

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

GOOGLE LLC,
Petitioner,

v.

HEADWATER RESEARCH LLC,
Patent Owner.

Case No. IPR2026-00203
Patent No. 9,232,403

DECLARATION OF DR. PATRICK G. TRAYNOR

Google Exhibit 1003 Google v. Headwater
--

TABLE OF CONTENTS

I.	PERSONAL AND PROFESSIONAL BACKGROUND	1
II.	MATERIALS REVIEWED AND CONSIDERED	5
III.	MY UNDERSTANDING OF PATENT LAW	8
	A. Anticipation.....	10
	B. Obviousness.....	11
IV.	PERSON OF ORDINARY SKILL IN THE ART (“POSA”).....	13
V.	THE ’403 PATENT.....	15
	A. Brief Description.....	15
	B. Prosecution History of the ’403 Patent.....	16
	C. Grounds for Unpatentability.....	17
VI.	CLAIM INTERPRETATION	19
VII.	<u>GROUND 1</u> : TS-23.140 AND OGAWA RENDER OBVIOUS CLAIMS 1, 3-6, AND 11-21.....	20
	A. TS-23.140 (EX-1004).....	20
	B. Ogawa (EX-1005)	24
	C. The MMS-Ogawa Combination.....	29
	1. Implementing TS-23.140’s User Device with a Modem	30
	2. Securing Interface MM1 Using SSL/TLS.....	35
	3. Applying Ogawa’s Symmetric Encryption Techniques For Network Communications	38
	4. Ogawa’s Decryption and Encryption Units	45
	5. Implementing TS-23.140’s User Device with an Interprocess Communication Bus.....	48
	6. Securing Interprocess Communications Within the Device	52
	7. MMS-Ogawa.....	56
	D. Claim Analysis	57
	1. Claim 1	57
	a. [1 PRE] “A mobile end-user-area device comprising:”	57

b.	[1A] “a wireless wide-area network (WWAN) modem to exchange Internet data via a connection to a first WWAN, when configured for and connected to the first WWAN;”	58
c.	[1B1] “a device messaging agent to receive secure Internet data messages, on behalf of a plurality of software applications capable of execution on the device, and over a secure connection to a network message server reachable via the WWAN,”	64
	i. “a device messaging agent to receive secure Internet data messages”	64
	ii. “on behalf of a plurality of software applications capable of execution on the device”	69
	iii. “and over a secure connection to a network message server reachable via the WWAN”	72
d.	[1B2] “wherein at least a subset of the secure Internet data messages contain an identifier for a corresponding one of the software applications and application data from a respective network application server corresponding to that application; and”	76
	i. “wherein at least a subset of the secure Internet data messages contain an identifier for a corresponding one of the software applications”	76
	ii. “and application data from a respective network application server corresponding to that application”	79
e.	[1C1] “a secure interprocess communication service,”	85
f.	[1C2] “wherein the device messaging agent, for each message in the subset of the secure Internet data messages, maps the identifier to the corresponding one of the software applications in order to forward the application data on the secure interprocess communication service to a software process corresponding to the identified software application.”	89
	i. “wherein the device messaging agent, for each message in the subset of the secure Internet data messages, maps the identifier to the corresponding one of the software applications”	89

ii.	“in order to forward the application data on the secure interprocess communication service”	91
iii.	“to a software process corresponding to the identified software application.”	92
2.	Claim 3: The mobile end-user device of claim 1, further comprising the plurality of software applications.	94
3.	Claim 4: The mobile end-user device of claim 3, wherein the plurality of applications include a first application that receives the application data in a first format, and a second application that receives the application data in a second format different than the first format.	94
4.	Claim 5: The mobile end-user device of claim 1, wherein the secure Internet data messages are received encrypted, the device messaging agent decrypting each message in the subset to obtain the corresponding identifier and application data.	95
5.	Claim 6: The mobile end-user device of claim 5, wherein the secure Internet data messages are transported to the device messaging agent using one or more of encryption on a transport services stack, IP (Internet Protocol) layer encryption, and transport via a tunnel.	97
6.	Claim 11: The mobile end-user device of claim 1, wherein the device messaging agent is further to send secure upload Internet data messages to the network message server over the secure connection, wherein at least a subset of the secure upload Internet data messages are sent responsive to a corresponding request received on the secure interprocess communication channel from a corresponding one of the software applications, the device messaging agent constructing from each such request a secure upload Internet data message containing an identifier for a respective network application server corresponding to the requesting software application; and content received with the request.	98
7.	Claim 12: The mobile end-user device of claim 11, wherein at least one of the upload Internet data messages comprises a key for the network application server corresponding to the requesting software application.	105

8. Claim 13: The mobile end-user device of claim 1, wherein the device messaging agent creates a log for the received secure Internet data messages.....	107
9. Claim 14: The mobile end-user device of claim 1, wherein the secure interprocess communication channel and the secure connection to the network message server are separately secured.....	109
10. Claim 15: The mobile end-user device of claim 1, wherein access by the software applications to the interprocess communication channel is subject to a security policy.....	110
11. Claim 16: The mobile end-user device of claim 1, wherein at least one of the secure Internet data messages comprises multiple identifier/data pairs.	111
12. Claim 17: The mobile end-user device of claim 1, the device messaging agent comprising an agent router to forward the application data on the secure interprocess communication channel to the software process corresponding to the identified software application.	114
13. Claim 18: The mobile end-user device of claim 1, wherein the secure interprocess communication service forwards the application data to at least one of the software processes in an encrypted format.	116
14. Claim 19: The mobile end-user device of claim 1, the device messaging agent further to initiate the secure connection to the network message server.	116
15. Claim 20: The mobile end-user device of claim 1, further comprising a network stack in communication with the device messaging agent and the WWAN modem, the secure connection terminated within the network stack.....	117
16. Claim 21: The mobile end-user device of claim 1, wherein at least one of the applications and the network application server corresponding to that application authenticate with each other prior to passing application data via the device messaging agent and network message server.....	120
VIII. <u>FOUNDATIONS 2A-2C</u>	122
A. GROUND 2A: MMS-Ogawa in View of Cole (EX-1006).....	123

1.	Implementing a WWAN Modem in View of Cole (Claims 1, 3-6, 11-21)	123
2.	Adding a WLAN Modem in View of Cole (Claim 2)	125
	a. Claim 2: The mobile end-user device of claim 1, further comprising a wireless local area network (WLAN) modem to exchange Internet data via a connection to a first WLAN, when configured for and connected to the first WLAN, the device messaging agent further to receive secure Internet data messages over a secure connection via the first WLAN to the network message server.....	133
	B. GROUND 2B: MMS-Ogawa in View of Sathish (EX-1031).....	134
	C. GROUND 2C: MMS-Ogawa-Cole-Sathish	135
IX.	<u> GROUNDS 3A-3D</u>	136
	A. Papineau (EX-1017).....	136
	B. GROUND 3A: Implementing Papineau’s JAM Teachings (Claims 7-10)	139
	1. MMS-Ogawa-Papineau.....	139
	2. Claim 7: The mobile end-user device of claim 1, wherein at least one of the applications comprises a service downloader, the service downloader authenticating a downloaded software application based on application data from at least one of the secure Internet data messages.	143
	3. Claim 8: The mobile end-user device of claim 7, wherein the service downloader is to download the downloaded software application using an Internet data connection other than the secure connection used for the secure Internet data messages.....	145
	4. Claim 9: The mobile end-user device of claim 7, wherein the downloaded software application comprises an updated version of the device messaging agent.....	146
	C. GROUNDS 3B-3D.....	147
X.	<u> GROUNDS 4A-4D</u>	148
	A. Ellison (EX-1019)	148
	B. GROUND 4A: Implementing Ellison’s Isolated Execution Environment (Claim 10).....	152

1. MMS-Ogawa-Ellison	152
2. Claim 10: The mobile end-user device of claim 1, wherein the device messaging agent runs in a secure execution environment on the device, and at least one of the applications runs outside of the secure execution environment on the device.	156
C. GROUNDS 4B-4D.....	157
XI. CLAIM LISTING APPENDIX.....	159

I, Patrick Gerard Traynor, declare:

1. I have been retained by Wolf, Greenfield & Sacks, P.C., counsel for Petitioner Google LLC to assess claims 1-21 (the “challenged claims”) of U.S. Patent No. 9,232,403 (“the ’403 patent”). My compensation is not dependent in any way upon the outcome of the *inter partes* review of the ’403 patent.

I. PERSONAL AND PROFESSIONAL BACKGROUND

2. I earned a B.S. in Computer Science from the University of Richmond in 2002 and an M.S. and Ph.D. in Computer Science and Engineering from the Pennsylvania State University in 2004 and 2008, respectively. My dissertation, entitled “Characterizing the Impact of Rigidity on the Security of Cellular Telecommunications Networks,” focused on security problems that arise in cellular infrastructure when gateways to the broader Internet were created.

3. I am currently a Professor in the Department of Computer and Information Science and Engineering (CISE) at the University of Florida. I was hired under the “Rise to Preeminence” Hiring Campaign and serve as the Interim Chair for my Department. I also hold the endowed position of the John and Mary Lou Dasburg Preeminent Chair in Engineering.

4. Prior to joining the University of Florida, I was an Associate Professor from March to August 2014 and an Assistant Professor of Computer Science from

2008 to March 2014 at the Georgia Institute of Technology. I have supervised many Ph.D., M.S., and undergraduate students during the course of my career.

5. My area of expertise is security, especially as it applies to mobile systems and networks, including cellular networks and messaging. As such, I regularly teach students taking my courses and participating in my research group to program and evaluate software and architectures for mobile and cellular systems. I have taught courses on the topics of network and systems security, cellular networks, and mobile systems at both Georgia Tech and the University of Florida. I also advised and instructed the Information Assurance Officer Training Program for the United States Army Signal Corps in the Spring of 2010. My PhD dissertation concentrated on security issues in messaging for cellular systems.

6. I have received numerous awards for research and teaching, including being named a Kavli Fellow (2017), a Fellow of the Center for Financial Inclusion (2016), and a Research Fellow of the Alfred P. Sloan Foundation (2014). I also won the Lockheed Inspirational Young Faculty Award (2012), was awarded a National Science Foundation (NSF) CAREER Award (2010), and received the Center for Enhancement of Teaching and Learning at Georgia Tech's "Thanks for Being a Great Teacher" Award (2009, 2012, 2013).

7. I have published over 100 articles in top conferences and journals in the areas of information security, mobile systems, and networking. Many of my

results are highly cited, and I have received multiple “Best Paper” Awards. I have also written a book entitled “Security for Telecommunications Networks”, which is used in wireless and cellular security courses at a number of top universities.

8. I am a Senior Member of the Association for Computing Machinery (ACM) and the Institute of Electrical and Electronics Engineers (IEEE). I am also a member of the USENIX Advanced Computing Systems Association.

9. I have served as an Associate Editor for IEEE Security and Privacy Magazine, have been the Program Chair for eight conferences and workshops, and have served as a member of the Program Committee for over 50 different conferences and workshops. I also previously served as the Security Subcommittee Chair for the ACM US Technology Policy Committee (USACM).

10. I was a co-Founder and Research Fellow for a private start-up, Pindrop Security, from 2012 to 2014. Pindrop provides anti-fraud and authentication solutions for Caller-ID spoofing attacks in enterprise call centers by creating and matching acoustic fingerprints. Pindrop Security currently employs over 200 people, and their technology is based off of my research (US Patent 9,037,113 B2).

11. I was a co-Founder and Chief Executive of a private start-up, CryptoDrop. CryptoDrop developed a ransomware detection and recovery tool to

provide state of the art protection to home, small business, and enterprise users.

This technology was also based off of my research (US Patent 10,685,114 B2).

12. I was also a co-Founder and Chief Executive of a private start-up, Skim Reaper. Skim Reaper developed tools to detect credit card skimming devices, and worked with a range of banks, international law enforcement, regulators, and retailers. This technology was also based off of my research (US Patent 10,496,914 B2).

13. I am a named inventor on more than ten US patents. These patents detail methods for determining the origin and path taken by phone calls as they traverse various networks, cryptographically authenticating phone calls, providing a secure means of indoor localization using mobile/wireless devices, detecting credit card skimmers, identifying cloned credit cards, blocking ransomware from encrypting data, and more.

14. My curriculum vitae, included with this declaration, includes a list of publications on which I am a named author. It contains further details regarding my experience, education, publications, and other qualifications to render an expert opinion in connection with this proceeding.

15. My curriculum vitae is provided as Appendix A.

II. MATERIALS REVIEWED AND CONSIDERED

16. My findings, as explained below, are based on my years of education, research, experience, and background in the field of services and application implementation in communication networks, as well as my investigation and study of relevant materials for this declaration. When developing the opinions set forth in this declaration, I assumed the perspective of a person having ordinary skill in the art (“POSA”), as set forth in Section IV below. In forming my opinions, I have studied and considered the materials identified in the list below.

Exhibit	Description
1001	U.S. Patent No. 9,232,403 (“the ’403 Patent”)
1002	Prosecution History of U.S. Patent No. 9,232,403 (“the ’403 FH”)
1004	3GPP TS 23.140 v6.9.0 (2005-03); 3rd Generation Partnership Project; Technical Specification Group Terminals; Multimedia Messaging Service (MMS); Functional Description; Stage 2 (“TS-23.140”)
1005	U.S. Patent No. 8,195,961 (“Ogawa”)
1006	U.S. Patent App. Pub. No. 2008/0080458 (“Cole”)
1007	U.S. Patent App. Pub. No. 2004/0111476 (“Trossen”)
1008	PCT Pub. No. 2008/048075 (“Lee”)
1009	U.S. Patent No. 7,975,147 (“Qumei”)
1010	U.S. Patent No. 9,032,192 (“Frank”)
1011	Open Mobile Alliance; OMA-ERELD-MMS-v1_2-20030923-C, Enabler Release Definition for MMS Version 1.2,” available at https://www.openmobilealliance.org/release/MMS/V1_2-20030923-C/OMA-ERELD-MMS-V1_2-20030923-C.pdf
1012	“Open Mobile Alliance; Multimedia Messaging Service Architecture Overview” (MMSARCH) specification, available at https://www.openmobilealliance.org/release/MMS/V1_2-20030923-C/OMA-MMS-ARCH-V1_2-20030920-C.pdf
1013	The Secure Sockets Layer (“SSL”) Protocol, V. 3.0, available at https://web.archive.org/web/19970614041044/http://home.netscape.com/eng/ssl3/ssl-toc.html and

	https://web.archive.org/web/19970617034012/http://home.netscape.com/eng/ssl3/3-SPEC.HTM#1
1014	The Transport Layer Security (“TLS”) Protocol, V. 1.1, available at https://datatracker.ietf.org/doc/html/rfc4346.html
1015	U.S. Patent App. Pub. No. 2003/0096625 (“Mi-Su Lee”)
1016	3GPP TS-23.234 v8.0.0 (2008-12); 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; 3GPP system to Wireless Local Area Network (WLAN) interworking; System description (Release 8) (“TS-23.234”)
1017	U.S. Patent No. 7,779,408 (“Papineau”)
1018	Liaison Statement, European Telecommunications Standards Institute AT-F Rapporteur Meeting, 4 to 6 February 2003 (ETSI / AT-F TD18), available at https://www.3gpp.org/ftp/tsg_sa/TSG_SA/TSGS_19/Docs/PDF/SP-030167.pdf
1019	U.S. Patent No. 7,082,615 (“Ellison”)
1020	3GPP TS-26.140 v6.2.0 (2005-03); 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Multimedia Messaging Service(MMS); Media formats and codecs
1021	Multimedia Messaging Service Encapsulation Protocol, available at www.openmobilealliance.org/release/MMS/V1_2-20050301-A/OMA-MMS-ENC-V1_2-20050301-A.pdf
1022	<i>Samsung Elecs. et al. v. Headwater Research LLC</i> , IPR2024-00341, Paper 28 (July 23, 2025)
1024	Declaration of Friedhelm Rodermund
1025	Mobile Information Device Profile for Java™ 2 Micro Edition Version 2.0 (Nov. 5, 2002) (“MIDP-Specification”)
1026	U.S. Patent App. Pub. No. 2005/0108571 (“Lu”)
1027	U.S. Patent App. Pub. No. 2005/0207379 (“Shen”)
1028	Transporting data between wireless applications using a messaging system—MMS, Miraj E Mostafa, <i>Wireless Communications and Mobile Computing</i> (2007) (“Mostafa”)
1029	Dictionary of Computer Science, Engineering, and Technology, CRC Press LLC, 2001
1030	Needham et al., “Using Encryption for Authentication in Large Networks of Computers” (ACM, Vol. 21, No. 12, Dec. 1978) (“Needham”)
1031	U.S. Patent No. 8,010,669 (“Sathish”)

1032	Saltzer et al., “The Protection of Information in Computer Systems” (IEEE Proceedings, Vol. 63, No. 9, Sept. 1975) (“Saltzer”)
1033	Li et al., “Symbian OS platform security model,” available at https://www.usenix.org/system/files/login/articles/73507-li.pdf (Login Magazine, Aug. 2010)
1034	Philip Zimmermann, “Pretty Good Privacy: RSA Public Key Cryptography for the Masses” PGP User’s Guide. Version 1.0, June 1991, available at https://www.techinsider.org/free-software/research/acrobat/910605.pdf (“Zimmerman”)
1035	B. Ramsdell, S/MIME Version 3 Message Specification, IETF RFC 2633, June 1999, available at https://datatracker.ietf.org/doc/html/rfc2633 (“Ramsdell”)
1036	Schroeder et al., “A Hardware Architecture for Implementing Protection Rings” (ACM, Vol. 15, No. 3, Mar. 1972) (“Schroeder”)
1037	Nokia E71 review: Nokia E71, available at https://www.cnet.com/reviews/nokia-e71-review/
1038	<i>Samsung Elecs. et al. v. Headwater Research LLC</i> , IPR2024-00341, Paper 4 (January 23, 2024)
1039	U.S. Patent App. Pub. No. 2007/0283170 (“Yami”)
1040	U.S. Patent App. Pub. No. 2008/0215883 (“Fok”)
1041	U.S. Patent App. Pub. No. 2003/0220835 (“Barnes”)
1042	U.S. Patent App. Pub. No. 2006/0154699 (“Ko”)
1043	U.S. Patent App. Pub. No. 2005/0207379 (“Shen”)
1044	U.S. Patent App. Pub. No. 2006/0025133 (“Shaheen”)
1045	Dictionary of Computing, S.M.H. Collin, Fifth Edition, Bloomsbury (2004)
1046	IEEE 100 The Authoritative Dictionary of IEEE Standards Terms, Seventh Edition (2000)
1047	U.S. Patent No. 7,962,798 (“Locasto”)
1048	U.S. Patent App. Pub. No. 2004/0002974 (“Kravitz”)
1049	Computer Desktop Encyclopedia, Alan Freedman, Ninth Edition, Osborne/McGraw-Hill (2001)
1050	U.S. Patent No. 5,612,866 (“Savanyo”)
1051	3GPP TS 25.321 v6.5.0 (2005-06); 3rd Generation Partnership Project; Technical Specification Group Radio Access Network; Medium Access Control (MAC) protocol specification; (“TS-25.321”)
1052	3GPP TS 25.322 v6.4.0 (2005-06); 3rd Generation Partnership

	Project; Technical Specification Group Radio Access Network; Radio Link Control (RLC) protocol specification; (“TS-25.322”)
1053	3GPP TS 25.323 v5.4.0 (2005-06); 3rd Generation Partnership Project; Technical Specification Group Radio Access Network; Packet Data Convergence Protocol (PDCP) specification; (“TS-25.323”)
1054	Lu et al., Heading for Multimedia Message Service in 3G, 6th IEE International Conference on 3G and Beyond, Washington, D.C., USA, Nov. 7-9, 2005
1055	RFC 4355, IANA Registration for Enumservices Email, Fax, MMS, EMS, and SMS (Jan. 2006)
1056	Rodermund, A Picture Speaks a Thousand Words – From SMS to MMS, in Business Briefing: Wireless Technology (2003)
1057	RFC 3164, The BSD syslog Protocol (Aug. 2001)

III. MY UNDERSTANDING OF PATENT LAW

17. In developing my opinions, I discussed various relevant legal principles with Petitioner’s attorneys. Though I do not purport to have prior knowledge of such principles, I understood them when they were explained to me and have relied upon such legal principles, as explained to me, in the course of forming the opinions set forth in this declaration. My understanding in this respect is as follows:

18. I understand that “*inter partes* review” (IPR) is a proceeding before the United States Patent & Trademark Office for evaluating the patentability of an issued patent’s claims based on prior-art patents and printed publications.

