentication that accepts and rejects packets would be necessary and included in view of *PacketCable*'s multimedia services architecture and *Handley*'s internet conferencing architecture. EX1003, ¶169

Thus, use of *PacketCable*'s multimedia services architecture with *Handley*'s teaching of discarding unwanted packets would have made use of a known data transmission architecture and known hardware in a known application to improve a similar method in the same way. *Id.*, ¶170. The combination also applies a known technique to a method ready for improvement to yield predictable results. *Id.*, ¶171. Using a data authentication technique in combination with data transmission involved the use of routine computers, equipment, protocols, and programming techniques, and a POSITA would have had a reasonable expectation of success in using the disclosed internet conferencing architecture with *Handley*. *Id.*, ¶170.

2. Independent Claim 1

a) Preamble [1]

To the extent the preamble is limiting, *PacketCable* disclosed, or at least rendered obvious, the preamble. *Id.*, ¶¶171-172. As described in Ground I, Preamble[1], Element [1.1], data transmission over the internet and the OSI Model

were known. *Id.*, ¶171. *PacketCable* discloses “interoperable equipment capable of providing packet-based voice, video and other high-speed multimedia services.” EX1007, 9. Further, *PacketCable* discloses the implementation of multimedia terminal adapters (“MTA”), which are “a PacketCable client device that contains a subscriber-side interface to the subscriber’s CPE (e.g., telephone) and a network-side signaling interface to call control elements in the network.” *Id.*, 18. MTAs are responsible for providing “codecs and all signaling and encapsulation functions required for media transport and call signaling.” *Id.* Further, *PacketCable* discloses that the “pkt-s6” interface is “end-to-end media packets between two MTAs.” *Id.*, 45. A POSITA would have understood that the MTAs in *PacketCable* are transmitting data between one another and are the equivalent to a transmitter and receiver because the “pkt-6” interface connects two MTAs to send end-to-end media packets. EX1003, ¶171.

To the extent *PacketCable* does not expressly disclose this element, it would have been obvious in view of *PacketCable* in combination with *Handley*. See Ground I, Preamble [1]; EX1003, ¶172.

b) **Element [1.1]**

See Ground II, Preamble [1]; EX1003, ¶173.

PacketCable disclosed this element or at least rendered it obvious. *Id.*, ¶¶173-175. *PacketCable* explicitly discloses RTP, an application protocol whose

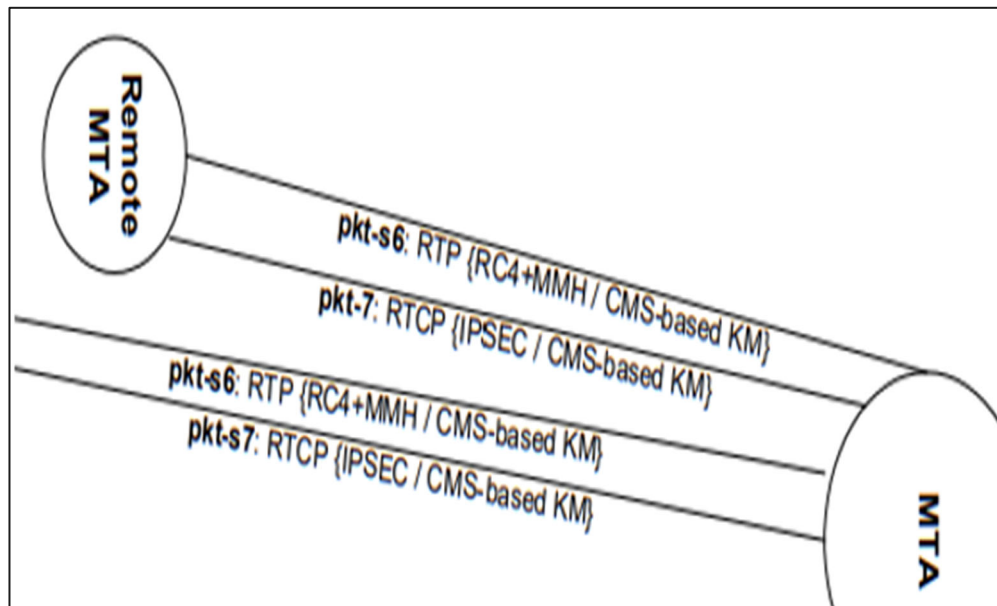
packets are created at the application layer. EX1007, 45. For example, *PacketCable* discloses that “RTP packets are encrypted directly with RC4, without any additional security layers. An MMH-based MAC (Message Authentication Code) optionally provides message integrity. Keys are distributed by the CMS to the two endpoints.” *Id.* As described in Ground I, Element [1.1], a POSITA would have understood (1) that manipulating RTP packets, *e.g.*, implementing RC4 or adding a MAC, results in authentication at the packet level and (2) RTP packets are created at the application layer because RTP is an application protocol. EX1003, ¶174. As such, *PacketCable* discloses this element. *Id.*, ¶174.

To the extent *PacketCable* does not expressly disclose this element, it would have been obvious in view of *PacketCable* in combination with *Handley*. See Ground I, Element [1.1], *Id.*, ¶175.

c) **Element [1.2]**

See Ground II, Preamble [1], Element [1.1]; *Id.*, ¶176.

PacketCable disclosed this element or at least rendered it obvious. *Id.*, ¶¶176-179. *PacketCable* discloses MAC utilization for message integrity and illustrates in Figure 13 (reproduced below) that the “RC4 + MMH / CMS-based KM” encrypts and adds authentication to the whole RTP packet in a “pkt-s6” interface. EX1007, 44.



EX1007, Fig. 13

Further, *PacketCable* discloses that the “MAC computation spans the packet’s unencrypted header and encrypted payload.” EX1007, 47. A POSITA would have understood and found obvious that placement of authentication data in the middle of the packet payload would negatively impact performance, cause packet fragmentation issues, and complicate the implementation of networking software. EX1003, ¶178. As such, a POSITA would have avoided such placement. *Id.*, ¶178. With this knowledge, a POSITA would have understood available authentication data placement options to include either at the beginning or the end of the payload. Further, a POSITA would have understood that the authentication data would go at the end of the whole RTP packet payload because it would reduce the latency of delivering the packet since placing the authentication data at the beginning of the payload requires storing the packet, computing the authentication

data, transmitting the packet to the network, and placing the authentication data at the end of the payload, which allows the packet to be transmitted to the network while calculating the authentication data then adding it to the end of the payload while the packet transmission is being completed. EX1003, ¶178. A POSITA would have had “finite number of identified, predictable solutions” and “good reason” to pursue a modification with either of the two identified placement solutions. EX1003, ¶178; *KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 402 (2007). Further, a POSITA would have understood and found obvious that placing authentication at the end of the payload would allow the transmitter to transmit data while simultaneously computing the checksum and prevent any unnecessary transmission delay by having to first compute the checksum and then transmit the payload. EX1003, ¶178. Thus, a POSITA would have understood and found obvious that placing authentication at the end of the whole RTP packet payload would be the best solution to increase transmission speeds and decrease memory usage needed to encrypt and decrypt the data. *Id.*, ¶178. A POSITA would have appreciated that the transmitter is the device to insert authentication data for the receiver to use and verify that it knows the transmitter and the data was not manipulated while in transit. *Id.*, ¶178.