19. I understand that, in this proceeding, Petitioner has the burden of proving that the challenged claims of the ’403 patent are unpatentable by a

preponderance of the evidence. I understand that “preponderance of the evidence” means that a fact or conclusion is more likely true than not true.

20. I understand that, in IPR proceedings, claim terms in a patent are given their ordinary and customary meaning as understood by a POSA in the context of the entire patent and the prosecution history pertaining to the patent. If the specification provides a special definition for a claim term that differs from the meaning the term would otherwise possess, the specification’s special definition controls. If a claim element is expressed as a “means” for performing a specified function, I understand that it covers the corresponding structure described in the specification and equivalents of the described structure. I have applied these standards in preparing the opinions in this declaration.

21. I understand that determining whether a particular patent or printed publication constitutes prior art to a challenged patent claim can require determining the effective filing date (also known as the priority date) to which the challenged claim is entitled. I understand that for a patent claim to be entitled to the benefit of the filing date of an earlier application to which the patent claims priority, the earlier application must have described the claimed invention in sufficient detail to convey with reasonable clarity to the POSA that the inventor had possession of the claimed invention as of the earlier application’s filing date. I understand that a disclosure that merely renders the claimed invention obvious is

not sufficient written description for the claim to be entitled to the benefit of the filing date of the application containing that disclosure.

22. I understand that for an invention claimed in a patent to be patentable, it must be, among other things, new (novel—*i.e.*, not anticipated) and not obvious from the prior art. My understanding of these two legal standards is set forth below.

A. Anticipation

23. I understand that, for a patent claim to be “anticipated” by the prior art (and therefore not novel), each and every limitation of the claim must be found, expressly or inherently, in a single prior-art reference. I understand that a claim limitation is disclosed for the purpose of anticipation if a POSA would have understood the reference to disclose the limitation based on inferences that a POSA would reasonably be expected to draw from the explicit teachings in the reference when read in light of the POSA’s knowledge and experience.

24. I understand that a claim limitation is inherent in a prior art reference if that limitation is necessarily present when practicing the teachings of the reference, regardless of whether a person of ordinary skill recognized the presence of that limitation in the prior art.

B. Obviousness

25. I understand that a patent claim may be unpatentable if it would have been obvious in view of a single prior-art reference or a combination of prior-art references.

26. I understand that a patent claim is obvious if the differences between the subject matter of the claim and the prior art are such that the subject matter as a whole would have been obvious to a person of ordinary skill in the relevant field at the time the invention was made. Specifically, I understand that the obviousness question involves a consideration of:

- the scope and content of the prior art;
- the differences between the prior art and the claims at issue;
- the knowledge of a person of ordinary skill in the pertinent art; and
- if present, objective factors indicative of non-obviousness, sometimes referred to as “secondary considerations.” I have not been made aware of any secondary considerations asserted by Patent Owner (“PO”) with respect to the ’403 patent.

27. I understand that in order for a claimed invention to be considered obvious, a POSA must have had a reason for combining teachings from multiple prior-art references (or for altering a single prior-art reference, in the case of obviousness in view of a single reference) in the fashion proposed.

28. I further understand that in determining whether a prior-art reference would have been combined with other prior art or with other information within the knowledge of a POSA, the following are examples of approaches and rationales that may be considered:

- combining prior-art elements according to known methods to yield predictable results;
- simple substitution of one known element for another to obtain predictable results;
- use of a known technique to improve similar devices in the same way;
- applying a known technique to a known device ready for improvement to yield predictable results;
- applying a technique or approach that would have been “obvious to try,” *i.e.*, choosing from a finite number of identified, predictable solutions, with a reasonable expectation of success.
- known work in one field of endeavor may prompt variations of it for use in either the same field or a different one based on design incentives or other market forces if the variations would have been predictable to one of ordinary skill in the art;
- some teaching, suggestion, or motivation in the prior art that would have led one of ordinary skill to modify the prior-art reference or

to combine prior-art reference teachings to arrive at the claimed invention. I understand that this teaching, suggestion or motivation may come from a prior-art reference or from the knowledge or common sense of one of ordinary skill in the art.

29. I understand that for a single reference or a combination of references to render the claimed invention obvious, a POSA must have been able to arrive at the claimed invention by altering or combining the applied references.

IV. PERSON OF ORDINARY SKILL IN THE ART (“POSA”)

30. I have been informed and understand that for purposes of assessing whether prior-art references disclose every element of a patent claim (thus “anticipating” the claim) and/or would have rendered the claim obvious, the patent and the prior-art references must be assessed from the perspective of a person having ordinary skill in the art (“POSA”) to which the patent is related, based on the understanding of that person at the time of the patent claim’s priority date. I have been informed and understand that a POSA is presumed to be aware of all pertinent prior art and the conventional wisdom in the art, and is a person of ordinary creativity. I have applied this standard throughout my declaration.

31. The ’403 patent involves technology in the field of services and application implementation in communication networks. I have been asked to provide my opinions as to the state of the art in this field by early 2009. I use this

timeframe because the face of the '403 patent indicates an earliest claimed priority date of January 28, 2009. Whenever I offer an opinion in this declaration about the knowledge of a POSA, the manner in which a POSA would have understood the claims of the '403 patent or its description, the manner in which a POSA would have understood the prior art, or what a POSA would have been led to do based on the prior art, I am referencing the early 2009 timeframe, even if I do not say so specifically in each case.

32. I understand that the PO may attempt to prove that the alleged invention recited in the challenged claims was conceived at some time prior to the earliest claimed priority date on the face of the patent. At the time of this declaration, I am unaware of the PO having alleged any earlier conception date or produced any evidence to establish any earlier conception date.

33. In my opinion, a person of ordinary skill in the art in the early 2009 timeframe ("POSA") would have had (1) at least a bachelor's degree in computer science, electrical engineering, or a related field, and (2) 3-5 years of experience in services and application implementation in communication networks. More education could substitute for experience, and vice versa. This person would have been capable of understanding and applying the teachings of the '403 patent and the prior-art references discussed in this declaration.

34. By early 2009, I had a Ph.D. in Computer Science and Engineering and had seven years of experience in services and application implementation in communication networks beyond my Bachelor's degree, and was therefore a person of more than ordinary skill in the art. Moreover, based on my experiences, I have a good understanding of the capabilities of a POSA because I have taught, participated in organizations with, and worked closely with many people who fit the characteristics of a POSA over the course of my career, and I am familiar with their level of skill. When developing the opinions set forth in this declaration, I assumed the perspective of a POSA, as set forth above.

V. THE '403 PATENT

A. Brief Description

35. The '403 Patent is directed to a "device" that includes a "*device messaging agent*" and a plurality of "*application[s] on the device.*" EX-1001, Abstract. The device messaging agent "securely communicates with a *network message server* over a wireless network." EX-1001, Abstract. The device messaging agent also communicates with the plurality of applications on the device using a "*secure interprocess communication service.*" *Id.*

36. In the Challenged Claims, the device messaging agent receives, from the network message server, “secure Internet data messages” (Element [1B1]¹) with a “subset” of the messages including “application data” from a “network application server” and an “identifier” corresponding to one of the device’s applications (Element [1B2]). Using the identifier, the device messaging agent forwards the application data on the secure interprocess communication service to a process corresponding to the identified application on the device (Elements [1C1]-[1C2]).

B. Prosecution History of the ’403 Patent

37. The continuation application that led to the ’403 Patent (application no. 14/667,353, “the ’353 application”) was filed March 24, 2015. The ’353 application claimed priority to multiple provisional applications. EX-1001, page 2 (field (60) under “Related U.S. Application Data”). The earliest of those provisional applications was filed on January 28, 2009 (“Critical Date”). PO preliminarily amended the ’353 application to replace the title and abstract and add new claims one day after the continuation application was filed, on March 25, 2015. EX-1002, 364-370.

¹ In this declaration, I identify limitations using the reference labels from the Claim Appendix in Section XI below.

38. During prosecution, the examiner rejected the claims “as being indefinite for failing to particularly point out and distinctly claim the subject matter” that the inventor “regards as the invention.” EX-1002, 616. The examiner identified several terms in the independent claim, including “first WWAN,” “device messaging agent,” “on behalf of a plurality of software applications,” “network message server,” “identifier,” and “secure interprocess communication service.” EX-1002, 616-619.

39. To traverse this rejection, PO attempted to identify specific examples (EX-1002, 730-37) for each missing term “to illustrate that those skilled in the art have the material at hand to ascertain what is meant by the various terms....” EX-1002, 730. I use the phrase “attempted to identify” because the language PO quoted in its remarks during prosecution are often attributed to the wrong paragraph number in the as-filed application. I considered the applicant’s prosecution statements in my analysis, and they are cited and discussed in the analysis below. Because PO’s citation errors pervade PO’s remarks during prosecution, where possible, my analysis focuses on the disclosure language that PO actually quoted, instead of PO’s misidentified paragraph numbers.

C. Grounds for Unpatentability

40. The table below identifies each of the obviousness grounds I discuss in this declaration.

Ground Number and Reference(s)		Claims	Basis
1	TS-23.140 (EX-1004), Ogawa (EX-1005)	1, 3-6, 11-21	103
2A	TS-23.140, Ogawa, Cole (EX-1006)	1-6, 11-21	103
2B	TS-23.140, Ogawa, Sathish (EX-1031)	1, 3-6, 11-21	103
2C	TS-23.140, Ogawa, Cole, Sathish	1-6, 11-21	103
3A	TS-23.140, Ogawa, Papineau (EX-1017)	7-9	103
3B	TS-23.140, Ogawa, Cole, Papineau	7-9	103
3C	TS-23.140, Ogawa, Sathish, Papineau	7-9	103
3D	TS-23.140, Ogawa, Cole, Sathish, Papineau	7-9	103
4A	TS-23.140, Ogawa, Ellison (EX-1019)	10	103
4B	TS-23.140, Ogawa, Cole, Ellison	10	103
4C	TS-23.140, Ogawa, Sathish, Ellison	10	103
4D	TS-23.140, Ogawa, Cole, Sathish, Ellison	10	103

41. I am informed and understand that each reference in the Grounds qualifies as prior art to each Challenged Claim even if PO could establish entitlement to an effective filing date of January 28, 2009, on at least the bases shown below:

Reference	Filed	Published	Prior Art Basis
TS-23.140		March 2005	102(b)
Ogawa	5/19/2008	10/1/2009	102(a)/102(e)
Cole	9/29/2006	4/3/2008	102(a)/102(e)
Sathish	10/15/2008	8/30/2011	102(a)/102(e)
Papineau	1/21/2004	8/17/2010	102(a)/102(e)
Ellison	9/22/2000	7/25/2006	102(b)

42. I have been asked to provide my opinion concerning whether claims 1-21 of the '403 patent are anticipated and/or would have been obvious to a POSA in light of the prior-art references identified in the Petition. For the reasons explained below, it is my opinion that each of claims 1-21 is anticipated and/or would have been obvious to a POSA.

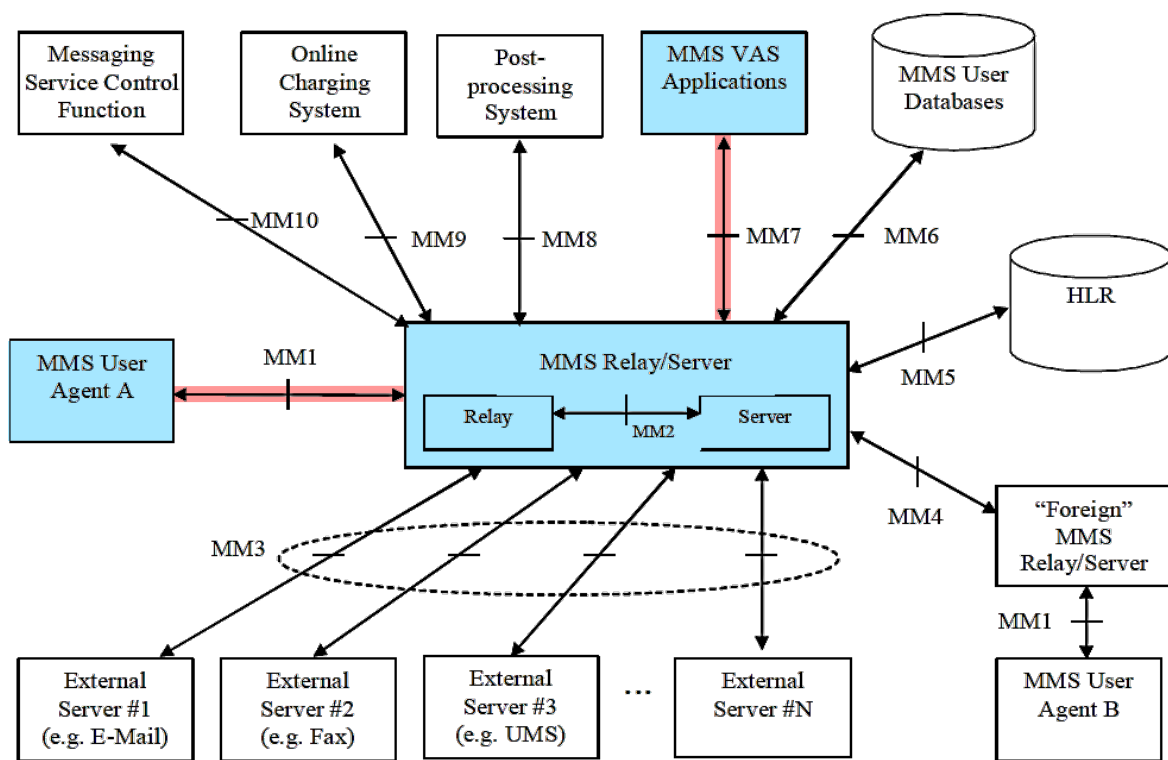
VI. CLAIM INTERPRETATION

43. I have been informed by Counsel and understand that the best indicator of claim meaning is its usage in the context of the patent specification as understood by one of ordinary skill. I further understand that the words of the claims should be given their plain meaning unless that meaning is inconsistent with the patent specification or the patent's history of examination before the Patent Office. Counsel has also informed me, and I understand that, the words of the claims should be interpreted as they would have been interpreted by one of ordinary skill as of the Critical Date (January 28, 2009). I have been informed by Counsel that I should use the Critical Date as the point in time for claim interpretation purposes.

VII. GROUND 1: TS-23.140 AND OGAWA RENDER OBVIOUS CLAIMS 1, 3-6, AND 11-21

A. TS-23.140 (EX-1004)

44. TS-23.140 is a standard describing a “non-realtime Multimedia Messaging Service, MMS.” EX-1004, 10. One MMS implementation environment is shown below, with annotations highlighting some portions of the environment:



EX-1004, 23, FIG. 3 (annotated)

45. In annotated FIG. 3 above, “MMS User Agent A” (blue box at left) is an “application residing on a UE [user equipment]... or... external device” that “performs MMS-specific operations on a user’s behalf and/or on another application’s behalf.” *Id.*, 14, 18-19. Those operations include, for example, retrieving multimedia messages (MMs) by “initiat[ing] MM delivery to the MMS

User Agent,” as well as generally “hand[ling] MMs (e.g. submitting, receiving, deleting of MMs).” EX-1004, 18-19.

46. The MMS Relay/Server (blue box in center in annotated FIG. 3 above) relays messages from the network to the MMS User Agent using interface (MM1, red). EX-1004, 17-18 (describing MMS architectural elements and FIG. 2), 21 (Section 5.2 describing MMS Relay/Server), 23-25 (Section 6 describing MMSE Architecture and Interfaces and FIGS. 3-4). The messages may include “MMS VAS” (Value Added Services) content from “MMS VAS Applications,” “provided... by third-party Value Added Service Providers (VASP)” via interface MM7 (shown in red) (*id.*, 14, 18, 23), and then relayed to the MMS User Agent to “provid[e] Value Added Services (e.g. news service or weather forecasts) to MMS users.” *Id.*, 14, 18, 23, 41. The MMS Relay/Server can relay messages from “*several* MMS VAS Applications”² in the network. *Id.*, 18.

47. “MMS may... be used to transport data specific to applications” “other than the MMS User Agent...” which also “reside on [the] MMS User Agent [device].” EX-1004, 14, 54-56. I refer to the MMS User Agent “device” here in view of the entirety of TS-23.140’s disclosures—including TS-23.140’s definitions of “MMS User Agent” as an “application residing on a UE, an MS or an external

² Emphasis is added throughout unless otherwise indicated.

device...” and of “Application Data” as “[i]nformation / data specific to an application other than the MMS User Agent...” EX-1004, 14. A POSA understood that when TS-23.140 discusses transporting data specific to applications that “*reside on* an MMS User Agent” (EX-10014, 54-56), the term “MMS User Agent” in that context refers to the client UE / MS / device.

48. The fact that the term “MMS User Agent” was sometimes used in this manner to refer to a client device with an MMS User Agent application on it is confirmed by contemporaneous references describing MMS. *E.g.*, EX-1028, 731 (“An MMS user agent (UA), also known as an MMS client, provides means for composing, sending, retrieving, viewing, and other controlling functions to users.”), 732-733 (Section 2.2, describing multiple applications residing on a user device terminal implemented to use MMS for messaging). EX-1028 is a 2007 article by Mostafa, titled “Transporting data between wireless applications using a messaging system—MMS,” which proposes a solution to make the enhancements in developing standards backward compatible, such that transporting application data using MMS has no impact on existing MMS services. EX-1028, Abstract. Since TS-23.140 (EX-1004) is a standard describing a non-realtime MMS and portions of the standard are expressly cited in the Mostafa reference (endnote 10), the Mostafa reference is pertinent to the TS-23.140 standard and reflects what POSAs understood the TS-23.140 standard to teach.

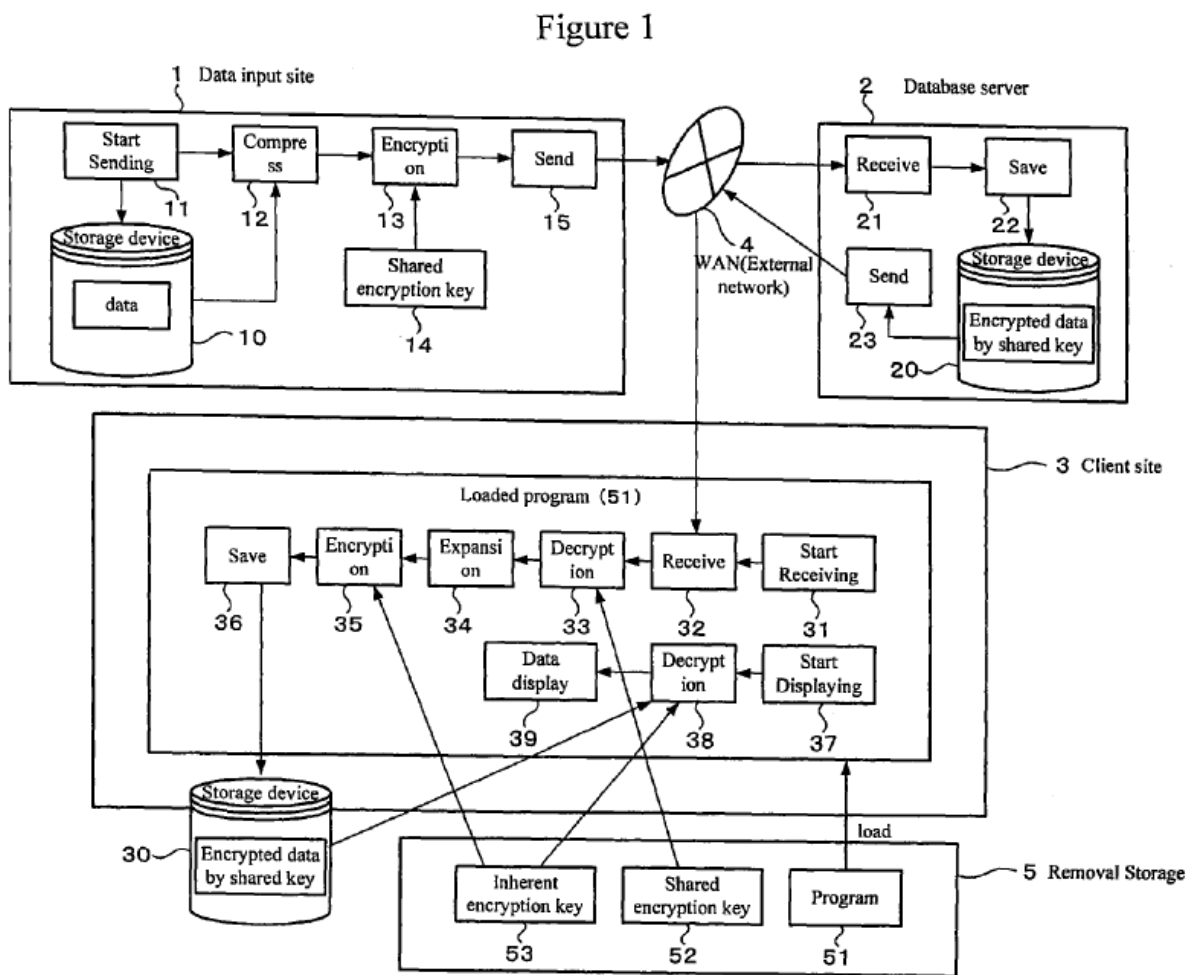
49. The “application data” which may be transported to applications other than the MMS User Agent application “may be of any content type and format.” EX-1004, 14. For such “application data,” “the MMS User Agent... route[s] the received MMS information on to the destination application” using a “destination application identifier” included with the received message. EX-1004, 14, 56; *see generally id.*, 54-56 (Section 7.1.18, Support for transporting Application Data). “Upon reception of an abstract message containing a destination application identifier[,] the receiving MMS User Agent... shall first check if the destination application resides on [the UE device].” *Id.*, 56. “If the destination application resides on a receiving MMS User Agent, the MMS User Agent shall immediately route the received MMS information on to the destination application that is referred to from the destination application identifier....” *Id.*

50. TS-23.140 also discloses multimedia message (“MM”) “encryption... on an end-user to end-user basis,” and using, e.g., Transport Layer Security (TLS) and “authentication mechanisms based on public/private key cryptography” for securing communications. EX-1004, 19, 41. For example, TS-23.140 notes that “[d]etails for implementation of the MM1 transfer protocol are described in Annex B.” EX-1004, 25 (§6.3), 162. Annex B refers to EX-1011, which in turn refers to EX-1012 for implementation details of MM1. *See* EX-1011, 11. EX-1012 explains that “[t]he TLS [WP-TLS] transport layer security protocol provides for secure

data transmission between the MMS Client and the MMS Proxy-Relay in architectural configurations that employ HTTP based protocol stacks for MMSM implementation.” EX-1012, 21.

B. Ogawa (EX-1005)

51. Ogawa discloses a “data encryption system” (EX-1005, 3:18-21), an example of which is shown below:

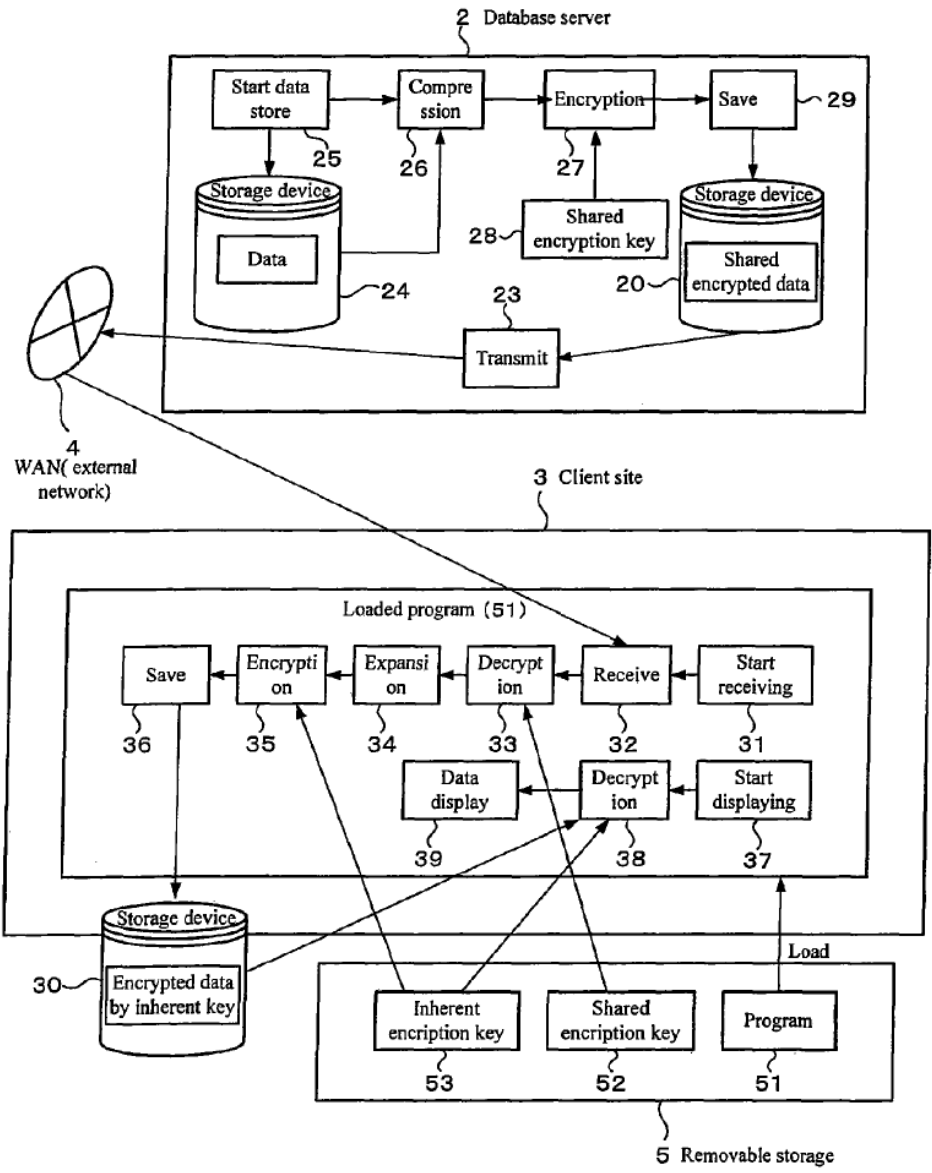


EX-1005, FIG. 1

With reference to FIG. 1, Ogawa’s system facilitates communications between multiple entities, including: (i) data input site 1, (ii) database server 2, and (iii)

client site 3. These entities are “operably linked via an external wide area telecommunication network 4,” for example, “the Internet” supported by “hyper text transfer protocol (HTTP).” EX-1005, 3:44-54 (describing FIG. 1).

52. Ogawa discloses several configurations for its system that facilitates secure communications between a database server and a client site through a network. *E.g.*, EX-1005, 9:15-38 (“...FIG. 7 shows the block configuration of an exemplary system in which the database server 2 is integrated with the function of a data input site 1 and the database server 2 and the client site 3 are connected through the external network 4.”), FIG. 7 (below). For example, FIG. 7 (below) illustrates a client-server system that performs functions described for FIG. 1 (described further below):



EX-1005, FIG. 7

In the system shown in FIG. 7, “the database server 2 is integrated with the function of a data input site 1” and “the database server 2 and the client site 3 are connected through the external network 4.” EX-1005, 9:16-20; *see generally id.*, 9:15-38 (describing FIG. 7).

53. Ogawa uses security protocols for data transmission between network entities, e.g., between a client computer and a server. “For example, in network communications utilizing a conventional TCP/IP (Transmission Control Protocol/Internet Protocol) or UDP (User Datagram Protocol), **encryption communication, such as** IPsec (Internet Protocol Security) or SSL (Secure Socket Layer), is utilized to prevent some security risks presented during the exchange of data between network terminals.” *Id.*, 3:64-4:4. POSAs understood that SSL was known as a predecessor of TLS, and thus, POSAs understood that TS-23.140 and Ogawa contemplated similar secure data transport within their networks. For documentary evidence corroborating my testimony, *see, e.g.*, EX-1010, 1:38-42 (“[T]he Secure Socket layer (SSL) protocol and its successor Transport Layer Security (TLS) provides a mechanism for securely sending data between a server and a client.”).

54. Ogawa teaches **further** securing data by, e.g., using a “shared encryption key” used by both client and server. EX-1005, 6:42-47 (describing “operations that distribute... shared encryption keys”), 7:11-21 (describing transmission and storage of encryption keys), 9:16-38 (describing use of “shared encryption key 28” in FIG. 7). For example, as shown in Ogawa’s FIG. 1, data input site 1 includes an encryption unit 13 that encrypts data using a “shared encryption key 14” and transmission unit 15 that transmits the encrypted data over

network 4 to data server 2. EX-1005, 4:48-57. Data server 2 stores this data “on a storage device 20 in an encrypted state” (*id.*, 4:58-61) and subsequently transmits it to the client site 3 via network 4. *Id.*, 5:18-23.

55. The shared key referenced above is used to (1) encrypt data that is transmitted as encrypted data to the client from the server (as discussed above), and (2) decrypt received encrypted data at the client. EX-1005, 9:21-34 (compress, encrypt, store data), 5:60-65 (decrypt). For example, Ogawa teaches that at client site 3, the encrypted data is received by the receive unit 32 and this encrypted data is decrypted by decryption unit 33 using a shared encryption key 52. *Id.*, 5:59-6:9, 7:11-21. The shared encryption key is stored in storage (e.g., a removable storage 5) and is provided to decryption unit 33 to perform the decryption. *Id.*

56. The shared encryption key is distributed to both data input site 1 and client site 3. EX-1005, 6:42-45. Once client site 3 has the key, “the client site 3 will send the encryption key to the removable storage 5,” which is “is operably linked to the client site 3 (physically or wirelessly). . . .” *Id.*, 6:46-47, 7:16-21. “When the removable storage 5 receives the encryption key from the client site 3[,] it stores the encryption key into the memory device in the removable storage. . . .” *Id.*, 7:18-21.

57. Ogawa also discloses an encryption unit 35 (shown above in FIGS. 1 and 7) for re-encrypting data for, e.g., transmission within the user device. EX-

1005, 5:59-6:9. For example, Ogawa teaches that a message received and decrypted at the client (using receive unit 32 and decryption unit 33) is decompressed and expanded as needed (using decompression/expansion unit 34), and then encrypted again—before being transmitted within the client device, e.g., for storage—by encryption unit 35, using an “inherent encryption key 53.” EX-1005, 5:59-6:26, 5:24-25. Ogawa’s inherent encryption key 53 is a second encryption key that is different from the first (“shared”) encryption key used to encrypt/decrypt data transmitted between the server and client as described above.

C. The MMS-Ogawa Combination

58. As discussed below, “MMS-Ogawa” refers to an encrypted MMS system in which communications with the MMS User Agent on TS-23.140’s user device—which may occur over a network (e.g., with an MMS Relay/Server) or within the device (e.g., with other applications)—are secured using Ogawa’s teachings. I am informed and understand that the Board considered a combination with these same modifications in a previous IPR, IPR2024-00341, for claims 1 and 27 of U.S. Patent No. 8,406,733 (“the ’733 Patent”), and I understand these claims of the ’733 patent were cancelled as unpatentable. EX-1038, 9-15, 30-31, 69-72; EX-1022, 49-51.

1. Implementing TS-23.140's User Device with a Modem

59. As I noted above (e.g., in Section VII.A, paragraphs 45, 47), TS-23.140 says its “MMS User Agent” “resid[es] on a UE [user equipment]... or... external device.” EX-1004, 14, 18-19. TS-23.140 describes the User Agent as communicating with the MMS Relay/Server using, e.g., “2G and 3G wireless networks,” but does not disclose details regarding how the device on which the User Agent resides facilitates communications over such networks. E.g., EX-1004, 17, 23-24, FIG. 2 (annotated below).

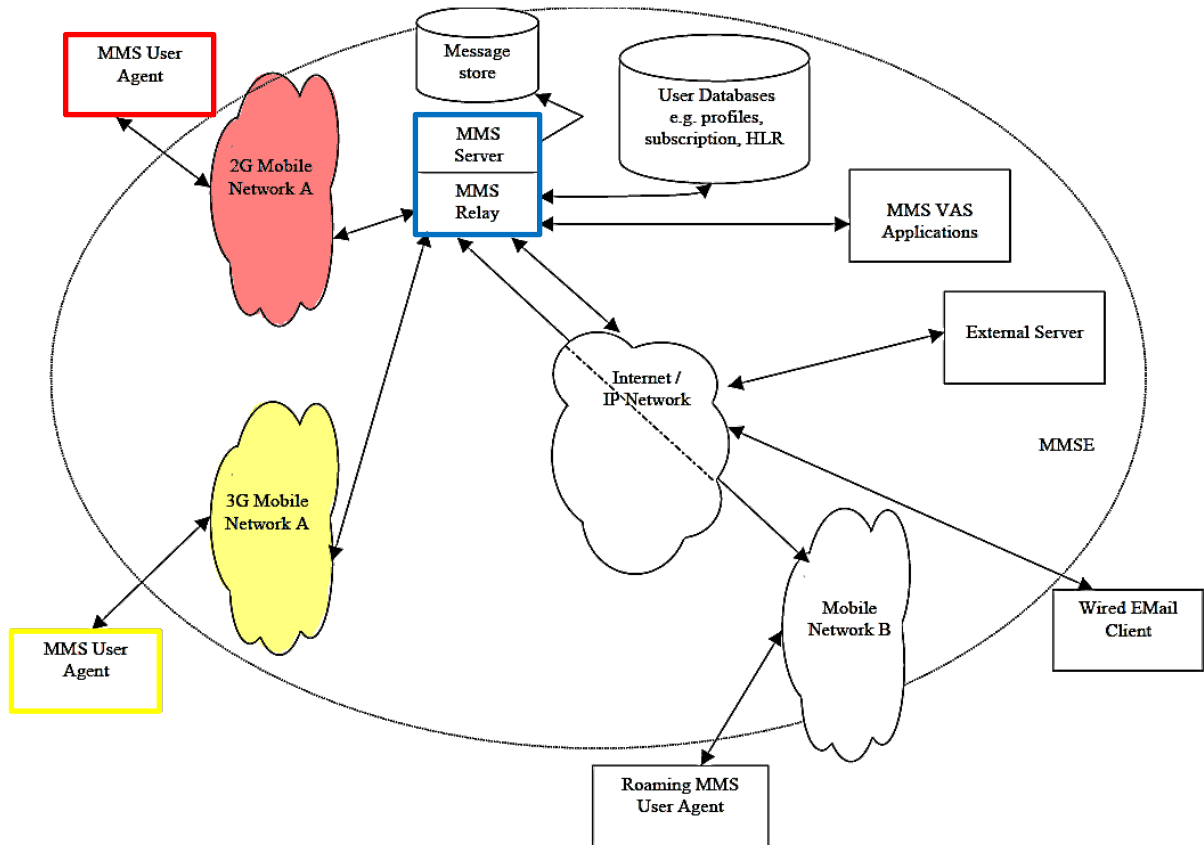


Figure 2: MMS Architectural Elements

EX-1004, FIG. 2 (annotated)

60. Before the Critical Date, it was well-known and well-documented to use a *modem* to enable communications over the networks described in TS-23.140, and a POSA would have been motivated to do so. For example, Cole (EX-1006) discloses a modem for connecting a client (Cole’s “mobile device”) to a server through a 2G or 3G network. *See* EX-1006, [0003], [0031]-[0035] (“The communication system 100 includes a mobile device 110 interfacing with a plurality of communication networks 120, such as a wireless wide area network (WWAN) 130 (e.g., ... {3G} cellular technologies, ... {2G} cellular technologies) ... Exemplary communication interfaces 225 that may be provided include a WWAN modem 230, a WLAN modem 235, a LAN device 240, a WPAN device 245, and a voice band modem 250 (e.g., V90).”); FIG. 2 (below).

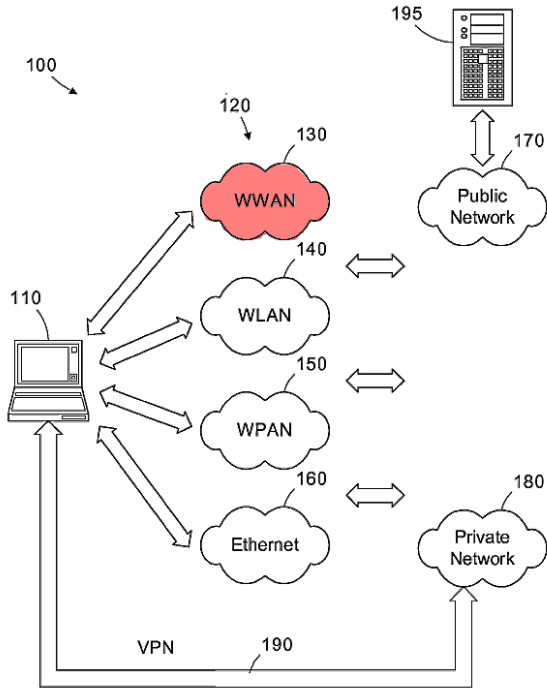


Figure 1

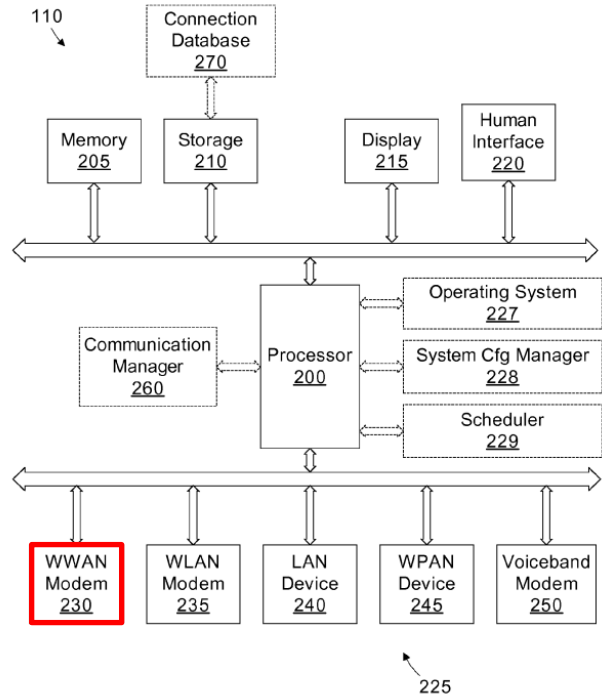
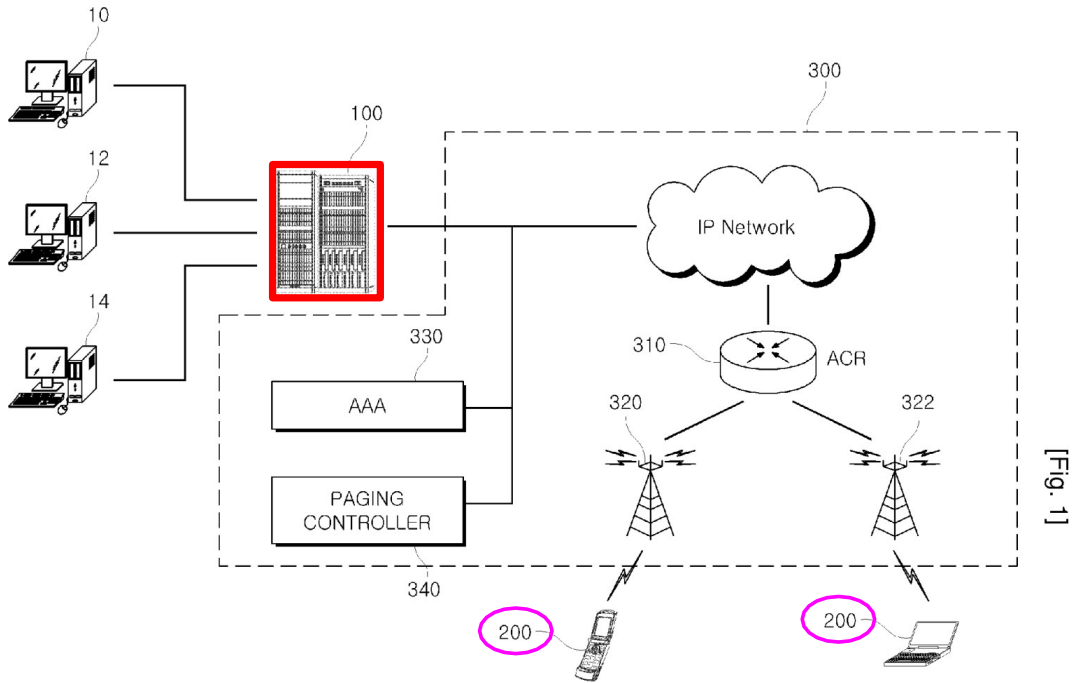