To the extent *PacketCable* does not expressly disclose this element, it would have been obvious in view of *PacketCable* in combination with *Handley*. See Ground I, Element [1.2]; EX1003, ¶179.

d) **Element [1.3]**

See Ground II, Preamble [1], Elements [1.1], [1.2]; EX1003, ¶180.

PacketCable disclosed this element or at least rendered it obvious. EX1003, ¶¶180-183. First, a POSITA would have understood that *PacketCable* discloses the use of RTP packet level authentication data and that this data is inserted by the transmitter. *Id.*, ¶181. For example, *PacketCable* discloses that in a “pkt-s6” interface there is “[e]nd-to-end media packets between two MTAs, or between MTA and MG. RTP packets are encrypted **directly** with RC4, without any additional security layers. An MMH-based MAC (Message Authentication Code) optionally provides message integrity. Keys are distributed by the CMS to the two endpoints.” EX1007, 45. A POSITA would have understood this disclosure to mean RTP packet level authentication because “RTP packets are encrypted directly with RC4” at the RTP packet level. EX1003, ¶181; EX1007, 45. Further, *PacketCable* discloses that “[e]ach media RTP packet is encrypted for privacy” where “[e]ach RTP packet may include an optional message authentication code (MAC)” and that the “MAC algorithm can also be negotiated.” EX1007, 47. A POSITA would have understood that this use of MAC is RTP packet level

authentication because each RTP packet has an included MAC at the RTP packet level. EX1003, ¶181.

Further, a POSITA would have understood and found obvious the receiver is ascertaining whether it knows the transmitter on the RTP packet level authentication described above because the authentication data is generated using symmetric or asymmetric encryption methods. Further, as discussed in Ground I, Elements [1.1], [1.3], a POSITA would have understood that a receiver knows whether the transmitter is using symmetric or asymmetric encryption. EX1003, ¶182. If using symmetric key encryption, the transmitter and receiver will exchange secret keys prior to data transmission, so the transmitter can use the secret key to encrypt the data and the receiver can use the secret key to decrypt the data. *Id.*, ¶182. *PacketCable* states that “[k]eys for the encryption and MAC calculation are derived from the End-End secret, which is exchanged between sending and receiving MTA as part of the call signaling.” EX1007, 47. *PacketCable* discloses that a secret key is “the cryptographic key used in a symmetric key algorithm, which results in the secrecy of the encrypted data depending solely upon keeping the key a secret, also known as symmetric key.” *Id.*, 66. A POSITA would have understood and found obvious that a receiver MTA knows the shared secret key, which it then uses with the received message to see if

the computed MAC matches the MAC received from the transmitter MTA. EX1003, ¶1832.

To the extent *PacketCable* does not expressly disclose this element, it would have been obvious in view of *PacketCable* in combination with *Handley*. See Ground I, Elements [1.1], [1.3]; EX1003, ¶183.

e) **Element [1.4]**

See Ground II, Preamble [1], Elements [1.1], [1.2], [1.3]; EX1003, ¶184.

PacketCable disclosed this element or at least rendered it obvious. EX1003, ¶¶184-186. A POSITA would have understood that if the authentication data, *i.e.*, the MAC, is not verified and accepted by the receiver, then the receiver would reject the data, meaning it would not process or decrypt the data. EX1003, ¶185.

Further, this element would have been obvious in view of *PacketCable* in combination with *Handley*. See Ground I, Element [1.4]; EX1003, ¶186.

3. Dependent Claim 2

See Ground II, Claim 1; EX1003, ¶187.

PacketCable disclosed this Claim or at least rendered it obvious. EX1003, ¶¶187-189. A POSITA would have understood and found obvious that a PDU is a single unit of information at a particular OSI layer (*e.g.*, an RTP packet at the application layer). *Id.*, ¶188. *PacketCable* discloses that “[k]eys for the encryption and MAC calculation are derived from the End-End secret, which is exchanged

between sending and receiving MTA as part of the call signaling. Thus, the key exchanges for media stream security are secured themselves by the call signaling security.” EX1007, 47. A POSITA would appreciate that the PDU in *PacketCable*, e.g., an RTP packet, is linked to a secret in two possible ways. EX1003, ¶188. First, the RTP packet may be encrypted and is linked to the secret via the keys used to encrypt and decrypt the RTP packet. *Id.*, ¶188. Second, the RTP packet may be used as the basis of an authentication code, e.g., the MAC, and is linked to the secret when the hash is encrypted. *Id.*, ¶188.

To the extent *PacketCable* does not expressly disclose this element, it would have been obvious in view of *PacketCable* in combination with *Handley*. See Ground I, Element [2.1], EX1003, ¶189.

4. Dependent Claim 3

See Grounds I-II, Elements [1.1], [1.3], Claim 2; EX1003, ¶190.

5. Dependent Claim 4

See Grounds I-II, Claims 2, 4; EX1003, ¶191.

PacketCable disclosed this Claim or at least rendered it obvious. EX1003, ¶¶191-192. *PacketCable* discloses the implementation of RTP in its architecture. EX1007, 31; EX1003, ¶192. As Dr. Houh describes, and as noted above in Ground I, Claims 4, a POSITA would have understood RTP inherently includes sequence numbers and timestamps. EX1003, ¶192. As discussed in Grounds I-II, Claim 2, a

POSITA would have understood that a protocol data unit would be anything part of a protocol exchange and, therefore, *PacketCable* discloses this element because RTP (including the sequence number and time stamp) is available at the application layer. *Id.*, ¶192.

6. Dependent Claim 5

See Grounds I-II, Claim 2. EX1003, ¶194.