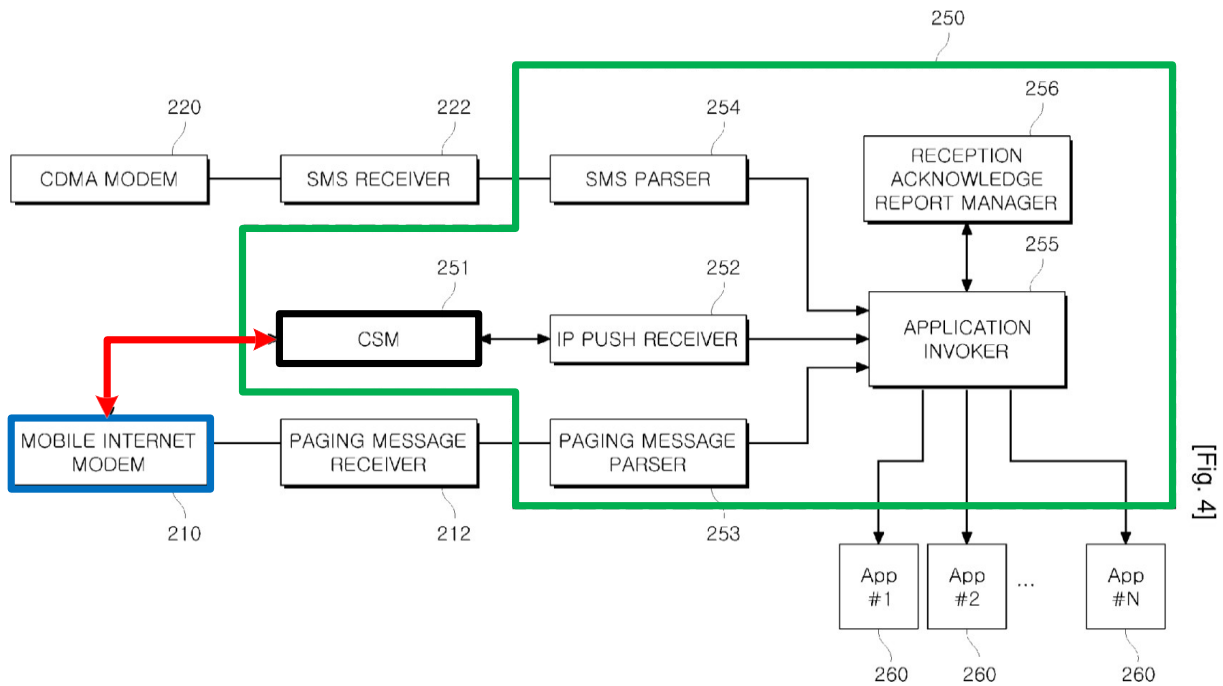
Figure 2

EX-1006, FIGS. 1-2 (annotated)

61. Another example is Lee (EX-1008), which is a corroborating reference in the general technical field of push communications between network entities and applications executing of a user device. Lee (EX-1008), Abstract, ¶¶1, 8, 27-28, 44, FIGS. 1, 4. Like the MMS User Agent on the user device in TS-23.140 that receives messages from MMS relay/server, in Lee, a service agent 250 (green in FIG. 4 below) on mobile terminal 200 (pink) receives messages from a service server 100 (red), and the mobile terminal 200 receives the messages through a mobile internet modem 210 (blue). EX-1008, ¶¶22-23, 27-28.



EX-1008, FIG. 1 (annotated)



EX-1008, FIG. 4 (annotated)

62. Using a modem to enable a user device to communicate over the wireless networks in TS-23.140 would have been a conventional, obvious way to

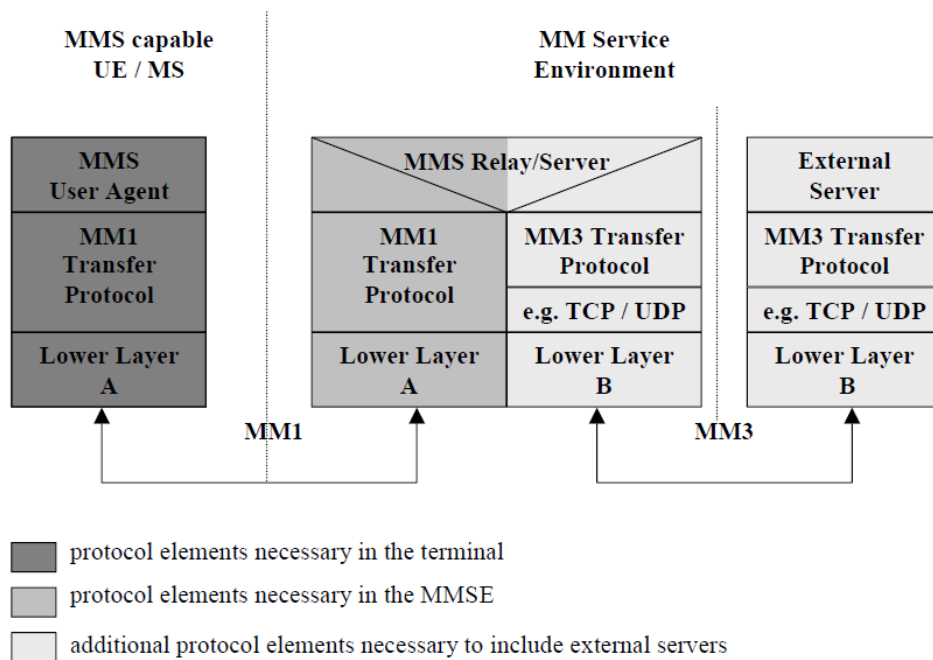
implement what TS-23.140 describes, and is nothing more than utilizing familiar, known components to achieve a predictable result of facilitating TS-23.140's communications. Indeed, a POSA would have found it obvious for a computing device to use known networking components, such as a modem, to achieve the predictable result of facilitating network communications, as already contemplated by TS-23.140. Thus, in view of TS-23.140's teachings of network communication between user devices and networked servers (which I discussed in, *e.g.*, paragraphs 45-46 above), and knowledge of a POSA before the Critical Date (as evidenced by Cole's (EX-1006) and Lee's (EX-1008) disclosures of the user device using a modem to facilitate such network communications, as I discussed in, *e.g.*, paragraphs 60-61 above), a POSA would have found it obvious to implement TS-23.140's system to include a *modem* to facilitate the above-described network communications. A POSA would have reasonably expected success in implementing TS-23.140's device with a modem because this was a well-known, conventional way (*e.g.*, EX-1006, [0031]-[0035]; EX-1008, ¶¶22-23, 27-28) of achieving the wireless network communications that TS-23.140 describes (*e.g.*, EX-1004, 17, 23-24, FIGS. 2-4).

63. In IPR2024-00341, the Board agreed, “find[ing] that the cited portions of the record show that modems were known for enabling communication over networks, such as those described in TS-23.140” and “credit[ing] Petitioner's

testimonial evidence regarding modems and that one of ordinary skill would have implemented TS-23.140 to include a modem with a reasonable expectation of success because the cited portions of the record support the testimony.” EX-1022, 40-41 (citing, e.g., EX-1004, 14, 17-19, 23-24; EX-1008, ¶¶27-28, 44, FIGS. 1, 4).

2. Securing Interface MM1 Using SSL/TLS

64. TS-23.140 explains that network communications between the MMS User Agent and the MMS Relay/Server use interface MM1. EX-1004, 24, FIG. 4 (below).



**Figure 4: Protocol Framework to provide MMS
EX-1004, 24, FIG. 4**

As shown in the figure above, the MM1 Transfer Protocol for the MM1 interface is implemented both at the MMS Relay/Server and the MMS UE. See EX-1004, 24, FIG. 4. A POSA would have understood that the MM1 interface which uses the

MM1 Transfer Protocol facilitates transmission/transport of network communications between the MMS User Agent and MMS Relay/Server (and with other network elements via the MMS Relay/Server, such as MMS VAS Applications, which additionally leverage the MM7 transport interface). *See also, e.g.,* EX-1018, 6 (MM1 Interface is an “[i]nterface between MMS Relay/Server and MMS User Agent”; MM7 is “[i]nterface between MMS Relay/Server and MMS VAS Applications”), 9 (FIG. 1 showing MMS architecture including MM1 and MM7 interfaces).

65. A POSA had multiple reasons to secure interface MM1 with an SSL/TLS protocol, as I explain below.

66. **First**, TS-23.140 expressly contemplates implementations which use “transport layer security mechanisms” (e.g., SSL/TLS) to secure its communication links, including MM1 between the user device (with the MMS User Agent) and the MMS Relay/Server. For example, TS-23.140 teaches an “WAP/OMA implementation” for the MM1 interface between the MMS User Agent (on the user device) and the MMS Relay/Server. EX-1004, 30, 55, 162, 174. WAP/OMA refers to Open Mobile Alliance’s “Wireless Application Protocol.” EX-1004, 13 (citing numerous “Open Mobile Alliance” references), 16 (defining WAP acronym). To describe “[d]etails” for this “implementation of the MM1 transfer protocol,” TS-23.140 incorporates-by-reference EX-1011, an “Enabler

Release Definition.” See EX-1004, 13, 24-25, 162. EX-1011, in turn, incorporates-by-reference EX-1012, an “Architecture Overview.” EX-1011, 4, 5, 10, 11. The Architecture Overview (EX-1012, 21) states that “[t]he *TLS* [WP-TLS] transport layer security protocol *provides for secure data transmission between the MMS Client and the MMS Proxy-Relay* in architectural configurations that employ HTTP based protocol stacks for MMSM implementation.” In TS-23.140’s disclosed WAP/OMA implementation of MMS, the “MMS User Agent is... responsible for sending and receiving MMs by utilising the message transfer services of the appropriate network protocols,” e.g. SSL/TLS for securing MM1 between the MMS User Agent and the MMS Relay/Server. EX-1012, 17, 21. TS-23.140 includes similar disclosures regarding other communications links, e.g. MM7 between the MMS Relay/Server and the MMS VAS Applications. *E.g.*, EX-1004, 41 (“interface [MM7] between MMS Relay/server and VASP may be carried over an encrypted and secure bearer, e.g. HTTP over SSL or TLS[.]”).

67. **Second**, it was well-known and well-documented before the Critical Date to use SSL/TLS to achieve secure communications between entities on a network, e.g. between a client and server. *See, e.g.*, EX-1012, 21 (disclosing “encrypt[ing]” the communication channel between the client and the backend MMS server with some security protocols, e.g., WTLS (Wireless Transport Layer Security)); EX-1010, 1:38-42 (“A variety of *cryptographic techniques* are known

for securing transactions in data networks. *For example, the Secure Socket layer (SSL) protocol and its successor Transport Layer Security (TLS) provides a mechanism for securely sending data between a server and a client.*”); EX-1013, 3 (“Cryptographic security[:] SSL should be used to establish a secure connection between two parties.”).

68. **Third**, such an implementation is nothing more than utilizing familiar, known protocols to achieve a predictable result of facilitating secure communications with TS-23.140’s User Agent over MM1.

69. A POSA would have reasonably expected success implementing MM1 to use SSL/TLS, given TS-23.140’s teachings and incorporated disclosures, and the widespread use of such security protocols before the Critical Date.

70. In IPR2024-00341, the Board agreed, “find[ing] that the cited portions of the record support that one of ordinary skill in the art would have used SSL/TLS protocol” for MM1, “and that such use would have had a reasonable expectation of success,” and “credit[ing] Petitioner’s testimonial evidence” regarding same “because the record supports it.” EX-1022, 41-42 (citing, e.g., EX-1004, 24-25, 55, FIG. 4; EX-1010, 1:38-42; EX-1012, 21; EX-1013, 3; EX-1018).

3. Applying Ogawa’s Symmetric Encryption Techniques For Network Communications

71. TS-23.140 discloses “encryption of an MM [Multimedia Message] on an end-user to end-user basis.” EX-1004, 19. In addition to “transport layer

security mechanisms,” TS-23.140 says “authentication mechanisms based on public/private key cryptography... may also be used.” EX-1004, 25, 41; EX-1012, 21 (“An aspect of the MMS user interface is that of conveying information related to the security and/or authentication of messages received or to be sent.”). For example, Public Key Infrastructure (PKI) is one security service that may be implemented via MM1 using WAP/OMA. EX-1004, 30 (WAP/OMA implementation), 41 (7.1.13.1 Authentication); EX-1012, 21 (PKI as example of “security services”).

72. A POSA understood that “an end-user” can be an MMS User Agent, or an MMS Relay/Server. *See e.g.*, EX-1004, 62 (“Reference point MM1 defines the transactions between the MMS User Agent and the MMS Relay/Server.

...Figure 6 illustrates some of these transactions and their relationships, in an **end-to-end** manner.”). Alternatively, it would have been obvious to a POSA to ensure a message (being sent between end-users via the MMS Relay/Server) is encrypted when it is stored on the MMS Relay/Server and sent from the MMS Relay/Server to the receiving MMS User Agent. *See* EX-1004, 17 (“The MMS Relay/Server is responsible for storage and handling of incoming and outgoing messages and for the transfer of messages between different messaging systems.”). A POSA would have been motivated to do so, for example, to make sure a message received at the MMS Relay/Server would be secure from, *e.g.*, theft, access, or manipulation while

stored there and when sent to the MMS User Agent. *See e.g.*, EX-1005, 8:16-21 (disclosing encryption of a message before storing it).

73. TS-23.140 does not provide details regarding how to implement the additional end-user-to-end-user encryption beyond SSL/TLS. However, before the Critical Date, it was well-known and well-documented to implement encryption for messages transmitted by a server (*e.g.*, TS-23.140's MMS Relay/Server) to an end-user device using symmetric encryption, with a key that is shared between the server and the end-user device and stored in the respective memories. Multiple references before the Critical Date disclose encryption solutions where the data in messages sent between server and a client is encrypted using a shared key stored by the client and server, and the shared key is used for encrypting/decrypting messages sent between the client and the server. For example, Shen (EX-1027) discloses data encryption over push messages (MMS messages) using a shared key established by the client and server and distributed to users, and using this shared key for encrypting/decrypting messages between user devices. EX-1027, [0054]-[0060]. Another example is Qumei (EX-1009). *See, e.g.*, EX-1009, 3:25-27 (“[A]n enciphering algorithm and an enciphering key may be stored in the electronic devices”), 8:1-5 (“Symmetric cyphering/enciphering may use one or multiple keys for both encryption and decryption”). And Ogawa (EX-1005), discussed above in Section VII.B, discloses details regarding how to implement symmetric encryption

on messages in a client-server environment like the one described in TS-23.140, where “SSL... is utilized to prevent some security risks presented during the exchange of data between network terminals.” EX-1005, 3:61-4:4, 9:16-34, 3:44-53.

74. A POSA had multiple reasons to implement Ogawa’s symmetric data encryption techniques with TS-23.140’s MMS system (“MMS-Ogawa Message Encryption”), as I explain below.

75. *First*, implementing the encryption taught in Ogawa to TS-23.140’s MMS system would have achieved a system “having improved security” and providing “an end-user to end-user” security solution for MMS applications, as contemplated by TS-23.140. EX-1004, 19 (describing “MMS User Agent may provide additional application layer functionalities”), 24-25 (describing the MM1 interface between the MMS User Agent and the MMS Relay/Server); EX-1012, 21 (“The TLS [WP-TLS] transport layer security protocol provides for secure data transmission between the MMS Client and the MMS Proxy-Relay in architectural configurations that employ HTTP based protocol stacks for MMSM implementation.”).

76. *Second*, encrypting MMS communications using an additional layer of security beyond SSL/TLS as taught in Ogawa would have been particularly beneficial for “enterprise applications” (e.g., large-scale software systems designed

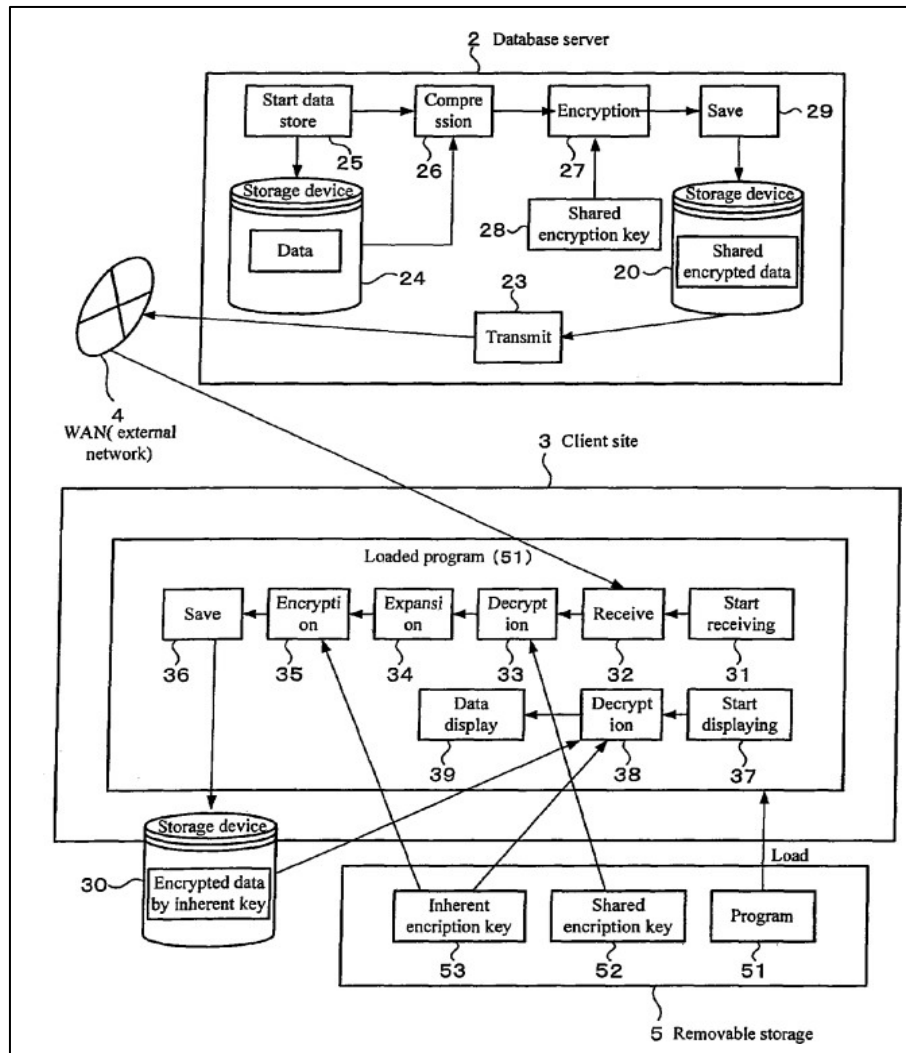
to integrate and automate a company's core business processes), which POSAs understood were examples of a value-added service application that a POSA would have been motivated to ensure its MMS system could handle. EX-1027, [0002] (“an end-to-end solution is needed for large enterprises to provide a high security message solution for enterprise applications”), [0003] (describing enterprise users communicating via MMS systems) [0017] (describing “ a direct communication channel between the MMS user device and the enterprise applications”), [0021]-[0022] (describing security for MMS for enterprise applications); EX-1004, 54 (describing support for transporting application data).