PacketCable disclosed this Claim or at least rendered it obvious. EX1003, ¶¶193-196. *PacketCable* discloses the use of a Hashed Message Authentication Code (“HMAC”) which is an encryption operation with a secret key that uses a one-way hash function like MD5, or SHA-1. *Id.*, ¶194. For example, *PacketCable* discloses “HMAC – Hashed Message Authentication code – a message authentication algorithm, based on either SHA-1 or MD5 hash.” EX1007, 62. As described in Ground II, Element [1.3], *PacketCable* discloses that “[k]eys for the encryption and MAC calculation are derived from the End-End secret, which is exchanged between sending and receiving MTA as part of the call signal.” *Id.*, 47. A POSITA would have understood that this end-end secret may be a secret key in a symmetric encryption scenario and known only to the transmitter and receiver.

PacketCable describes one-way hash functions and MD5. EX1007, 72; EX1003, ¶195. A POSITA would have understood and found obvious that MD5 is a one-way hash algorithm that may be implemented in a symmetric key scenario

such as HMAC, where keys are known only to the parties involved, *i.e.*, the transmitter and receiver. EX1003, ¶195.

To the extent *PacketCable* does not expressly disclose this element, it would have been obvious in view of *PacketCable* in combination with *Handley*. See Ground I, Claim 5; EX1003, ¶196.

7. Dependent Claim 6

See Grounds I-II, Claim 2; EX1003, ¶197.

PacketCable disclosed this Claim or at least rendered it obvious. EX1003, ¶197-200. As described in Ground II, Elements [1.1], [1.3] and [5.1], *PacketCable* discloses the use of symmetric and asymmetric encryption/decryption as well as the use of MD5. *Id.*, ¶198. A POSITA would have understood that the ascertaining would stem from whether the receiver knows the transmitter's secret key or has cached the receiver's public key in an encryption/decryption scenario that uses only keys and no cryptographic checksum. *Id.*, ¶198. As Dr. Houh describes, and as noted in Grounds I, Claim 6, a POSITA would have understood and found obvious the steps required to implement encryption and a cryptographic checksum. *Id.*, ¶198.

As discussed in Ground II, Elements [1.2] and [5.1], *PacketCable* discloses the implementation of MAC signatures, as well as asymmetric or symmetric encryption and decryption. *Id.*, ¶199. A POSITA would have understood that

because the signature is encrypted it must also be decrypted and, in the symmetric scenario, uses the same key it was encrypted with. *Id.*, ¶199.

To the extent *PacketCable* does not expressly disclose this element, it would have been obvious in view of *PacketCable* in combination with *Handley*. *Id.*, ¶200. See Ground I, Elements [1.1], [1.2], [1.3], Claim 6. EX1003, ¶200.

8. Dependent Claim 7

See Grounds I-II, Claims 2, 6; EX1003, ¶201

9. Dependent Claim 8

See Grounds I-II, Claim 2; EX1003, ¶202.

PacketCable disclosed this Claim or at least rendered it obvious. EX1003, ¶¶202-204. As discussed in Ground II, Elements [1.1], [1.3], [5.1], *PacketCable* discloses the use of MD5 and asymmetric and symmetric encryption. A POSITA would have understood that linking to the secret is carried out by either asymmetric or symmetric encryption as well as a cryptographic checksum and key information since only the parties that have the secret key can properly decrypt the data. Thus, it keeps the data secret from other non-parties. EX1003, ¶203.

To the extent *PacketCable* does not expressly disclose this element, this element would have been obvious in view of *PacketCable* in combination with *Handley*. EX1003, ¶204. As discussed in Ground I, Elements [1.1], [1.3], [5.1],

Handley discloses the use of MAC signatures, MD5, SHA, and the use of asymmetric and symmetric encryption. EX1006, 13, 16.

10. Dependent Claim 9

See Grounds I-II, Elements [1.1], [1.3], Claim 8; EX1003, ¶205.

11. Dependent Claim 10

See Grounds I-II, Claims 4, 8; EX1003, ¶206.

12. Dependent Claim 11

See Grounds I-II, Claims 5, 8; EX1003, ¶207.

13. Dependent Claim 12

See Grounds I-II, Elements [1.1], [1.3], Claim 8; EX1003, ¶208.

14. Dependent Claim 13

See Grounds I-II, Elements [1.1], [1.3], Claim 12; EX1003, ¶209.

15. Dependent Claim 14

See Grounds I-II, Claims 4, 12; EX1003, ¶210.

16. Dependent Claim 15

See Grounds I-II, Claims 5, 12; EX1003, ¶211.

17. Dependent Claim 16

See Grounds I-II, Claims 6, 1; EX1003, ¶212

18. Dependent Claim 17

See Grounds I-II, Elements [1.1], [1.3], Claim 16; EX1003, ¶213

19. Dependent Claim 18

See Grounds I-II, Claims 4, 17; EX1003, ¶214.

20. Dependent Claim 19

See Grounds I-II, Claims 5, 19; EX1003, ¶215.

21. Dependent Claim 20

See Grounds I-II, Claim 19.

PacketCable disclosed this Claim or at least rendered it obvious. As discussed in Ground II, Element [1.3], *PacketCable* discloses the use of symmetric keys. A POSITA would have understood that authentication is carried out between the transmitter and the receiver before the data is transmitted from the transmitter to the receiver because exchanging of symmetric keys occurs prior to data transmission. EX1003, ¶¶216-218.

To the extent *PacketCable* does not expressly disclose this element, this element would have been obvious in view of *PacketCable* in combination with *Handley. Id.*, ¶218. As discussed in Ground I, Elements [1.1], [1.3], *Handley* discloses DES, a symmetric key encryption algorithm, and a POSITA have understood and found obvious that DES keys need to be exchanged prior to encryption such that the transmitter can use the secret key to encrypt the data, and the receiver can use the secret key to decrypt the data. *Id.*, ¶218.

22. Dependent Claim 21

See Grounds I-II, Claim 20. EX1003, ¶219.