77. **Third**, implementing Ogawa's data encryption into TS-23.140's MMS system would have been nothing more than implementing known methods/techniques (symmetric encryption using a shared key as taught in Ogawa) to known systems/devices (the MMS Relay/Server and the MMS User Agent device taught in TS-23.140) to achieve predictable results (end-user-to-end-user encrypted data, as contemplated by TS-23.140).

78. A POSA would have reasonably expected success implementing Ogawa's encryption techniques to the TS-23.140 system given (1) the similar client-server communication architectures taught by both references (with connections between network elements protected by, e.g., SSL/TLS) and (2) TS-23.140 contemplates end-user-to-end-user encryption, and Ogawa provided an

exemplary, predictable implementation of such encryption using a symmetric, shared encryption key. *See also* my discussion above in paragraph 73, discussing additional references that discuss use of a symmetric, shared encryption keys.

79. Specifically, in the same field of client-server communications, Ogawa explains how symmetric encryption can be implemented for securing communications in similar networked environments. Similar to TS-23.140, Ogawa uses the SSL protocol (which is a predecessor of the TLS protocol) to facilitate secure network communications between a server and a user device. EX-1005, 3:60-4:4. As illustrated in FIG. 7 of Ogawa (reproduced below), database server 2 transmits an encrypted message to client site 3 through network 4 using this secure, SSL-enabled communication link and in doing so, Ogawa seeks “to prevent some security risks presented during the exchange of data between network terminals.” EX-1005, 3:44-53; 9:16-34 (network 4 uses HTTP protocol for message transmission).



EX-1005, FIG. 7

80. Additionally, Ogawa discloses applying data encryption to messages transmitted between devices, and for such data to be encrypted, using a symmetric/shared key between the client and server, as discussed above. EX-1005, 4:53-54 (shared encryption key 14), 7:10-12 (key is transmitted to data input site 1). Ogawa provides an example of how such symmetric encryption works—i.e., by storing a shared key in each respective device’s memory that would be used for both encrypting and decrypting a transmitted or received message. EX-1005, 4:48-

57, 5:59-65, 6:64-7:21. This is further corroborated by Qumei. *See e.g.*, EX-1009, 3:25-27 (“an enciphering algorithm and an enciphering key may be stored in the electronic devices”), 8:1-5 (“symmetric cyphering/enciphering may use one or multiple keys for both encryption and decryption”).

81. A POSA thus would have reasonably expected success implementing Ogawa’s encryption teachings of using a shared encryption key to encrypt server-client communications. Such a combination would have also been predictable and straightforward to implement (and a POSA would have had a reasonable expectation of success in doing so) given the similar MMS/client-server architectures taught by both references and given that TS-23.140 contemplates end-to-end encryption and Ogawa provides one example and predictable implementation of such encryption using a symmetric/shared encryption key.

82. As explained further in the following section (Section VII.C.4), in IPR2024-00341, the Board agreed that applying Ogawa’s symmetric encryption techniques for network communications, as described above, was obvious.

4. Ogawa’s Decryption and Encryption Units

83. Ogawa teaches a “decryption unit” for decrypting received encrypted data as part of its symmetric encryption system. EX-1005, 5:59-6:9 (“[R]eceived shared key encrypted data will be decrypted using the shared encryption key 52 which was supplied to the decryption unit 33 and beforehand stored....”). A POSA

would have had reason to (and would have been motivated to) implement the decryption unit *as part of the MMS User Agent* application in TS-23.140, because the MMS User Agent is responsible for receiving data from TS-23.140’s MMS Relay/Server and distributing the data to the correct applications on the user device. EX-1004, 19. TS-23.140 also expressly identifies the MMS User Agent as “provid[ing]... functionalities such as ... *decryption*.” *Id.*; see generally EX-1004, 54-56 (Section 7.1.18, Support for transporting Application Data). A POSA would have recognized that such an implementation would have, e.g., beneficially allowed the MMS User Agent to decrypt an encrypted message from the MMS Relay/Server and any information identifying the destination application to which the message should be routed—particularly in the case where the entire message is encrypted (including the identifying information specifying where to route the received message).

84. Ogawa also teaches an encryption unit for encrypting (or “re-encrypt[ing]”) data before transmitting it to another part of the device, e.g., storage. EX-1005, 5:59-6:9. As with the decryption unit, a POSA would have had reason to (and would have been motivated to) implement the encryption unit as part of the MMS User Agent in TS-23.140, because (1) secure transmission and storage of data within a user device was a known feature that desirably would have helped prevent, e.g., theft, (2) the MMS User Agent is responsible for “transport of

application data” and “all aspects of storing” messages on TS-23.140’s user device, and (3) TS-23.140 expressly identifies the MMS User Agent as “provid[ing]... functionalities such as... *encryption*.” EX-1004, 19; EX-1037 (describing a commercially available mobile device containing “security features, including memory encryption”).

85. A POSA would have reasonably expected success in an implementation with Ogawa’s encryption and decryption units incorporated into TS-23.140’s MMS User Agent application, because the prior art components would continue to perform functions they performed prior to the combination—MMS User Agents and the MMS Relay/Server would continue to exchange data using secure communication links, and Ogawa’s decryption and encryption units (as implemented as part of each MMS User Agent) would use a shared encryption key to encrypt/decrypt data received from the MMS Relay/Server. In my opinion, based on my understanding of a POSA’s skill as discussed in Section IV above, such a combination would have been well within a POSA’s capability to implement.

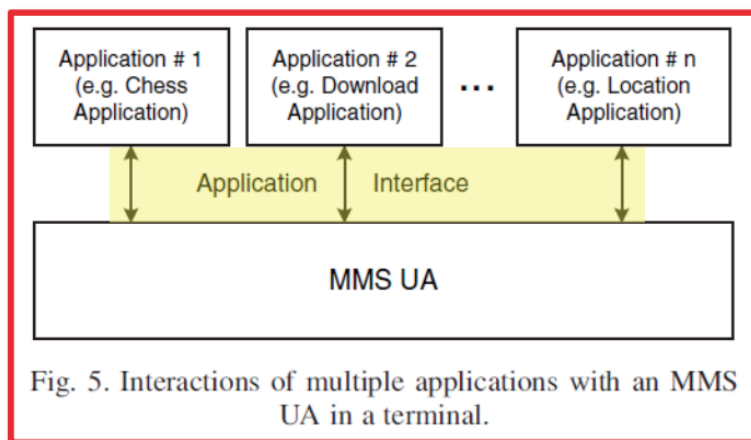
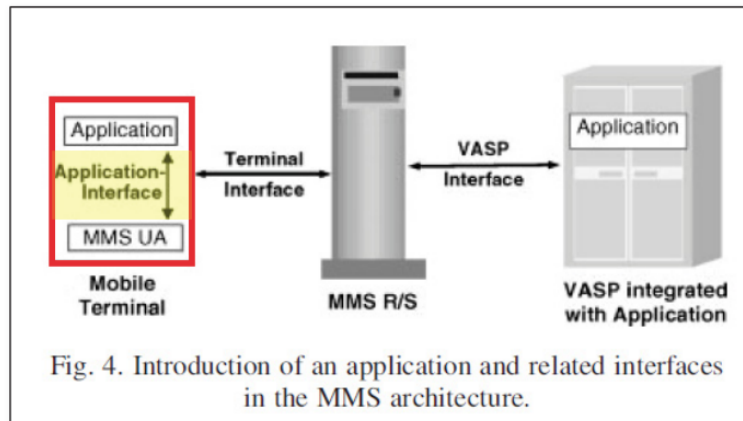
86. The Board in IPR2024-00341 considered the modifications of MMS described in TS-23.140 in view of Ogawa that I described in Sections VII.C.3-VII.C.4 above, and the Board agreed that those modifications would have been obvious, “find[ing] that the cited portions of the record support that it was known

to implement symmetric encryption in message transmissions like those in TS-23.140 and that Ogawa discloses how to implement symmetric encryption,” and “credit[ing] Petitioner’s testimonial evidence that one of ordinary skill in the art would have implemented Ogawa’s decryption and encryption units in TS-23.140 for the reasons asserted by Petitioner, and that such implementation would have had a reasonable expectation of success, because the cited portions of the record support it.” EX-1022, 42-43 (citing, e.g., EX-1004, 14, 17, 19, 24-25, 41, 54-56, 62; EX-1005, 3:60-4:4, 4:34-57, 5:59-6:9, 6:64-7:21, 8:16-19, 9:16-34, FIG. 7; EX-1009, 3:25-27, 8:1-5; EX-1012, 21; EX-1027, [0017], [0021]-[0022], [0054]-[0060]).

5. Implementing TS-23.140’s User Device with an Interprocess Communication Bus

87. TS-23.140 discloses its MMS User Agent receiving “data specific to an application other than the MMS User Agent” from MMS VAS Applications and “rout[ing]” that “received MMS information” (such as application-specific data) “to [a] destination application” on the user device—but leaves to a POSA the “[d]etails of... how an MMS User Agent... would *interface* with” the user device’s destination applications. EX-1004, 54-56 (Section 7.1.18; Support for transporting Application Data). Contemporaneous references discussing MMS confirm that TS-23.140’s UE included an interface for communications between the MMS User Agent and other applications. *E.g.*, EX-1028, 729-730 (discussing

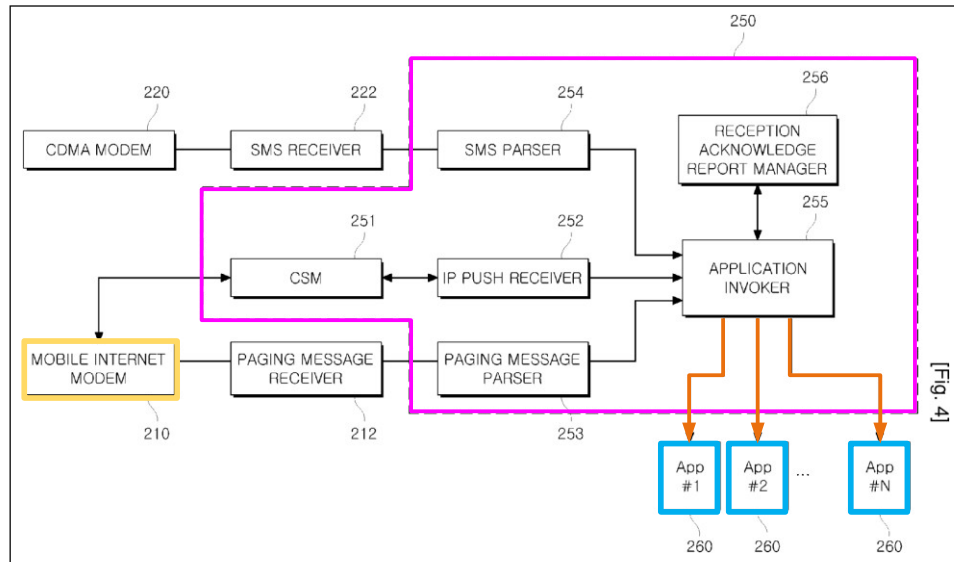
the “interface between an application and MMS in a terminal”), 732-733 (discussing the “application-interface between an application and an MMS UA” “within an MMS-capable terminal”), FIGS. 4-5 (annotated below).



EX-1028, FIGS. 4-5 (annotated)

88. As another example, Lee (EX-1008) corroborates that there would be a communication channel in the end-user device between a push service agent 250 that receives messages intended for applications resident on a device from external sources, and a target application 260 to which the service agent 250 sends the message. EX-1008, ¶28. As shown in Lee’s FIG. 4 below, the push service agent 250 (*service control device link agent*) is *communicatively coupled* to the multiple

applications 260 (*the plurality of device agents*) through a communication bus including respective communication channels (depicted in orange in figure below).



EX-1008, FIG. 4 (annotated)

89. A POSA would have had reason to (and would have been motivated to) implement the interface between TS-23.140’s MMS User Agent application and the user device’s other applications using a software *bus* for interprocess communications—for example, a D-bus. Before the Critical Date, it was well-known and well-documented to use a software bus (e.g., a D-bus) for inter-process communications that enable applications to interface with one another, as TS-23.140 describes. *See, e.g.*, EX-1031, 10:56-62 (“[T]he D-Bus may be an example of a device inter-process communication channel used to send information between applications. As such, method calls made over the D-Bus may be serialized as a message addressed to a particular application. The particular application may then listen and perform the service requested and place an output back on the D-Bus

addressed to the browser application.”); EX-1048, [0001], [0003], [0004], [0019] (describing “communication” between “one or more processes” occurring over a “bus”); EX-1049, 907 (defining “software bus”: “A programming interface that allows software modules to transfer data to each other...”); EX-1050, 2:64-3:6 (describing “software bus” that “interconnects” a “control software’s subsystems”), 3:51-67, FIG. 5.

90. Using such a bus to enable an MMS User Agent to communicate with other applications on TS-23.140’s user device would have been a conventional, obvious way to implement what TS-23.140 describes, and is nothing more than utilizing familiar, known components (as I discussed in paragraphs 87-89 above) to achieve a predictable result of facilitating TS-23.140’s communications.

91. Moreover, a POSA would have reasonably expected success with such an implementation because a software bus (e.g., a D-bus) was a well-known, conventional way of achieving the communications between applications that TS-23.140 describes (as evidenced by, for example, the references I discussed in paragraphs 87-89 above).

92. In IPR2024-00341, the Board agreed with Petitioner that the implementation I discussed above in this Section would have been obvious. In response to Petitioner’s “argu[ment] that [a POSA] would have implemented communications between the MMS User Agent and other applications over a bus,

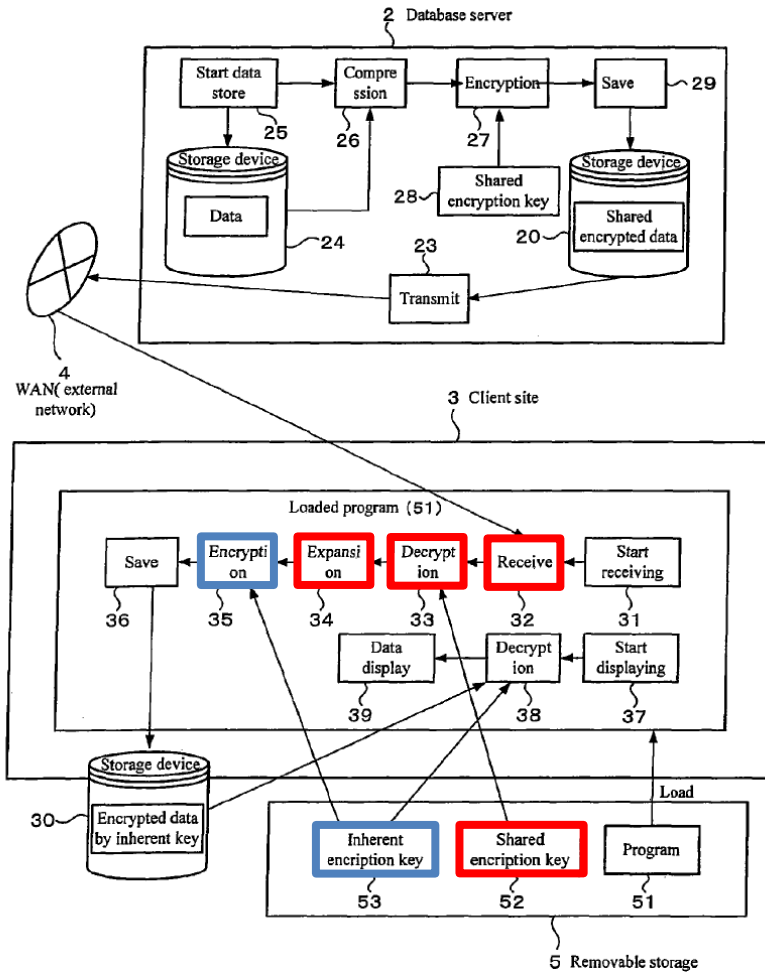
such as a D-bus” (EX-1022, 45), the Board found that a POSA “would have implemented a bus for communications between the destination applications and the MMS User Agent[,]” “credit[ing] Petitioner’s testimonial evidence that a bus was known for inter-process communication and would have been a conventional way to implement what TS-23.140 describes, because the cited portions of the record support it... [and] that using such a bus would have been using a known component to achieve the predictable result of providing the described communications between the MMS User Agent and destination applications.” EX-1022, 48 (citing, e.g., EX-1008, [0028], FIG. 4; EX-1028, 732-733; EX-1031, 10:56-62). The Board “also credit[ed] the testimony that the implementation would have had a reasonable expectation of success.” EX-1022, 48-49.

6. Securing Interprocess Communications Within the Device

93. As I discussed in Section VII.C.4 above, Ogawa discloses an encryption unit 35 for re-encrypting data that is transmitted within a user device. EX-1005, 6:1-9. In Ogawa, an encrypted message that is received by the client (e.g., from a server) is decrypted at the client using decryption unit 33 and shared encryption key 52. EX-1005, 5:24-27, 5:59-6:26. It is then decompressed and expanded as needed using decompression/expansion unit 34. *Id.* It is then encrypted again by encryption unit 35, using an “inherent encryption key 53” before being transmitted elsewhere within the client device, e.g., for storage. *Id.*

Such secure communications within a device using encryption was known to a POSA and well-documented before the Critical Date, for example as described in EX-1039 (Yami, “System and Method for Secure Inter-process Data Communication”), which describes secure communications between applications within a device using symmetric encryption keys. *See* EX-1039, Abstract, [0001] (“The subject application is directed to a system and process for secure inter-process data communication...”), [0002]-[0004], [0007] (“The system includes means adapted for receiving job data and means adapted for receiving symmetric key data...”), [0008].

94. Ogawa’s inherent encryption key 53 is a second encryption key that is different from the first “shared” encryption key used to encrypt/decrypt data transmitted between the server and client, as I discussed in paragraph 57 above. When the encrypted message is accessed again within the client device, “the decryption unit 38... decrypt[s] the... encrypted data... using the inherent encryption key 53[.]” EX-1005, 6:10-14, FIG. 7 (annotated below).



EX-1005, FIG. 7 (annotated)

95. A POSA had motivation and reason to include the re-encryption functionality as part of MMS-Ogawa’s MMS User Agent, e.g., to secure decrypted received application data that was received over MM1 before transmitting it to any other component on the user device—and would have reasonably expected success doing so—for at least three reasons. First, such internal encryption within the device using an inherent key was expressly taught by Ogawa, as I discussed in paragraphs 57 and 93-94 above. Second, securing interprocess communications, e.g. between applications on a device, was well-known and well-documented. *See,*

e.g., EX-1039 (Yami, “System and method for secure inter-process data communication”), Abstract (securing inter-process communications using symmetric keys), [0001]-[0004] (background; describing a need for a system and method for secure inter-process communications), [0007]-[0008] (summary; high level description of securing inter-process communications using symmetric keys); *see also* EX-1037 (corroborating that that encrypting the received data within a device before storing it there was a well-known technique). Third, an implementation where internal communications were protected using an inherent key would have improved security of data in MMS-Ogawa’s user device when stored or internally transmitted and would have helped, e.g., prevent unauthorized access to the data by rogue/unauthorized software on the device.

96. A POSA would also have had reason to implement MMS-Ogawa’s User Agent such that it performed any intermediary functions/steps between decryption (e.g., by Ogawa’s unit 33) and re-encryption (e.g., by Ogawa’s unit 35) that were required to ensure that the destination (on the user device) received data with correct format and content. EX-1005, 5:65-6:3. A POSA would have reasonably expected success implementing MMS-Ogawa’s User Agent to perform the intermediary functions described in Ogawa, as discussed above—e.g., by incorporating Ogawa’s expansion unit 34—because the functions of MMS-Ogawa’s User Agent would not be affected by the intermediary steps. For

example, the MMS User Agent would continue to receive, decrypt, and send data within TS-23.140's user device, and Ogawa's expansion and encryption units (implemented as part of the MMS User Agent) would continue to convert decrypted data into its original format before encrypting it for transmission within the user device. *Id.*

97. I note that in IPR2024-00341, the Board “determine[d]” that the arguments discussing Ogawa's inherent keys and securing communications within MMS-Ogawa's device (EX-1038, 69-72) were “fully supported by the record” (EX-1022, 50-51).

7. MMS-Ogawa

98. “MMS-Ogawa” refers to the above-discussed encrypted MMS system that a POSA would have been led to form based on TS-23.140 and Ogawa.

99. MMS-Ogawa implements TS-23.140's user device (configured to use TS-23.140's MMS service) with a modem for wireless network communications. *See* my discussion above in Section VII.C.1, at paragraphs 59-63.

100. MMS-Ogawa also implements TS-23.140's MMS Relay/Server and user device so that messages transmitted across the interface between TS-23.140's user device and the MMS Relay/Server are secured both with SSL/TLS and encrypted using MMS-Ogawa Message Encryption. *See* my discussion above in Sections VII.C.2-VII.C.3, at paragraphs 64-82.

101. In MMS-Ogawa, Ogawa’s encryption and decryption units are implemented as part of the MMS User Agent in TS-23.140’s user device. *See* my discussion above in Section VII.C.4, at paragraphs 83-86.

102. Additionally, MMS-Ogawa implements an interprocess bus for communications that occur within TS-23.140’s user device between applications (*see* my discussion above in Section VII.C.5, paragraphs 87-92), which is secured using Ogawa’s inherent key encryption teachings (*see* my discussion above in Section VII.C.6, paragraphs 93-96).

D. Claim Analysis

1. Claim 1

a. [1 PRE] “A mobile end-user-area device comprising:”

103. The ’403 specification³ does not describe a “mobile end-user-area⁴ device,” but uses the term “end user device.” *See, e.g.*, EX-1001, 8:3-15, 8:60-9:15. The ’403 specification defines no requirements for an “end user device” (*see*

³ When I refer to “the ’403 specification” in this declaration, I refer to EX-1001, 1:1-163:36 and FIGs. 1-64, which reflect the disclosure that was included in the March 24, 2015 continuation application. EX-1002, 10-250. This does not include the title or abstract, which were added later, on March 25, 2015. EX-1002, 365.

⁴ The inclusion of “-area” in [1PRE] appears to be a typographical error, because each dependent claim starts with “The mobile end-user device of claim 1.”

EX-1001, 8:3-15, 8:60-9:15 (generic descriptions of the devices)), using the term to include “networked” devices that have “services delivered” to them. EX-1001, 5:65-6:28 (“connected (e.g., networked)” devices), 6:49-56 (“service” functions, consumption, costs). A POSA understood that the “devices” described in columns 5-6 were the “end user devices” described in columns 8-9. A POSA understood TS-23.140’s “UE,” “MS,” “external device,” and “mobile phone” to be examples of subscriber wireless devices, such as the specification describes (EX-1001, 5:65-6:28). In addition, in IPR2024-00341, the Board found “that the User Equipment of TS-23.140 teaches, suggests, or would have been understood to disclose, to the extent the preamble is limiting, ‘[a]n end-user device.’” EX-1022, 15.

104. A POSA would have understood that MMS-Ogawa’s “UE [*user* equipment]”/“MS”/“external *device*”/“*mobile* phone”—with an MMS User Agent that “performs... operations on a *user*’s behalf,” and through which “Value Added Services” are “provided” (delivered) to “*users*”—is a “*mobile* end-user-area device.” EX-1004, 14 (MMS User Agent), 18, 54.

b. [1A] “a wireless wide-area network (WWAN) modem to exchange Internet data via a connection to a first WWAN, when configured for and connected to the first WWAN;”

105. As I discussed above in Section VII.C.1, MMS-Ogawa includes a “*modem*.” In IPR2024-00341, the Board credited Petitioner’s showing that MMS-Ogawa “had a modem” for network communications because “TS-23.140 shows

an MMS User Agent communicating through interface MM1 with an MMS Relay/Server, MMS VAS Applications, MMS User Databases, and other components,” and “[t]he other-relied upon portions describe those components and sending and receiving messages between an MMS User Agent or MMS VAS Application and a destination application over wireless networks.” EX-1022, 15-16 (citing EX-1004, 14, 18, 23-25, 55-56, FIGs. 2-4).

106. Element [1A] requires “a *wireless wide-area network (WWAN)* modem.” The ’403 specification uses the term “WWAN modem” to include “a wide area access technology modem such as 2G, 2.5G, 3G or 4G.” EX-1001, 29:52-53; *see also id.*, 33:57-63 (describing a “2G and/or 3G WWAN” “access modem”), 2:17-23 (3G and 4G WWANs), 12:63-13:3 (2G, 3G and 4G capable), 25:29-45 (3G and 4G WWANs), 27:38-44 (WWAN modem). MMS-Ogawa’s modem for 2G and 3G wireless networks (*see my discussion above in Section VII.C.1*) is thus “a *wireless wide-area network (WWAN) modem*,” as claimed.

107. A POSA further understood that MMS-Ogawa’s modem for 2G and 3G wireless networks (*see my discussion above in Section VII.C.1*) is “[for] exchang[ing] Internet data via a connection to a first WWAN, when configured for and connected to the first WWAN,” as claimed. This is confirmed, for example, in FIG. 2 of TS-23.140, which shows an exemplary Multimedia Messaging Service Environment (MMSE) in which TS-23.140’s UE/device (MMS-Ogawa’s user

device—see my discussion above in Section VII.D.1.a, discussing [1PRE])

operates, comprising 2G (red) and 3G (yellow) wireless networks:

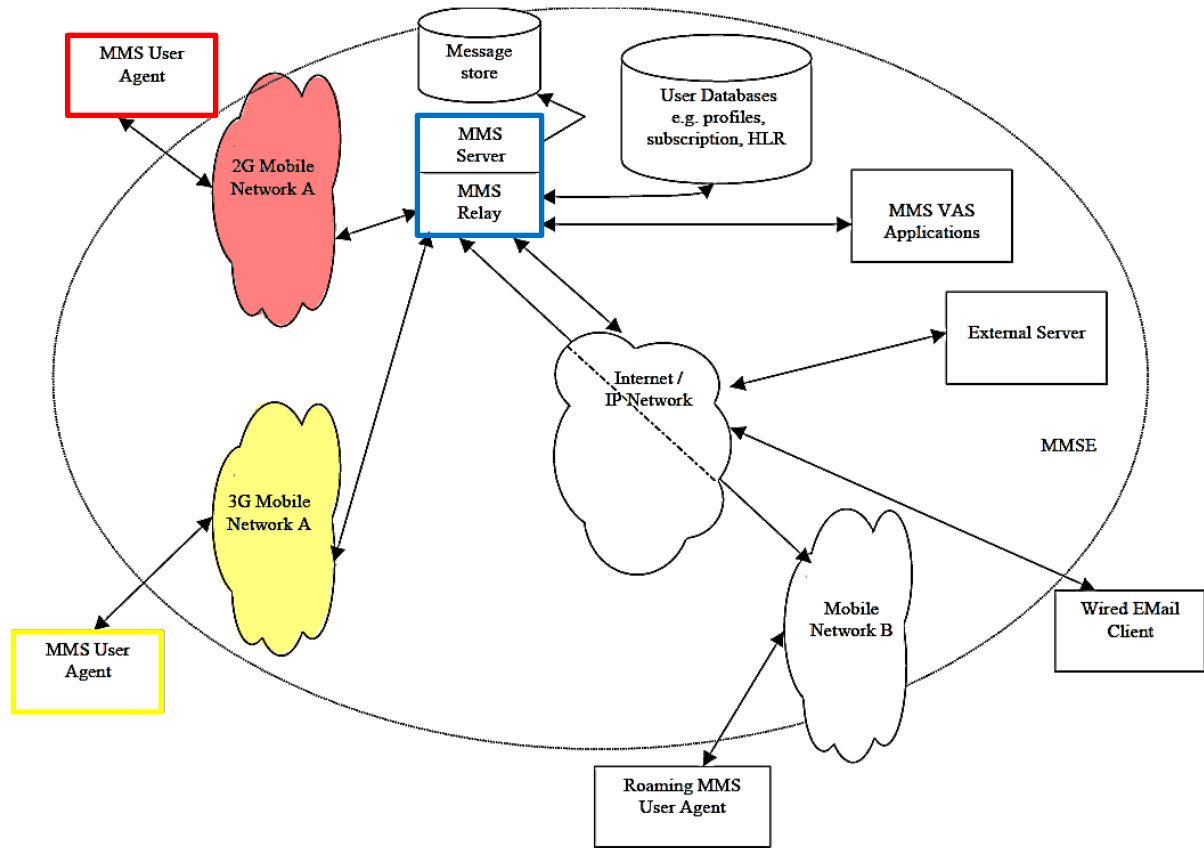
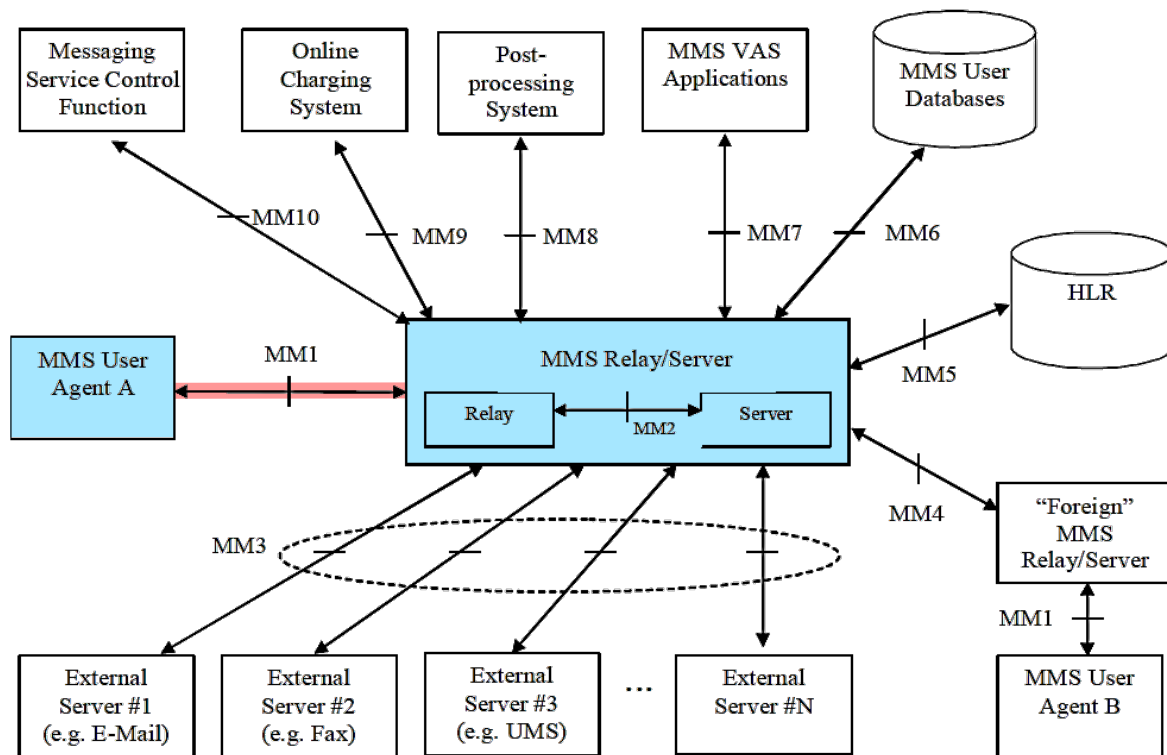


Figure 2: MMS Architectural Elements
EX-1004, 17, FIG. 2 (annotated)

Above, “Figure 2 shows that multimedia messaging may encompass many different network types.” EX-1004, 17. According to TS-23.140, “[t]he basis of *connectivity* between these different networks shall be provided by the *Internet* protocol and its associated set of messaging protocols,” which “enables messaging in 2G and 3G wireless networks to be compatible with messaging systems found

on the *Internet*.” *Id.*; *see also id.*, 18 (“MMS *connectivity* across different networks (MMSEs) is provided based on *Internet* protocols.”).

108. In the figure above, multiple “MMS User Agent” UEs are illustrated—including one (top left) that is configured for communications with the MMS Relay/Server through a connection to “2G Mobile Network A,” and another (bottom left) that is configured for communications with the MMS Relay/Server through a connection to “3G Mobile Network A.” EX-1004, 14, 17-19, 23-24 (each discussing 2G and 3G). A POSA understood these connections are part of the respective interface MM1 for each user device which facilitates the “*Internet* protocol” communications between each user device’s MMS User Agent and the MMS Relay/Server. EX-1004, 17, 23-24, FIG. 4 (Protocol Framework to provide MMS). Interface MM1 is annotated red in FIG. 3 below:



EX-1004, 23, FIG. 3 (annotated)

109. The modem in MMS-Ogawa’s user device also enables an MMS User Agent to exchange messages containing *data* with network elements through interface MM1—including with MMS-Ogawa’s MMS Relay/Server, and multiple MMS VAS applications (via the MMS Relay/Server). EX-1004, 14 (definition of application data), 18 (4.3 describing addressing for messages), 23-24 (depicting/describing FIG. 3). For example, using the modem, the MMS User Agent may “transport *data* specific to applications other than MMS” (e.g., in “abstract messages”) to or from an MMS VAS Application. EX-1004, 54-55; *see generally id.*, 17 (FIG. 2, showing/describing MMS architectural elements), 54-56 (Section 7.1.18, Support for transporting Application Data).

110. Thus, a POSA understood that MMS-Ogawa’s modem (*see* my discussion above in Section VII.C.1)—which, as I discussed, e.g., in paragraph 106 above, is a “WWAN” modem, as claimed—is used “to exchange **Internet data** via a connection to” MMS-Ogawa’s 2G or 3G WWAN, “when configured for and connected to” each of those respective networks. A POSA understood that the data exchanged is “Internet data” because it is exchanged across the Internet via an Internet protocol.

111. Moreover, MMS-Ogawa’s WWAN modem is “to exchange Internet data via **a first WWAN, when configured for and connected to the first WWAN,**” as claimed. During prosecution, PO stated that “[t]he modifier ‘first’ is merely indicative that this is an identifiable WWAN to which the modem can connect, and there could be several.” EX-1002, 730. In MMS-Ogawa, the 2G and 3G networks are each an identifiable WWAN to which MMS-Ogawa’s user device can connect, when configured for and connected to it. Thus MMS-Ogawa meets Element [1A].
Id.

- c. **[1B1] “a device messaging agent to receive secure Internet data messages, on behalf of a plurality of software applications capable of execution on the device, and over a secure connection to a network message server reachable via the WWAN,”**

112. MMS-Ogawa meets each limitation in Element [1B1], as I discuss below. I note that in IPR2024-00341, the Board found that MMS-Ogawa met substantially the same limitation in the '733 Patent. EX-1022, 36, §II.D.3(i).

- i. **“a device messaging agent to receive secure Internet data messages”**

113. The '403 specification uses the term “agent” to refer to a component—which may, e.g., be “implemented... entirely in software” (EX-1001, 42:48-49)—that performs some function on behalf of an entity (e.g., client, server). EX-1001, 15:58-16:12 (describing device agents that perform service policy implementation or management functions), 42:48-55 (agents talk to one another); *see also* EX-1029, 12 (computer science dictionary definition of agent). The specification does not set forth any *required* function. EX-1001, 16:2-12 (“It will be apparent to those of ordinary skill in the art that the division in functionality between one device agent and another is a design choice...”). In IPR2024-00341, the Board agreed that the specification “uses the term ‘agent’ to include components implemented ‘entirely in software’ to perform a function on behalf of a client or server.” EX-1022, 27.

114. Element [1B1] requires that the claimed “device” comprise a “*device messaging agent*.” The ’403 specification never uses the term “device messaging agent.” However, PO stated during prosecution that “service control device link 1691” is an “embodiment” of the claimed “device messaging agent.” EX-1002, 731. The ’403 specification says “service control device link 1691” may perform a “central agent communication hub function” (EX-1001, 42:9-15, 44:62-45:5) and “provides the device side of a system for transmission and reception of” messages, for example “service agent” messages “to/from network element functions” (EX-1001, 37:40-43). PO’s statements during prosecution indicate that the term “*device messaging agent*” includes an *agent* (on a *device*) that is used for *messaging*—for example receiving and transmitting of messages to/from a network element. EX-1002, 731. For example, PO argued that “the specification envisions such an agent being useful for more than just service control, e.g., also useful to manage ‘a set of secure messages that allow communication between certain agents or service processor functions and entities outside of the service processor operating environment.’” *Id.*⁵

⁵ To support this argument, PO cited para. [00184] of the as-filed application (which can be found at EX-1002, 65). EX-1002, 731. As I noted earlier in paragraph 39, the language PO quoted in its remarks during prosecution are often

115. MMS-Ogawa’s “MMS User Agent” is a device-side “application” (i.e., implemented in software) on TS-23.140’s user device that receives and transmits messages to/from the MMS Relay/Server over MM1 and “*performs MMS-specific operations on a user’s behalf and/or on another application’s behalf*” (see my discussion above in Section VII.A, e.g., at paragraphs 45-50)—e.g., transportation of application-specific data to/from third-party MMS VAS Applications via the MMS Relay/Server. EX-1004, 14 (setting forth the standard’s definitions, including for “abstract message,” “application data,” “MMS Relay/Server,” “MMS User Agent,” and “MMS VAS Application”), 19 (Section 5.1.1, describing MMS User Agent operations, including “transport of application data”), 23-24 (MMS architecture, showing interface MM1 between the MMS User Agent and the MMS Relay/Server), 30-31 (Section 7.1.3.1, Terminal Capability Negotiation, which includes indicating whether the “MMS User Agent supports transporting application data”), 35-36 (Section 7.1.7, Support for Streaming in

from different paragraph numbers in the as-filed application. The quoted language—“a set of secure messages that allow communication between certain agents or service processor functions and entities outside of the service processor operating environment”—is actually from para. [0181], which can be found at EX-1002, 63, and which corresponds to EX-1001, 41:59-42:18.

MMS, and describing types of data streaming that may be support), 41-42 (Section 7.1.13, Support for Value Added Services (VAS) in MMS), 54-56 (Section 7.1.18, Support for transporting Application Data).

116. MMS-Ogawa’s application-specific data is sent in what TS-23.140 calls “abstract *messages*.” EX-1004, 14 (“abstract messages: information which is transferred between two MMS entities used to convey an MM and/or associated control information between these two entities”). MMS-Ogawa’s User Agent—which is an application on the MMS User Agent device that “performs MMS-specific operations on a user’s behalf and/or on another application’s behalf” (EX-1004, 14) and transports application data in the form of abstract messages (EX-1004, 14, 19, 54-56)—is thus a “*device messaging agent*,” as claimed. This is consistent with the Board’s finding in IPR2024-00341 that MMS-Ogawa’s MMS User Agent is “a *service control device link* agent on the end-user device” (EX-1022, 17-18) which is what PO identified as the “device messaging agent” during prosecution (EX-1002, 731).

117. Further, MMS-Ogawa’s device messaging agent is used “*to receive secure Internet data messages*,” as claimed. The ’403 specification does not use the term “secure Internet data messages,” or require “Internet data messages” to be secured in any specific way. Instead, the ’403 specification leaves to a POSA how

communications are “secur[ed], sign[ed], encrypt[ed] and/or otherwise protect[ed]” when sent. EX-1001, 69:22-25.

118. As I discussed above in Section VII.D.1.b for [1A] (e.g., paragraphs 106-110), MMS-Ogawa’s modem enables the MMS User Agent to exchange “*Internet* protocol” communications via interface MM1, including “abstract *messages*” with “*data* specific to applications other than MMS.” EX-1004, 54; *see also id.*, 14 (definition of application data), 17-18 (showing and describing MMS architectural elements, e.g. the MMS User Agent and MMS Relay/Server in FIG. 2), 23-24 (showing and describing FIGS. 3 and 4, which include implementation details regarding interface MM1 between the MMS User Agent and MMS Relay/Server), 54-56 (7.1.18, Support for transporting Application Data). MMS-Ogawa’s device messaging agent is thus used to “receive... Internet data messages,” as claimed.