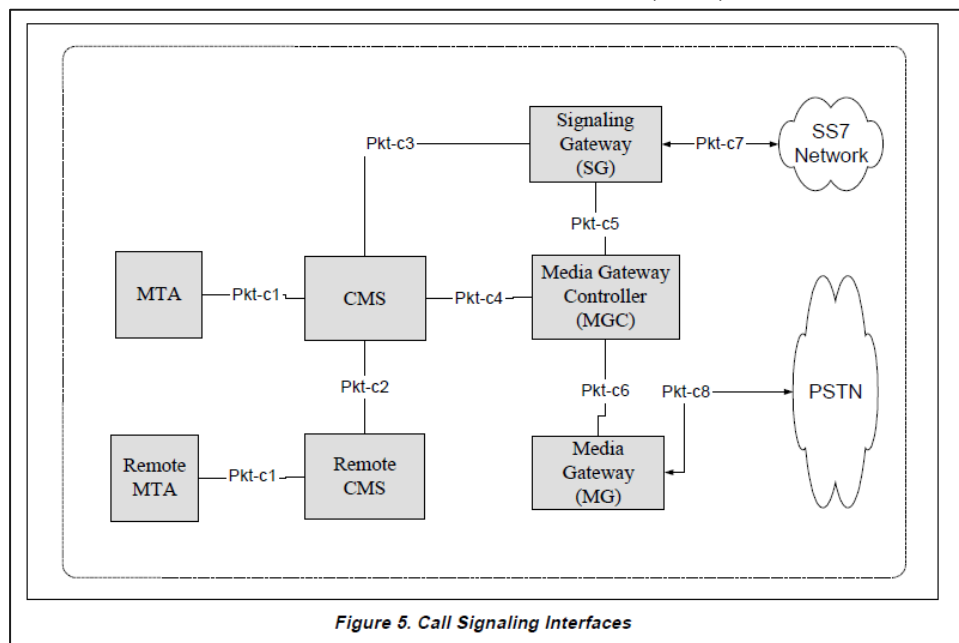
PacketCable disclosed this Claim or at least rendered it obvious. *PacketCable* discloses that “MTA – Media Terminal Adapter – contains the interface to subscribers’ CPE, a network interface, CODECs, and all signaling and encapsulation functions required for VoIP transport, class features signaling, and QoS signaling.” EX1007, 64. EX1003, ¶¶219-221. A POSITA would have understood that “VoIP” stands for “Voice over Internet Protocol,” a technology that uses packets to carry telephone traffic for Internet telephony.

To the extent *PacketCable* does not expressly disclose this element, this element would have been obvious in view of *PacketCable* in view of *Handley*. See Ground I, Claim 21; EX1003, ¶221

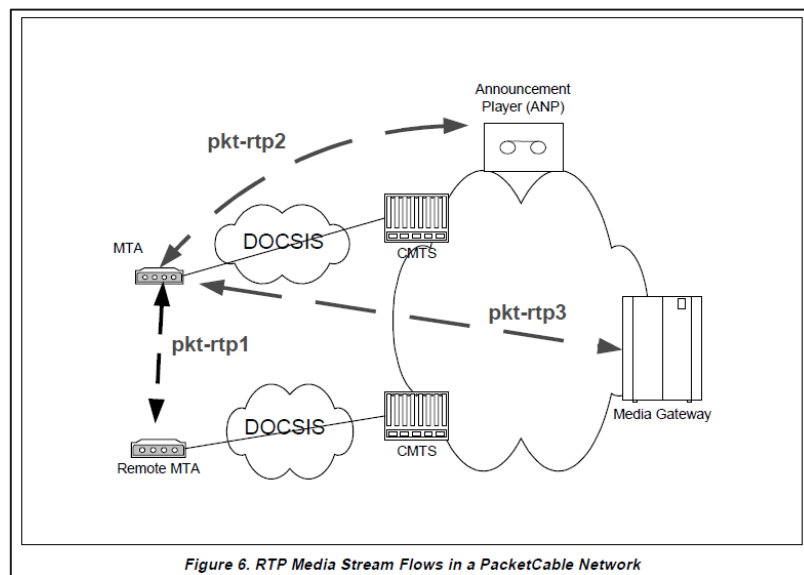
23. Dependent Claim 22

See Ground II, Claim 21. EX1003, ¶222.

PacketCable disclosed this Claim or at least rendered it obvious. EX1007, 64. EX1003, ¶¶222-224. *PacketCable* discloses the use of media gateways in Figures 5 and 6.



EX1007, Figure 5.



EX1007, Figure 6.

A POSITA would have understood the media gateways act to switch data between a particular IP flow and a particular PSTN port on the gateway, and the receiver and transmitter would be disposed in these switches. EX1003, ¶224. Further, a POSITA would have understood and found obvious to include this

functionality in a media gateway, which terminates the IP data side into the PSTN and ultimately acts as a switch between a particular IP flow and a particular PSTN port. *Id.*, ¶224. Inclusion of a gateway would have been obvious to a POSITA due to the benefits of having data network devices communicate with all pre-existing telephones on the PSTN such that the data network devices can conduct a telephone call with users beyond just those on the data network. *Id.*, ¶224.

24. Dependent Claim 23

See Grounds I-II, Claims 21, 23. EX1003, ¶225.

25. Dependent Claim 24

See Ground II, Claim 22; Grounds I-II, Claim 23. EX1003, ¶226.

26. Independent Claims 25

a) **Preamble [25]**

See Grounds I-II, Preamble [1]; EX1003, ¶227.

b) **Element [25.1]**

See Grounds I-II, Elements [1.1], [1.2], [1.3], Preamble [25]; EX1003, ¶228.

c) **Element [25.2]**

See Grounds I-II, Elements [1.1], [1.3], [1.4], [25.1], [25.2]; EX1003, ¶229.

d) **Element [25.3]**

See Grounds I-II, Elements [1.1], [25.2], [25.3], Claim 4; EX1003, ¶230.

A POSITA would have understood and found obvious that the whole RTP packet payload comprises a whole RTP packet payload header comprising at least

one of a sequence number and a time stamp because RFC 1889 discloses that “the RTP header contains timing information and a sequence number that allow the receivers to reconstruct the timing produced by the source” and “RTP Fixed Header Fields” including “sequence number” and “timestamp.” EX1008, 5, §2.1, 10, §5.1; EX1003, ¶231.

e) **Element [25.4]**

See Grounds I-II, Elements [1.2], [25.3], [25.4]; EX1003, ¶232.

f) **Element [25.5]**

See Grounds I-II, Elements [1.1], [1.3], [25.4], [25.5], Claim 4; EX1003, ¶233.

PacketCable disclosed this element or at least rendered it obvious. EX1003, ¶¶233-234 As described in Ground II, Elements [1.1], [1.2], [1.3], [1.4], *PacketCable* discloses that the “MAC computation spans the packet’s unencrypted header and encrypted payload.” EX1007, 47. A POSITA would have understood and found obvious that RTP inherently includes a sequence number and timestamp as described in Grounds I-II, Claim 4. EX1003, ¶234. Further, because “RTP packets are encrypted directly with RC4,” a symmetric encryption scheme with secret keys, that there is a common secret between the transmitter and receiver. EX1007, 45; EX1003, ¶234.

g) **Element [25.6]**

See Grounds I-II, Elements [1.1], [1.2], [1.3], [25.5], [25.6]; EX1003, ¶235.

h) **Element [25.7]**

See Ground I, Elements [1.1], [1.3], [1.4], [25.6], [25.7]; EX1003, ¶236.