119. Moreover, in MMS-Ogawa, the Internet data messages received over MM1 are “*secure[d]*” in two ways: (1) using SSL/TLS, because MMS-Ogawa’s MM1 interface between the MMS User Agent on the user device and the MMS Relay/Server (over which the messages are sent) is secured using SSL/TLS (*see* my discussion above in Section VII.C.2, e.g., paragraphs 64-69); and (2) using symmetric MMS-Ogawa Message Encryption which employs a shared encryption key as taught by Ogawa to further secure messages beyond SSL/TLS (*see* my

discussion above in Section VII.C.3, e.g., paragraphs 71-81, and Section VII.C.4, e.g., paragraphs 83-85). MMS-Ogawa's MMS User Agent is thus "***a device messaging agent to receive secure Internet data messages,***" as claimed.

ii. **"on behalf of a plurality of software applications capable of execution on the device"**

120. Element [1B1] requires the claimed "device messaging agent" to "receive secure Internet data messages, ***on behalf of a plurality of software applications capable of execution on the device.***" During prosecution, PO stated that "'on behalf of' is used in its ordinary meaning indicating operation in a proxy capacity," and that "on behalf of a plurality of software applications capable of execution on the device" "merely indicates that the secure Internet data messages are directed to the device messaging agent, with the expectation that the device messaging agent will deliver the internal contents of the messages to the appropriate processes on the device based on the application ID(s) in each message." EX-1002, 732.

121. As I discussed above in Section VII.D.1.c.i, MMS-Ogawa's MMS User Agent application—i.e., the claimed "device messaging agent"—"performs MMS-specific operations on a user's behalf and/or ***on another application's behalf.***" EX-1004, 14 ("MMS User Agent" definition). Among these operations is the "transport [of] data specific to ***applications***"—i.e., a ***plurality*** of applications—"other than the MMS User Agent," that are also on MMS-Ogawa's user device—

for example, a “chess *application*” that “initially need[s] to register with the appropriate MMS User Agent” in order to send and receive such data. EX-1004, 14, 54-56 (*e.g.*, Section 7.1.18.1); EX-1028, 732-733 (Section 2.2, describing presence of multiple applications residing on a user device terminal, implemented to use MMS for transportation of application data). TS-23.140 discloses, for example, “received MMS information” (from, *e.g.*, an MMS VAS Application) being “immediately route[d]” by the MMS User Agent “on to the destination application... without presentation to the user.” EX-1004, 56 (Section 7.1.18.2.2).

122. MMS-Ogawa’s “other” destination applications (EX-1004, 14) are separate from (and in communication with) the user device’s MMS User Agent application. EX-1028, 732-733 (contemplating, in the context of MMS, the presence of “multiple [destination] applications” on a mobile terminal); *see also* my discussion above in Section VII.C.5, paragraphs 87-92. The Board in IPR2024-00341 agreed. EX-1022, 30-31 (“[W]e find that an application residing on an MMS User Agent would also be understood as being differentiated from the MMS User Agent,” and that “TS-23.140 discloses that its MMS User Agent and destination applications can be separate and in communication with each other....”). Confirming this is the fact TS-23.140 discloses that the destination applications may be “downloadable... to a mobile phone” or “integrat[ed] into a

mobile phone” through an “application registration process.” EX-1004, 54 (Section 7.1.18, Support for transporting Application Data).

123. Based on TS-23.140’s disclosures regarding destination applications—e.g., what they do (e.g., provide chess or news or weather services; EX-1004, 14, 55) and how they are added to a user device (e.g., by downloading to or integrating into a mobile phone; EX-1004, 54)—a POSA understood that TS-23.140’s multiple destination applications (a plurality, e.g., chess, news, and weather) run on the device and are therefore “a plurality of *software* applications *capable of execution on the device*,” as claimed. *See* EX-1046, 46 (defining *application*: “A *software* program consisting of one or more processes and supporting functions.”), *id.* (defining *application* program: “A program *executed* with the processor in user mode.”); EX-1045, 126 (defining *execute*: “to run or carry out a computer program or process”), *id.* (defining *execution*: “the process of carrying out a computer program or process”).

124. Thus, a POSA would have understood that MMS-Ogawa’s MMS User Agent—which “performs MMS-specific operations... on another application’s behalf” (EX-1004, 14) and “transport[s] data specific to [other] applications” (*id.*) on the user device to/from MMS VAS Applications via MM1 and the MMS Relay/Server—is a “device messaging agent” that “receive[s] secure Internet data

messages” (see my discussion above in Section VII.D.1.c.i) “*on behalf of a plurality of software applications capable of execution on the device*” as claimed.

iii. “and over a secure connection to a network message server reachable via the WWAN”

125. Element [1B1] also requires that the claimed “device messaging agent” “receive secure Internet data messages” “*over a secure connection to a network message server reachable via the WWAN.*”

126. The ’403 specification never uses the term “network message server.” However, PO stated during prosecution that “service control server link 1638”—which may “provide[] the network side of a system for transmission and reception” of messages—is an embodiment of the claimed “network message server.” EX-1002, 732-733, citing para. [00259] of the as-filed application, but quoting language that appears to correspond to para. [0256] of the as-filed application (EX-1002, 101) and EX-1001, 68:9-48. PO characterized the omission of “messages” in the language it quoted as a “typographical error.” EX-1002, 732-733. PO’s statements during prosecution indicate that a “*network message server*” includes a *server* (on a *network*) that is used to relay *messages* between a device and another network element. EX-1001, 16:13-16 (service controller 122 includes one or more server functions), 68:9-48 (describing functionality of the service control server link 1638 of FIG. 16).

127. As discussed above in Section VII.D.1.b, MMS-Ogawa’s “MMS Relay/*Server*” is part of the MMSE (which TS-23.140 describes as a “*network*”) and can be reached by the MMS User Agent over the WWAN (e.g. a 2G or 3G wireless network). EX-1004, 17-18 (equating “networks” with “MMSEs”: “MMS connectivity across different *networks (MMSEs)*....”), FIG. 2. Moreover, as discussed above in Section VII.A, TS 23.140 discloses that the MMS Relay/Server relays messages between the MMS User Agent and MMS VAS Applications using various communication interfaces between the MMS Relay/Server and other elements in the MMSE—e.g., MM1 (EX-1004, 17-18, 21, 23-25) for communications between a MMS User Agent and the MMS Relay/Server (*id.*, 23-24, FIG. 4) and MM7 for communications between the MMS Relay/Server and MMS VAS Applications (*id.*, 14, 18, 23-26, 41). *See* my discussion in paragraphs 44-46 above. A POSA thus understood that MMS-Ogawa’s MMS Relay/Server is a “*network message server reachable via the WWAN,*” as claimed. This understanding is consistent with the Board’s finding in IPR2024-00341 that MMS-Ogawa’s MMS Relay/Server is “*a service control server link* element of the network system” (EX-1022, 34-36) which is what PO identified as the “network message server” during prosecution (EX-1002, 732-33).

128. A POSA likewise understood that communications between MMS-Ogawa's MMS User Agent and MMS Relay/Server occur "***over a secure connection,***" as claimed, as discussed below in paragraphs 129-131.

129. While the '403 specification does not use the term "secure connection," or require the connection between the claimed network message server and the claimed device messaging agent to be secured in any particular way, the specification describes a "service control link 1653" as the connection between what PO identified as a network message server (service control server link 1638 of service controller 122) and what PO identified as a device messaging agent (service control device link 1691 of service processor 115) as "provid[ing] for a ***secure (e.g., encrypted)*** communications link for providing secure, bidirectional communications..." EX-1001, 68:18-27; FIG. 16.

130. As I discussed above in Section VII.D.1.c.i, the '403 specification leaves to a POSA precisely how "the service control server link 1638 provides for ***securing, signing, encrypting and/or otherwise protecting*** the communications before sending such communications over the service control link 1653." EX-1001, 69:22-25. For example, the '403 specification describes use of "***secure transport protocols running over Transmission Control Protocol (TCP)***," and notes that "approaches for implementing a secure control channel over the Internet includ[e]... running TCP Transport Layer Security (TLS)..." EX-1001, 17:2-24

(“...TCP provides a more reliable delivery channel for control traffic that is not as sensitive to delay or jitter.”); *see also id.*, 69:25-30 (describing the service control server link 1638 role in securing communications), 87:55-62 (various known security encryption techniques can be used). The specification uses “TLS” and Secure Socket Layer (SSL) interchangeably. *E.g.*, EX-1001, 94:3 (“TLS/SSL”), 98:30 (“TLS/SSL”); 99:15-16 (“TLS/SSL”); 101:52-55 (providing “basic TCP setup, TLS/SSL” as example protocols used in “socket assignment and session management” layer in FIGS. 31-37).

131. As I discussed above in Section VII.D.1.c.i, in MMS-Ogawa, the MMS User Agent (the claimed “device messaging agent”) communicates with the MMS Relay/Server (the claimed “network message server”) over interface MM1. And as I discussed above at Section VII.C.2, MMS-Ogawa’s MM1 interface is secured using SSL/TLS. *See also* EX-1004, 24-25 (describing the MM1 interface between the MMS User Agent and the MMS Relay/Server); EX-1012, 21 (“[t]he ***TLS*** [WP-TLS] transport layer security protocol ***provides for secure data transmission between the MMS Client and the MMS Proxy-Relay*** in architectural configurations that employ HTTP based protocol stacks for MMSM implementation”). MMS-Ogawa’s messages are also encrypted using symmetric MMS-Ogawa Message Encryption (*see* my discussion above in Section VII.C.3) when sent using MMS, including when transmitted over MM1.

132. MMS-Ogawa’s MM1 is thus a “secure connection,” and the secure Internet data messages received from the MMS Relay/Server over MM1 (*see* my discussion above in Section VII.D.1.c.i) are received “***over a secure connection to***” the “***network message server reachable via a WWAN,***” as claimed. This is consistent with the Board’s finding in IPR2024-00341 that MM1 between MMS-Ogawa’s User Agent and Relay/Server is a “service control *link,*” *i.e.*, connection, “***secured*** by an encryption protocol.” EX-1022, 16-17.

- d. **[1B2] “wherein at least a subset of the secure Internet data messages contain an identifier for a corresponding one of the software applications and application data from a respective network application server corresponding to that application; and”**

133. MMS-Ogawa meets each limitation in Element [1B2], as I discuss below. I note that in IPR2024-00341, the Board found that MMS-Ogawa met substantially the same limitation in the ’733 Patent. EX-1022, 37-39, §§II.D.3(k)-II.D.3(l).

- i. **“wherein at least a subset of the secure Internet data messages contain an identifier for a corresponding one of the software applications”**

134. As discussed above in Section VII.D.1.c.i (paragraphs 113-119), in MMS-Ogawa, MMS is “used to transport data specific to applications” on the user device that are not the MMS User Agent. EX-1004, 54-56; EX-1028, 732-733; *see also* my discussion above in Section VII.A. As discussed above in Section

VII.D.1.c.ii (paragraphs 120-124), MMS-Ogawa's destination applications are "software applications," as claimed.

135. MMS-Ogawa's MMS User Agent receives application data from the MMS Relay/Server via MM1, in the form of "abstract messages" that each include an "*application identifier of the destination application.*" EX-1004, 14 ("[a]bstract messages": "information which is transferred between two MMS entities used to convey an MM and/or associated control information between these two entities."), 54-56 ("Abstract messages that are sent by an MMS User Agent or an MMS VAS Application on behalf of an originating application shall contain *a destination application identifier*. They may, in addition, ...contain additional application/implementation specific control information."). TS-23.140 discloses, as an example of a destination application, a "*chess*" application. EX-1004, 55.

136. MM1-retrieve.RES is an example of an abstract message with a "destination application identifier" as well as "MMS control information and the MM content." *Id.*, 56, 60 (FIG. 6, showing "Example Abstract Message Flow," including an MM1_retrieve.RES abstract message going from MMS Relay/Server to MMS User Agent), 69 (Table 10, showing "Abstract messages for retrieval of MM in MMS," including MM1_retrieve.RES abstract message sent from MMS Relay/Server to MMS User Agent), 72-73 (Table 12, showing various

“information elements” in MM1-Retrieve.RES abstract message, including Applic-ID, described as “Identification of the destination application” and Content), 111 (showing “MM1_Retrieve.RES” abstract message in the context of supporting Value Added Services from Value Added Service Providers). The “application identifier” is the identifier that allows the MMS User Agent to “immediately route[]” a received application-specific message to the *corresponding one of the software applications*, *i.e.*, the “destination application that is referred to from the destination application identifier.” EX-1004, 56.

137. During prosecution, PO stated that the “*at least a subset*” term “is intended to indicate that the secure Internet data messages sent to the device messaging agent can be sent for multiple purposes, only one of which is to provide data to applications.” EX-1002, 733. PO also stated that the claimed “*identifier* is a value that the device messaging agent and network message server both associate with an agent/function/process/application on the device.” *Id.* Because in MMS-Ogawa, one of MMS’s uses is to transport data specific to applications (as I discussed above in, *e.g.*, Section VII.D.1.c.i (paragraphs 113-119))—and because this routing takes place using an application identifier “present in an abstract message” (EX-1004, 55) that is sent by the MMS Relay/Server to the MMS User Agent—a POSA understood that “*at least a subset of*” MMS-Ogawa’s “*secure*

Internet data messages” (which I discussed in Section VII.D.1.c.i above) *contain an identifier for a corresponding one of the software applications,*” as claimed.

ii. **“and application data from a respective network application server corresponding to that application”**

138. As discussed above in Sections VII.D.1.c.i-VII.D.1.c.ii, MMS-Ogawa’s secure Internet data messages (including the “abstract messages” subset that I discussed in Section VII.D.1.d.i above) are “used to transport data specific to applications” other than the MMS User Agent. EX-1004, 54-56; EX-1028, 732-733; *see also* my discussion above in Section VII.A. TS-23.140 expressly calls such data “application data.” EX-1004, 14 (defining “Application Data”), 54-56 (disclosing “transporting” such “Application Data” in “abstract messages”). A POSA thus understood that TS-23.140’s application-specific data—which is transported in MMS-Ogawa’s messages (e.g., abstract messages) to specific applications “without alteration” (EX-1004, 14, 54-56)—constitute “*application data*” as claimed.

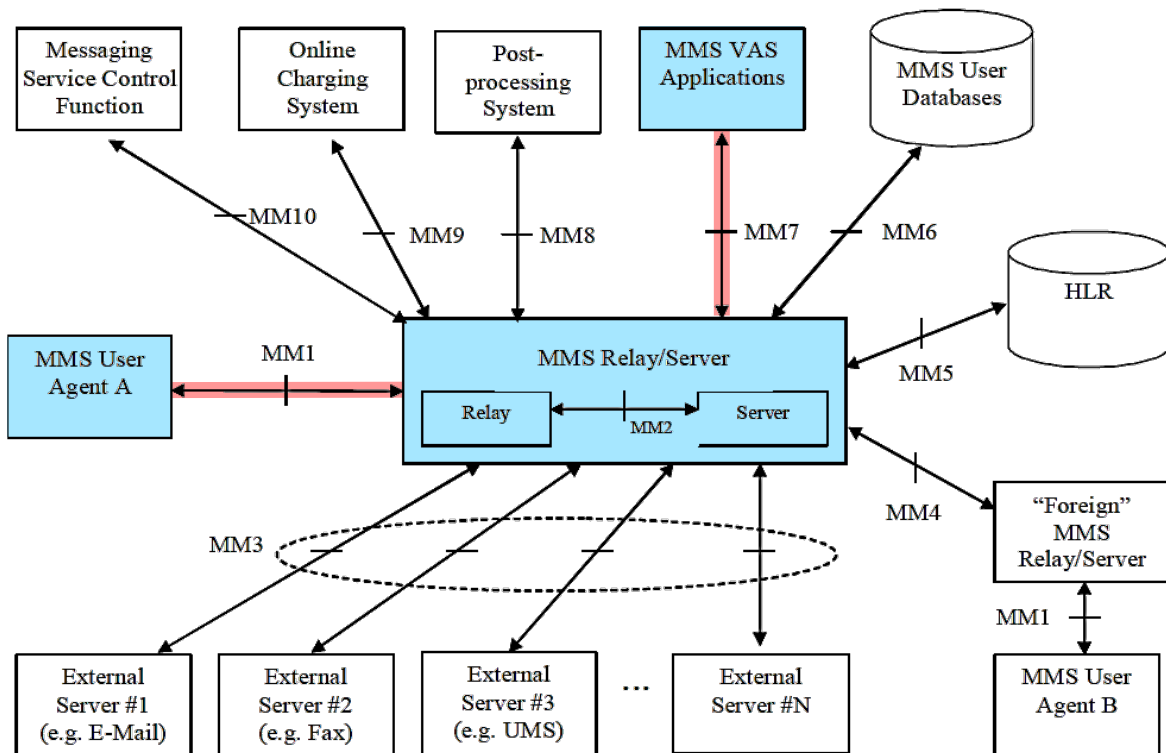
139. Further, as discussed below, a POSA understood that the “at least a subset of” MMS-Ogawa’s secure Internet data messages (which I discussed in Section VII.D.1.d.i above) “contain... *application data from a respective network application server corresponding to that application.*” The ’403 specification never uses the term “network application server.” Based on the claim’s plain

language, a POSA understood that the term “network application server” includes a server (or other network element) that interacts with an application (or agent) on a user device to deliver some service to a user. *Cf.* EX-1001, 86:59-87:5 (broadly describing “service controller elements” that “interact with... certain network elements, and/or the service processor agents... to form a reliable device based on service delivery solution and/or platform.”).

140. As discussed above in Section VII.D.1.c.i, MMS-Ogawa’s messages containing application data (including the “abstract messages” subset I discussed in Section VII.D.1.d.i above) include data from third-party MMS VAS Applications. EX-1004, 41 (“The MMS Relay/Server may support services... provided... by third-party [VASPs]. ... Messages originated from the VASP may be targeted to the recipient...”). These messages contain application data “of any content type and format,” and are used to “provid[e] Value Added *Services* (e.g. news service or weather forecasts) to MMS users.” EX-1004, 14; *see also* my discussion above in Section VII.A.