27. Independent Claim 26

a) **Preamble [26]**

See Ground I, Preamble [1]; EX1003, ¶237.

b) **Element [26.1]**

See Ground I-II, Elements [1.2], [26.1]; EX1003, ¶238.

c) **Element [26.2]**

See Ground I-II, Preamble [1], Elements [1.1], [1.2], [26.1], [26.2]; EX1003, ¶239.

d) **Element [26.3]**

See Grounds I-II, Elements [26.2], [26.3], Claim 4; EX1003, ¶240.

e) **Element [26.4]**

See Grounds I-II, Elements [1.1], [1.2], [1.3], [25.3], [25.4], [25.5], [25.6], [26.1], Claim 4; EX1003, ¶241.

28. Independent Claim 27

a) **Preamble [27]**

See Grounds I-II, Preamble [1]; EX1003, ¶242.

b) **Element [27.1]**

See Grounds I-II, Preamble [1], Preamble [27], Element [26.1]; EX1003, ¶243.

A POSITA would have understood and found obvious that, similar to a transmitter, a receiver would also have a processing unit to decrypt data received from the transmitter and manage the protocol of the packets received from the transmitter. EX1003, ¶240. Further, a POSITA would have understood and found obvious that the processing unit of the receiver is communicably connecting to the transmitter through the use of device driver software that instructs the processor how to utilize the packets received from the network. *Id.*, ¶243.

c) **Element [27.2]**

See Grounds I-II, Elements [1.1], [1.2], [1.3], [27.1], [27.2]; EX1003, ¶244.

d) **Element [27.3]**

See Grounds I-II, Elements [1.4], [27.2], [27.3]; EX1003, ¶245.

e) **Element [27.4]**

See Ground I, Elements [25.3], [27.3], Claim 4; EX1003, ¶246.

f) **Element [27.5]**

See Grounds I-II, Elements [1.1], [1.2], [1.3], [25.5], [27.4], [27.5], Claim 4. EX1003, ¶247.

A POSITA would have understood and found obvious that the receiver and its processing unit would obtain the authentication data after the transmitter transmits data because it was added to the end of the whole RTP packet payload as a MAC, and the MAC travels with the whole RTP packet payload. EX1003, ¶248.

29. Independent Claim 28

a) **Preamble [28]**

See Grounds I-II, Preamble [1], Preamble [28]; EX1003, ¶249.

b) **Element [28.1]**

See Grounds I-II, Elements [1.1], [1.2], [28.1]; EX1003, ¶250.

c) **Element [28.2]**

See Grounds I-II, Elements [1.3], [28.1], [28.2]; EX1003, ¶251.

d) **Element [28.3]**

See Grounds I-II, Elements [1.4], [28.2], [28.3]; EX1003, ¶252.

e) **Element [28.4]**

See Grounds I-II, Elements [28.3], [28.4], Claim 4; EX1003, ¶253.

f) **Element [28.5]**

See Grounds I-II, Elements [1.2], [28.4]; EX1003, ¶254.

g) **Element [28.6]**

See Grounds I-II, Elements [1.1], [1.2], [1.3], [25.5], [28.5], Claim 4;
EX1003, ¶255.

h) **Element [28.7]**

See Grounds I-II, Elements [1.1], [1.2], [1.3], [28.6], Claim 4. EX1003,
¶256.

i) **Element [28.8]**

See Grounds I-II, Elements [1.1], [1.3], [1.4], [25.7], [28.7], [28.8]; EX1003, ¶257.

C. Ground III: Claims 1-28 Are Rendered Obvious Over *PacketCable* in Combination with *Basturk*

1. Motivation to Combine

Although *PacketCable* does not expressly disclose accepting/rejecting the whole RTP packet payload, using *PacketCable*' architecture framework which provides high-speed multimedia data transport using RTP in combination with *Basturk*'s secure transport protocols would have been obvious to a POSITA in view of *PacketCable*. EX1003, ¶¶258-263. A POSITA would have been motivated to use *Basturk*'s "on-the-fly" signature computation with *PacketCable*' architecture framework due to the known communication benefits, which include faster data transmission, reduced jitter, and reduced computing power.

For example, *PacketCable* aimed to "enable real-time multimedia communications" in a high-speed environment. EX1007, 14. Similarly, as described in X.A.1, *Basturk*'s goal was to compute signatures "on-the-fly" to "avoid slowing down the regular flow of data." EX1006, 12; EX1003, ¶259.

A POSITA would have also appreciated that, although *Basturk* does not explicitly disclose RTP, its secure transport protocol is compatible with *PacketCable*'s architectural framework EX1003, ¶260. For example, *Basturk*

discloses that its proposed protocol “should be able to operate in both environments,” including “software encryption and decryption for secure video conferences” at both low- and high-resolution environments. EX1006, 5. A POSITA would have understood that RTP is commonly used for video conferences. EX1003, ¶261. Further, *Basturk* teaches its discussion of security features “reflects the need to add security functionality in a way which is consistent with, and best matches, the existing communication structure” and so that “[a]pplications using the regular transport protocol should not have to be modified at all.” EX1006, 3, 5.

A POSITA would have understood that the systems of *PacketCable* and *Basturk* are compatible or related because both implement MACs for authentication. EX1003, ¶262. In addition to the reasons described in Sections X.A.1 and X.B.1, a POSITA would have understood that *PacketCable* recognized the need for authentication and discloses that the “PacketCable core security services include a mechanism for providing end-to-end encryption of RTP media streams, thus substantially reducing the threat to privacy.” EX1007, 42-43; EX1003, ¶263.

In using *PacketCable*’s architectural framework and *Basturk*’s secure transport protocol, a POSITA would have understood that hardware to implement multi-media transport may be necessary and included. EX1003, ¶264.

Thus, use of *PacketCable's* architectural framework, in conjunction with *Basturk's* secure transport protocol, would have made use of known multi-media data transport architecture and a known authentication method to improve data transmission security. *Id.*, ¶265. Said combination also applies a known technique to a known device or method ready for improvement to yield predictable results. *Id.*, ¶265. Using an internet conferencing architecture in combination with a secure transport protocol involves the use of routine computers, equipment, and techniques, and a POSITA would have had a reasonable expectation of success in using “on-the-fly” data authentication with *PacketCable*. A POSITA would have found the use and implementation of such hardware to implement *Basturk's* authentication methods with the internet conferencing architecture obvious. *Id.*, ¶265.

2. Independent Claim 1

a) Preamble [1]

See Grounds I-III, Preamble [1]; EX1003, ¶264.

b) Element [1.1]

See Grounds I-III, Preamble [1], Element [1.1]; EX1003, ¶265.

c) Element [1.2]

See Grounds I-III, Preamble [1], Elements [1.1], [1.2]; EX1003, ¶266.

d) Element [1.3]

See Grounds I-III, Preamble [1], Elements [1.2], [1.3]; EX1003, ¶267.

e) **Element [1.4]**

See Grounds I-III, Preamble [1], [1.3], [1.4]; EX1003, ¶268.

3. Dependent Claim 2

See Grounds I-III, Claim 2; EX1003, ¶269.

4. Dependent Claim 3

See Grounds I-III, Elements [1.1], [1.3], Claim 2, 3; EX1003, ¶270.

5. Dependent Claim 4

See Grounds I-III, Claims 2, 4; EX1003, ¶271.

6. Dependent Claim 5

See Grounds I-III, Claims 2, 5; EX1003, ¶272.

7. Dependent Claim 6

See Grounds I-III, Claims 2, 6; EX1003, ¶273.

8. Dependent Claim 7

See Grounds I-III, Claims 2, 6, 7; EX1003, ¶274.

9. Dependent Claim 8

See Grounds I-III, Claims 2, 8; EX1003, ¶275.

10. Dependent Claim 9

See Grounds I-III, Elements [1.1], [1.3], Claims 8, 9; EX1003, ¶276.

11. Dependent Claim 10

See Grounds I-III, Claims 4, 8, 10; EX1003, ¶277.

12. Dependent Claim 11

See Grounds I-III, Claims 5, 8, 11; EX1003, ¶278.

13. Dependent Claim 12

See Grounds I-III, Elements [1.1], [1.3], Claims 8, 12; EX1003, ¶279.

14. Dependent Claim 13

See Grounds I-III, Elements [1.1], [1.3], Claims 12, 13; EX1003, ¶280.

15. Dependent Claim 14

See Grounds I-III, Claims 4, 12, 14; EX1003, ¶281.

16. Dependent Claim 15

See Grounds I-III, Claims 5, 12, 15; EX1003, ¶282.

17. Dependent Claim 16

See Grounds I-III, Claims 6, 1, 16; EX1003, ¶283.

18. Dependent Claim 17

See Grounds I-III, Elements [1.1], [1.3], Claims 16, 17; EX1003, ¶284.

19. Dependent Claim 18

See Grounds I-III, Claims 4, 17, 18; EX1003, ¶285.

20. Dependent Claim 19

See Grounds I-III, Claims 5, 19; EX1003, ¶286.

21. Dependent Claim 20

See Grounds I-III, Claims 19, 20; EX1003, ¶287.

22. Dependent Claim 21

See Grounds I-III, Claims 20, 21; EX1003, ¶288.

23. Dependent Claim 22

See Ground II-III, Claims 21, 22. EX1003, ¶289.

24. Dependent Claim 23

See Grounds I-III; Claims 21, 23, EX1003, ¶290.

25. Dependent Claim 24

See Grounds II-III, Claims 22, 24; Grounds I-II, Claims 22, 23; EX1003, ¶291.

26. Independent Claims 25

a) **Preamble [25]**

See Grounds I-III, Preamble [1]; EX1003, ¶292.

b) **Element [25.1]**

See Grounds I-III, Elements [1.1], [1.2], [1.3], Preamble [25]; EX1003, ¶293.

c) **Element [25.2]**

See Grounds I-III, Elements [1.1], [1.3], [1.4], [25.1], [25.2]; EX1003, ¶294.

d) **Element [25.3]**

See Grounds I-III, Elements [1.1], [25.2], [25.3], Claim 4; EX1003, ¶295.

e) **Element [25.4]**

See Grounds I-III, Elements [1.2], [25.3], [25.4]; EX1003, ¶296.

f) **Element [25.5]**

See Grounds I-III, Elements [1.1], [1.3], [25.4], [25.5], Claim 4; EX1003, ¶297.

g) **Element [25.6]**

See Grounds I-III, Elements [1.1], [1.2], [1.3], [25.5], [25.6]; EX1003, ¶298.

h) **Element [25.7]**

See Grounds I-III, Elements [1.1], [1.3], [1.4], [25.6], [25.7]; EX1003, ¶299.

27. Independent Claim 26

a) **Preamble [26]**

See Grounds I-III, Preamble [1]; EX1003, ¶300.

b) **Element [26.1]**

See Ground I-III, Elements [1.2], [26.1]; EX1003, ¶301.

c) **Element [26.2]**

See Ground I-III, Preamble [1], Elements [1.1], [1.2], [26.1], [26.2]; EX1003, ¶302.

d) **Element [26.3]**

See Grounds I-III, Elements [26.2], [26.3], Claim 4; EX1003, ¶303.

e) **Element [26.4]**

See Grounds I-III, Elements [1.1], [1.2], [1.3], [25.3], [25.4], [25.5], [25.6], [26.1], Claim 4; EX1003, ¶304.

28. Independent Claim 27

a) **Preamble [27]**

See Grounds I-III, Preamble [1]; EX1003, ¶305.

b) **Element [27.1]**

See Grounds I-III, Preamble [1], Preamble [27], Element [26.1]; EX1003, ¶306.

c) **Element [27.2]**

See Grounds I-III, Elements [1.1], [1.2], [1.3], [27.1], [27.2]; EX1003, ¶307.

d) **Element [27.3]**

See Grounds I-III, Elements [1.4], [27.2], [27.3]; EX1003, ¶308.

e) **Element [27.4]**

See Grounds I-III, Elements [25.3], [27.3], Claim 4; EX1003, ¶309.

f) **Element [27.5]**

See Grounds I-III, Elements [1.1], [1.2], [1.3], [25.5], [27.4], [27.5], Claim 4. EX1003, ¶310.

29. Independent Claim 28

a) **Preamble [28]**

See Grounds I-III, Preamble [1], Preamble [28]; EX1003, ¶311.

b) **Element [28.1]**

See Grounds I-III, Elements [1.1], [1.2], [28.1]; EX1003, ¶312.

c) **Element [28.2]**

See Grounds I-III, Elements [1.3], [28.1], [28.2]; EX1003, ¶313.

d) **Element [28.3]**

See Grounds I-III, Elements [1.4], [28.2], [28.3]; EX1003, ¶314.

e) **Element [28.4]**

See Grounds I-III, Elements [28.3], [28.4], Claim 4; EX1003, ¶315.

f) **Element [28.5]**

See Grounds I-III, Elements [1.2], [28.4]; EX1003, ¶316.

g) **Element [28.6]**

See Grounds I-III, Elements [1.1], [1.2], [1.3], [25.5], [28.5], Claim 4; EX1003, ¶317.

h) **Element [28.7]**

See Grounds I-III, Elements [1.1], [1.2], [1.3], [28.6], Claim 4. EX1003, ¶318.

i) **Element [28.8]**

See Grounds I-III, Elements [1.1], [1.3], [1.4], [25.7], [28.7], [28.8]; EX1003, ¶319.

**D. Ground IV: Claims 22 and 24 Are Rendered Obvious Over
Handley in Combination with *Basturk* and *PacketCable***

1. Motivation to Combine

Although *Handley* does not disclose the use of switching nodes and switching installations, in addition to the reasons described in Sections X.A.1, X.B.1, and X.C.1, a POSITA would have been motivated to add the media gateway of *PacketCable* to the combination of *Handley* and *Basturk*. EX1003, ¶320. A POSITA would have had a reasonable expectation of success because *PacketCable*, *Handley*, and *Basturk* all involve communications in a digital

environment, in which interfaces between different types of communications networks can be easily created by a POSITA using existing hardware interfaces combined with straightforward software programming techniques to convert between different data formats and protocols. EX1003, ¶320. Gateways were known in the art and a gateway was known “to mean any device that connects two or more dissimilar networks.” EX1009, 418. A POSITA would have understood and found obvious that a media gateway allows for telephony communications between devices on the data network and devices on the Public Switched Telephone Network, such as land lines and cell phones. EX1003, ¶320. Further, a POSITA would have understood and found obvious the benefits of having data network devices communicate with all pre-existing telephones on the PSTN so that the data network devices can conduct a telephone call with users beyond just those on the data network. EX1003, ¶320.

2. Dependent Claim 22

See Grounds I-III, Element [1.2], Claims 21, 22; EX1003, ¶321.

3. Dependent Claim 24

See Grounds I-III, Claim 23; EX1003, ¶322.

See Ground IV, Claim 22; EX1003, ¶323.

XI. SECONDARY CONSIDERATIONS

Petitioners are unaware of any evidence of secondary considerations that would support a finding of non-obviousness. *See* EX1003, ¶324. The asserted prior

art demonstrates there is no evidence of failure by others and that the features recited in the Challenged Claims were readily available in the prior art. In any event, no secondary considerations would overcome the strong *prima facie* case of obviousness shown in this Petition. *See Wyer v. Master Lock Co.*, 616 F.3d 1231, 1246 (Fed. Cir. 2010). Should Patent Owner attempt to meet its burden to show secondary considerations, Petitioners reserve the right to respond.

XII. CONCLUSION

Petitioners request institution of an IPR and cancellation of the Challenged Claims.

Petition for *Inter Partes* Review of U.S. Patent No. 7,266,682

Date: May 24, 2023

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

Attorneys for Petitioners

CERTIFICATE OF COMPLIANCE

This Petition complies with the type-volume limitations as mandated in 37 C.F.R. § 42.24, totaling 13,945 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 1001 through 1015 as listed in Petitioner's Exhibit List on page iv of this document, was served on May 24, 2023, via overnight delivery, upon agreement under 37 CFR § 42.105, directed to the attorney/agent of record for the patent as identified on USPTO Patent Center and the address on record for the current assignee as follows:

STAAS & HALSEY LLP
Suite 700
1201 New York Avenue, N.W.
Washington, DC 20005

VL COLLECTIVE IP LLC
2303 Saint Francis Drive
Palo Alto, CA 94303

A courtesy copy was also served by electronic mail on PO's counsel of record in the following related matter:

VideoLabs, et al v. Meta Platforms, Inc., et al, Case No. 1:22-cv-00680 (D. Del.).

Petition for *Inter Partes* Review of U.S. Patent No. 7,266,682

Brian E. Farnan
Michael J. Farnan
919 N. Market St., 12th Floor
Wilmington, DE 19801
Telephone: (302) 777-0300
Fax: (302) 777-0301
bfarnan@farnanlaw.com
mfarnan@farnanlaw.com

Christine E. Lehman
Brian Baran
REICHMAN JORGENSEN
LEHMAN & FELDBERG LLP
1909 K St, NW, Suite 800
Washington, DC 20006

Jaime F. Cardenas-Navia
Michael Matulewicz-Crowley
REICHMAN JORGENSEN LEHMAN &
FELDBERG LLP
400 Madison Ave, Suite 14D
New York, NY 10017

Taylor N. Mauze
Navid Bayar
REICHMAN JORGENSEN LEHMAN
& FELDBERG LLP
7500 Rialto Blvd., Ste. 1-250
Austin, Texas 78735

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

APPENDIX A

No.	Claim Language
Preamble [1]	A method for transmitting data from a transmitter to a receiver, comprising:
Element [1.1]	providing transmitter-to-receiver authentication at a Real Time Transport Protocol (RTP) packet level as an application protocol on an application layer
Element [1.2]	by inserting, at the transmitter, authentication data at end of a whole RTP packet payload;
Element [1.3]	ascertaining, by the receiver, whether the receiver knows the transmitter based on the RTP packet level authentication data; and
Element [1.4]	accepting, by the receiver, the whole RTP packet payload, if the receiver knows the transmitter, and otherwise rejecting the whole RTP packet payload.
Claim 2	The method as claimed in claim 1, wherein the authentication data are determined based on at least part of a protocol data unit available on the application layer, and the application protocol being linked to a secret by the transmitter.
Claim 3	The method as claimed in claim 2, wherein the secret between the transmitter and the receiver is one of a key for symmetric encryption and a key pair for asymmetric encryption.
Claim 4	The method as claimed in claim 2, wherein the part of the protocol data unit available on the application layer includes at least one of a sequence number and a timestamp.
Claim 5	The method as claimed in claim 2, wherein the secret is an encryption operation performed using one of a one-way hash function and a message authentication code with a key known only to the receiver and the transmitter to be authenticated.
Claim 6	The method as claimed in claim 2, wherein said ascertaining includes cryptographic verification using parameters and including at least one of a decryption operation and a cryptographic checksum check.
Claim 7	The method as claimed in claim 2, wherein said ascertaining includes cryptographic verification using parameters and including at least one of a decryption operation and a

	cryptographic checksum check.
Claim 8	The method as claimed in claim 2, wherein the linking to the secret is carried out by at least one of cryptographical linking to at least one further parameter, particularly by virtue of an encryption operation and a cryptographic checksum and key information.
Claim 9	The method as claimed in claim 8, wherein the secret between the transmitter and the receiver is one of a key for symmetric encryption and a key pair for asymmetric encryption.
Claim 10	The method as claimed in claim 8, wherein the part of the protocol data unit available on the application layer includes at least one of a sequence number and a timestamp.
Claim 11	The method as claimed in claim 8, wherein the encryption operation is performed using one of a one-way hash function and a message authentication code with a key known only to the receiver and the transmitter to be authenticated.
Claim 12	The method as claimed in claim 8, wherein said ascertaining includes verification at the receiver to determine whether the receiver knows the transmitter.
Claim 13	The method as claimed in claim 12, wherein the secret between the transmitter and the receiver is one of a key for symmetric encryption and a key pair for asymmetric encryption.
Claim 14	The method as claimed in claim 12, wherein the part of the protocol data unit available on the application layer includes at least one of a sequence number and a timestamp.
Claim 15	The method as claimed in claim 12, wherein the encryption operation is performed using one of a one-way hash function and a message authentication code with a key known only to the receiver and the transmitter to be authenticated.
Claim 16	The method as claimed in claim 12, wherein the verification is cryptographic verification using parameters and including at least one of a decryption operation and a cryptographic checksum check.
Claim 17	The method as claimed in claim 16, wherein the secret between the transmitter and the receiver is one of a key for symmetric encryption and a key pair for asymmetric encryption.
Claim 18	The method as claimed in claim 17, wherein the part of the protocol data unit available on the application layer includes at least one of a sequence number and a timestamp.

Claim 19	The method as claimed in claim 18, wherein the encryption operation is performed using one of a one-way hash function and a message authentication code with a key known only to the receiver and the transmitter to be authenticated.
Claim 20	The method as claimed in claim 19, wherein the authentication is carried out between the transmitter and the receiver before the data are transmitted from the transmitter to the receiver.
Claim 21	The method as claimed in claim 20, wherein the receiver and transmitter are used in Internet telephony.
Claim 22	The method as claimed in claim 21, wherein the receiver and transmitter are each disposed in at least one of a switching node and a switching installation.
Claim 23	The method as claimed in claim 20, wherein the receiver and transmitter provide packet-switched telephone services.
Claim 24	The method as claimed in claim 23, wherein the receiver and transmitter are each disposed in at least one of a switching node and a switching installation.
Preamble [25]	A system for transmitting data, comprising:
Element [25.1]	a transmitter providing transmitter-to-receiver authentication at a Real Time Transport Protocol (RTP) packet level by inserting authentication data at end of a whole RTP packet payload; and
Element [25.2]	a receiver to determine whether the transmitter is known based on the RTP packet level authentication data, to accept the whole RTP packet payload if the transmitter is known and to reject the whole RTP Packet payload if the transmitter is not known,
Element [25.3]	wherein the whole RTP packet payload comprises a whole RTP packet payload header comprising at least one of a sequence number and a time stamp, and
Element [25.4]	wherein the inserting the authentication data at end of the whole RTP packet payload comprises
Element [25.5]	generating a message authentication code (MAC) according to a common secret between the transmitter and the receiver and using the whole RTP packet payload header sequence number and timestamp,
Element [25.6]	and appending the MAC to the end of the whole RTP packet payload,

Element [25.7]	and if a MAC obtained by a receiver is same as a MAC contained in a transmitted whole RTP packet Payload received by the receiver, the receiver accepts the transmitted whole RTP packet.
Preamble [26]	A transmitter for sending data to a receiver, comprising:
Element [26.1]	a processor providing transmitter-to-receiver authentication at a Real Time Transport Protocol (RTP) packet level by inserting authentication data at end of a whole RTP packet payload; and
Element [26.2]	a transmitter to send the whole RTP packet payload including the inserted authentication data to the receiver,
Element [26.3]	wherein the whole RTP packet payload comprises an RTP header comprising at least one of a sequence number and a time stamp, and
Element [26.4]	wherein the processor inserts the authentication data at the end of the whole RTP packet payload by generating a message authentication code (MAC) according to a common secret between the transmitter and the receiver and using the whole RTP packet payload header sequence number and timestamp, and appending the MAC to the end of the whole RTP packet payload.
Preamble [27]	A receiver for receiving data from a transmitter, comprising:
Element [27.1]	a processing unit communicably connecting with the transmitter,
Element [27.2]	determining whether the transmitter is known based on the transmitter providing transmitter-to-receiver authentication at a Real Time Transport Protocol (RTP) packet level by inserting authentication data at end of a whole RTP packet payload, and
Element [27.3]	accepting the whole RTP packet payload if the transmitter is known, and rejecting the whole RTP packet payload if the transmitter is not known, based upon the RTP packet level authentication data,
Element [27.4]	wherein the whole RTP packet payload comprises a whole RTP packet payload header comprising at least one of a sequence number and a time stamp, and
Element [27.5]	wherein the processing unit obtains the authentication data added by the transmitter to the end of the whole RTP packet payload by generating a message authentication code (MAC) according to a common secret between the transmitter and the receiver and using the whole RTP packet payload header sequence number and timestamp.

Preamble [28]	A method of transmitting data from a transmitter to a receiver, comprising:
Element [28.1]	providing transmitter-to-receiver authentication at a Real Time Transport Protocol (RTP) packet level by inserting authentication data at end of a whole RTP packet payload;
Element [28.2]	ascertaining, by the receiver, whether the receiver knows the transmitter based on the RTP packet level authentication data; and
Element [28.3]	accepting the whole RTP packet Payload at the receiver, if the receiver knows the transmitter, and otherwise rejecting the whole RTP Packet payload,
Element [28.4]	wherein the whole RTP packet payload comprises a whole RTP Packet payload header comprising at least one of a sequence number and a time stamp,
Element [28.5]	and wherein the inserting the authentication data at the end of the whole RTP packet payload
Element [28.6]	comprises generating a message authentication code (MAC) according to a common secret between the transmitter and the receiver and using the whole RTP packet payload header sequence number and timestamp,
Element [28.7]	and appending the MAC to the end of the whole RTP packet payload,
Element [28.8]	and if a MAC obtained by a receiver is same as a MAC contained in a transmitted whole RTP packet payload received by the receiver, the receiver accepts the transmitted whole RTP packet payload.