141. The MMS Relay/Server can relay messages from “several MMS VAS [Value Added Services] Applications” in the MMS network environment. *Id.*, 18; *see also id.*, 23 (MMS reference architecture showing communication between MMS Relay/Server and MMS VAS Applications). Interface “MM7 is used to transfer MMs from MMS Relay/Server to MMS VAS applications and to transfer

MMs from MMS VAS applications to MMS Relay/Server.” EX-1004, 25 (Section 6.9). Received messages from MMS VAS Applications are “targeted to the recipient” MMS User Agent by the MMS Relay/Server, through MM1. *Id.*, 41 (Section 7.1.13). The path of application data between the MMS VAS Applications and the MMS User Agent is annotated red below:

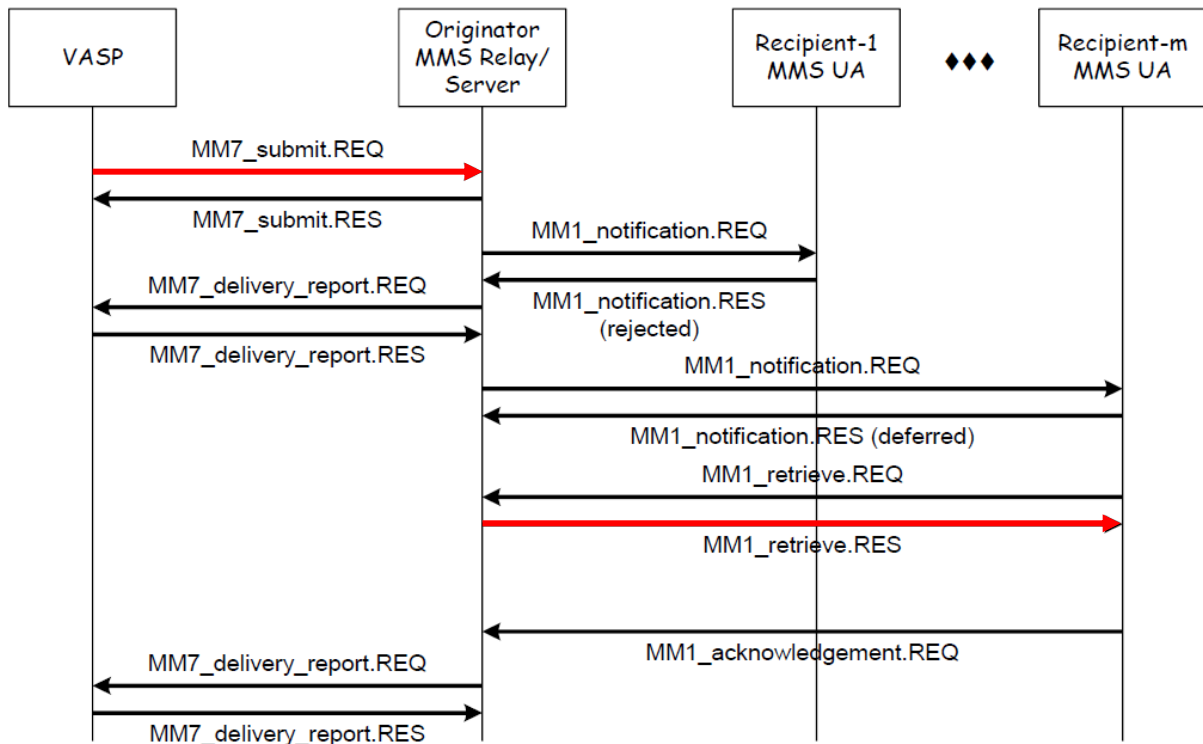


EX-1004, 23, FIG. 3 (annotated)

Above, application data is transmitted by VAS applications to the MMS Relay/Server via interface MM7, then relayed “without alteration” (EX-1004, 14) to the MMS User Agent via interface MM1 and then routed internally to a specific destination application. *See* EX-1004, 14 (defining application data), 18 (Section

4.3, Addressing between entities), 23-25 (showing FIG. 3 and explaining the interactions between the entities), 41 (Section 7.1.13, Support for Value Added Services (VAS) in MMS), 54-56 (Section 7.1.18, Support for transporting Application Data).

142. FIG. 8 (annotated) below helps illustrate how message content is transmitted from the VASP to a recipient MMS User Agent, using TS-23.140's exemplary MM7_submit.REQ and MM1_retrieve.RES abstract messages.



**Figure 8. Sample data flow of MM7 message distribution
EX-1004, 111, FIG. 8 (annotated)**

Above, MM7_submit.REQ includes content from the VASP and an Applic-ID. These same information elements are conveyed to the MMS User Agent in MM1_retrieve.RES. *See* EX-1004, 111-116 (describing VASP-added “Content” and “Applic-ID... indicat[ing] that... abstract message shall be provided to an application residing on an MMS User Agent” in MM7_submit.REQ), 69-73 (describing “Content” conveyed from originator and “Applic-ID” in MM1_retrieve.RES, including Table 12, showing various “information elements” in MM1-Retrieve.RES abstract message, including Applic-ID and Content).

143. The Board in IPR2024-00341 agreed with Petitioner that “application-specific data from VAS applications” is “provided” to user agents “by VAS providers communicating through the MMS Relay/Server.” EX-1022, 38 (citing EX-1004, 23-24, 25-26, 41, 112, FIG. 3).

144. Because MMS-Ogawa’s “MMS VAS Applications” serve messages to the MMS Relay/Server that contain *application-specific* data that is then transported to specific applications on the user device by the MMS User Agent (as I discuss in paragraphs 138-143 above), MMS-Ogawa’s “MMS VAS Applications” are “*network application servers*,” as claimed. The MMS VAS Application is a “server” because it serves multiple applications and user devices—

for example, the MMS VAS Applications interact with agents on the user device (e.g., MMS User Agent) to deliver services to the user, such as delivering application-specific services in the form of application-specific data.

145. In MMS-Ogawa, an MMS VAS Application (a network-side component) sends targeted messages to a specific destination application on the user device (a device-side component) to provide a specific “Value Added *Service[]*” e.g., “weather forecasts” or “chess.” EX-1004, 14 (defining application data), 18 (4.3, Addressing between entities), 23 (showing FIG. 3, which shows that the path of data between the MMS VAS Applications and the MMS User Agent is through interface MM7, the MMS Relay/Server, and MM1), 41 (7.1.13, Support for Value Added Services (VAS) in MMS). TS-23.140 explains that “applications that intend to transport application specific data using MMS” “initially need to register with the appropriate MMS User Agent or MMS VAS Application” and “negotiate... the details... of information to be exchanged between the two entities.” EX-1004, 54. This, for example, “may... be the initial step after the download of a downloadable application to a mobile phone.” *Id.* TS-23.140 thus teaches a scenario in which a specific destination application receives targeted data from a specific network-side application registered with an MMS VAS Application—e.g., a scenario in which a downloaded weather application that receives messages from a specific weather VASP, or a downloaded chess

application that receives messages from a specific chess VASP. *Id.* Based on this, a POSA understood that the “abstract messages” subset of MMS-Ogawa’s secure Internet data messages (that I discussed in Section VII.D.1.d.i above) contain “application data *from a respective network application server corresponding to that application,*” as claimed, because the specific (network-side) application registered with the MMS VAS Application *corresponds to* the destination application on the device.

e. [1C1] “a secure interprocess communication service,”

146. The ’403 specification never uses the term “interprocess communication service.” However, PO stated during prosecution that “agent communication bus 1630” is an example of the claimed “secure interprocess communication service” and that “[i]n various embodiments, the secure interprocess communication service is embodied as ‘an inter-process software communication bus,’ which can be ‘*a variant of D-bus* (e.g., a message bus system for inter-process software communication that, for example, helps applications/agents to talk to one another),’ or ‘another inter-process communication protocol or system, running a session bus in which all communications over the session bus can be *secured*, signed, encrypted or otherwise protected.’” EX-1002, 734; *see also* EX-1001, 41:42-43:4.

147. As discussed above at Section VII.C.5, TS-23.140 discloses its MMS User Agent receiving “data specific to an application other than the MMS User Agent” and “rout[ing]” it “to [a] destination application” on the user device over *some* interface. EX-1004, 14, 54-56. In IPR2024-00341, PO was unable to “dispute that a destination application is in communication with an MMS User Agent,” and the Board found that a POSA “would have... understood that [TS-23.140’s] destination applications would have been communicatively coupled to the MMS User Agent by ‘an agent communication bus,’ as that term is used in the ’733 patent, because they are, in the words of the ’733 patent, ‘talk[ing] to one another.’” EX-1022, 28-29; *see also id.*, 18-32; EX-1001, 42:51-58. For this reason, based on the specification and PO’s above prosecution statements—stating that the claimed “interprocess communication service” may be “embodied as ‘an inter-process software communication bus,’” (EX-1002, 734)—a POSA understood MMS-Ogawa includes a “interprocess communication service,” as claimed. EX-1001, 41:42-43:4; EX-1002, 734.

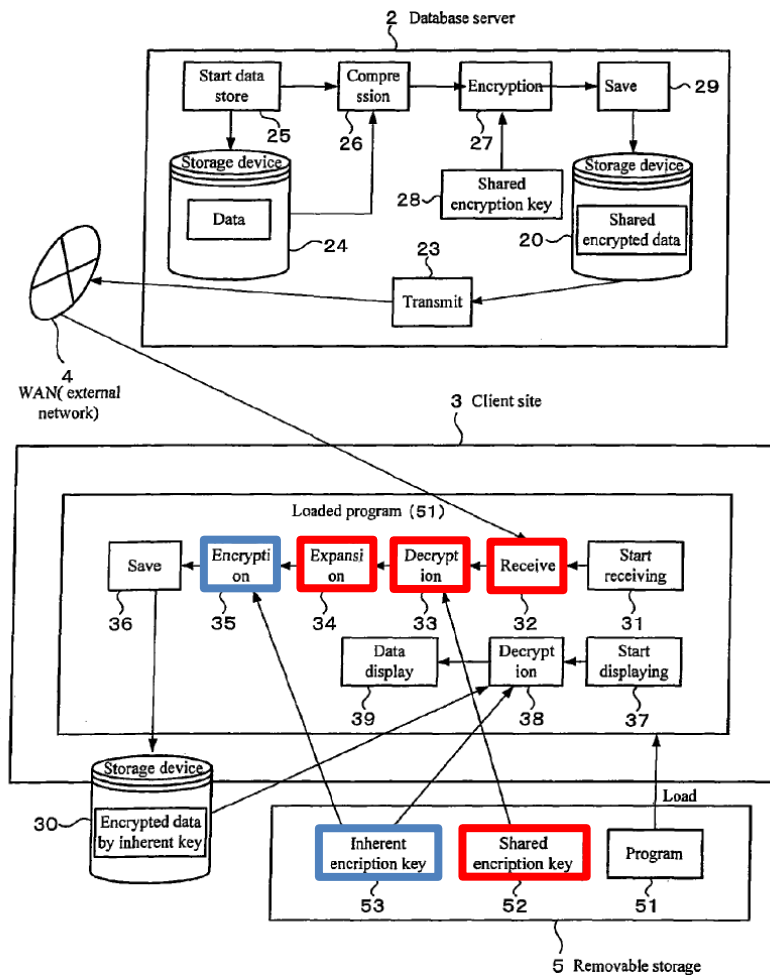
148. Separately, as discussed above in Section VII.C.5, it also would have been obvious to implement the interface between MMS-Ogawa’s MMS User Agent application and the user device’s destination applications using a software *bus* for interprocess communications—e.g. a “D-bus.” *See, e.g.*, EX-1031, 10:56-62 (disclosing a D-bus). As I noted above in Section VII.C.5 (see paragraph 92),

the Board in IPR2024-00341 agreed this was obvious. EX-1022, 48-49. For this alternative reason, again based on the specification and PO's prosecution statements (EX-1002, 734), MMS-Ogawa includes an interprocess communication service.

149. Element [1C1] further requires that the “interprocess communication service” be “*secure*.” When describing “agent communication bus 1630”—which PO cited when identifying support for the “inter-process software communication bus” term (EX-1002, 733-734)—the '403 specification states that the bus can be secured using “point[-] to[-]point” or “bus-level” “message exchange encryption using one or more keys that are partially shared or shared within the service processor 115 agent group” on the device-side. EX-1001, 41:59-42:18, FIG. 16 (depicting 115, 1630). A POSA thus understood that an “interprocess communication service” secured by shared or partially shared encryption keys was “secure,” as claimed.

150. As discussed above in Section VII.C.6, MMS-Ogawa's interprocess communications are secured using Ogawa's inherent key when data is routed by the MMS User Agent over TS-23.140's software *bus* for interprocess communications (*see* my discussion above in Section VII.C.5) to a destination application. Ogawa discloses an encryption unit for re-encrypting data for, e.g., transmission within the user device. EX-1005, 5:59-6:14. In Ogawa, a message

received and decrypted at the client (using receive unit 32 and decryption unit 33) is converted back to its original format (using decompression/expansion unit 34), and encrypted again—before being transmitted within the client device, e.g., for storage—by encryption unit 35, using an “inherent encryption key 53.” EX-1005, 5:59-6:26, 5:24-27, FIG. 7 (annotated below).



EX-1005, FIG. 7 (annotated)

151. MMS-Ogawa thus includes “a *secure* interprocess communication service,” as claimed, because communications through the bus to the destination application are encrypted by the inherent key.

- f. **[1C2] “wherein the device messaging agent, for each message in the subset of the secure Internet data messages, maps the identifier to the corresponding one of the software applications in order to forward the application data on the secure interprocess communication service to a software process corresponding to the identified software application.”**
- i. **“wherein the device messaging agent, for each message in the subset of the secure Internet data messages, maps the identifier to the corresponding one of the software applications”**

152. I note that in IPR2024-00341, the Board found that MMS-Ogawa met substantially the same limitation in the ’733 Patent. EX-1022, 39-40, §II.D.3(m).

153. Element [1C2] requires “the device messaging agent” to “*map*” each “identifier” recited in Element [1B2] “to the corresponding one of the software applications.” The ’403 specification uses “mapping” to describe an “association.” EX-1001, 79:1 (“a mapping (e.g., an association)”). A POSA thus understood that “map” in element [1C2] encompasses the device messaging agent *associating* the correct software application with the received identifier when forwarding the application data to the destination application.

154. That is precisely what MMS-Ogawa’s MMS User Agent does. As I explained above in Section VII.D.1.d.i, MMS-Ogawa’s MMS User Agent “transport[s] *data* specific to *applications*” contained in application-specific messages (including the “abstract messages” subset I discussed above in Section VII.D.1.d.i) by “rout[ing]... received MMS information on to the” correct

“destination application” based on a “destination application identifier” that is included in the received message. EX-1004, 14, 55-56; *see also* my discussion in paragraphs 49 and 135-137 above.

155. TS-23.140 expressly describes the correct “destination application” as being “*referred to from* the destination application identifier.” EX-1004, 56. A POSA would have understood that the MMS User Agent’s determination of which application is “*referred to*” by the identifier constituted the MMS User Agent associating the correct destination application with the received destination application identifier. A POSA thus understood that the MMS User Agent (the claimed “*device messaging agent*,” *see* my discussion above in Section VII.D.1.c.i), “*for each message in the subset of the secure Internet data messages*” (including each abstract message discussed above in Section VII.D.1.d.i), “*maps*” the destination application identifier of the destination application (“*the identifier to the corresponding one of the software applications*”) to the destination application “referred to from” the destination application identifier (“*to the corresponding one of the software applications*,” as discussed above in Section VII.D.1.d.ii), as claimed. A POSA understood that TS-23.140 contemplates that each of the application-specific messages that transport application data to specific applications are handled this way when the message contains a valid application ID. *See* EX-1004, 55-56 (Section 7.1.18.2.2).

ii. “in order to forward the application data on the secure interprocess communication service”

156. Element [1C2] further requires that the claimed mapping be done “in order to forward the application data on the secure interprocess communication service.” In MMS-Ogawa, MMS is used to “transport data specific to applications,” including “*application data*” from third-party MMS VAS Applications that is sent in the abstract messages (discussed in §VII.D.1.d.ii above) to provide a specific “Value Added *Service[]*” e.g., “weather forecasts.” EX-1004, 14, 18, 23-25, 41, 54-56, 69, 111-116; *see* my discussion of these pages in paragraphs 138-145 above.

157. As discussed in Section VII.D.1.e, the application data in such application-specific messages are encrypted using Ogawa’s inherent key (as I discussed above in Section VII.C.6) and “transported without alteration” of the underlying data (EX-1004, 14)—i.e., forwarded—by the MMS User Agent to the correct destination application over a “secure interprocess communication service” (i.e., over a bus). A POSA thus understood that MMS-Ogawa’s “mapping” described above is done “*in order to forward*” the claimed “*application data* on the *secure interprocess communication service*,” as claimed.

iii. “to a software process corresponding to the identified software application.”

158. Element [1C2] finally requires that the claimed forwarding be directed “*to a software process* corresponding to the identified software application.” The ’403 specification does not use the term “software process” or discuss a “software process corresponding to the identified software application.” However, a POSA understood that TS-23.140’s forwarding of application data to destination applications constitutes forwarding the data “to a software process corresponding to the identified software application,” because a POSA understood that TS-23.140’s destination applications consisted of one or more processes in which specific functions of the application (e.g., receiving data) are executed. *See* EX-1046, 46 (defining application: “a software program consisting of one or more processes and supporting functions”); EX-1047, 5:41-61 (“A software application can include multiple processes...”); EX-1048, [0003]-[0004] (describing “[d]ifferent processes within an application”). For example, given a “chess” destination application, application data from an abstract message may be forwarded to, e.g., a specific process for moving the opponent’s chess pieces, or a specific “instance[]” of the “chess” application (e.g., “chess application #02”). EX-1004, 55-56. A POSA thus understood that “a software process corresponding to the destination application identified by the destination application identifier” receives the application data forwarded by the MMS User Agent.

159. To the extent [1C2] is interpreted to imply use of additional, unclaimed information *beyond* [1B2]’s “identifier for a corresponding one of the software applications”—e.g., something corresponding to a particular “software process” of a destination application that enables routing application data to that *specific* software process—TS-23.140 teaches that as well. As discussed above, MMS-Ogawa’s secure Internet data messages are in the form of TS-23.140’s “abstract messages,” which each include a “destination application identifier” (the claimed “identifier”) and the application-specific data that the abstract message is being used to transport (the claimed “application data”). EX-1004, 14, 18, 23-25, 41, 54-56, 69, 111-116. See my discussion of these pages in paragraphs 138-145 above. TS-23.140 teaches that “abstract messages” *also* include “additional application/implementation specific control information.” EX-1004, 55; *id.*, 14 (describing “associated control information”). This “additional... control information” is used for “needs... not supported by the application identifier of the destination application and the identifier of the originating application, such as specifying a particular logical channel in the application addressing method (e.g., ‘discussion thread #05’) or distinguishing between multiple instances of the same application (e.g., ‘chess application #02’).” EX-1004, 55. A POSA understood that this “additional application/implementation specific control information” that specifies a particular logical channel or application instance constitutes information

that allows MMS-Ogawa's MMS User Agent (the claimed "device messaging agent") to forward data to a *specific* "software process corresponding to the destination application identified by the destination application identifier," to the extent that is required by element [1C2].

2. Claim 3: The mobile end-user device of claim 1, further comprising the plurality of software applications.

160. Claim 3 requires claim 1's "device" to "further compris[e] the plurality of software applications" (on behalf of which the device messaging agent receives secure Internet data messages). As discussed above in Section VII.D.1.c.ii, MMS-Ogawa's device includes "a plurality of software applications capable of execution on the device" (TS-23.140's destination applications) to which application-specific data is "transport[ed]," e.g. "news service" (EX-1004, 14), "weather" (*id.*), "chess" (*id.*, 54), or "messaging" (*id.*) applications. MMS-Ogawa's user device thus "comprises the plurality of software applications," as claimed.

3. Claim 4: The mobile end-user device of claim 3, wherein the plurality of applications include a first application that receives the application data in a first format, and a second application that receives the application data in a second format different than the first format.

161. Claim 4 requires claim 3's "the plurality of applications" to "include" a "first" and "second" application that respectively "receive[] application data" in different "format[s.]" During prosecution, PO said "the use of 'first' and 'second'

applications merely indicates two... elements that receive different types of formatted data in the messages from their respective network application servers.” EX-1002, 734. TS-23.140 discloses a “[m]inimum set of supported formats,” e.g., “Plain Text,” “Audio,” “Still Image,” and “Video.” EX-1004, 20; EX-1020 (3GPP TS-26.140 (Multimedia Messaging Service; Media formats and codecs)). A POSA understood that MMS-Ogawa’s different applications (e.g., “chess,” “weather”) would receive application data in different formats specific to the respective applications, and thus would meet claim 4. For example, a POSA understood that TS-23.140’s chess application (a “first” application) would receive application data in a “first” format that is used by the chess application (e.g., a format suitable for transmitting data about an opponent’s chess move), while TS-23.140’s weather application (a “second” application) would receive application data in a “second” format that is used by the weather application (e.g., a format suitable for transmitting data about an upcoming hurricane). EX-1004, 14, 55.

4. **Claim 5: The mobile end-user device of claim 1, wherein the secure Internet data messages are received encrypted, the device messaging agent decrypting each message in the subset to obtain the corresponding identifier and application data.**

162. Claim 5 requires claim 1’s “secure Internet data messages” to be “*received encrypted*” by “the device messaging agent,” which “decrypt[s] each

message in” claim 1’s “subset to obtain” claim 1’s “corresponding identifier and application data.”

163. As explained above in Section VII.D.1.c.iii, MMS-Ogawa’s MMS User Agent (the “device messaging agent,” *see* my discussion above in Section VII.D.1.c.i) receives messages from the MMS Relay/Server over secure interface MM1. A POSA understood that such messages are “*received encrypted*,” as claimed, using *both* MMS-Ogawa Message Encryption and SSL/TLS. *See* my discussion above in Sections VII.C.2-VII.C.4.

164. In MMS-Ogawa, such received encrypted messages are decrypted *by the MMS User Agent*—which provides MMS-Ogawa’s “decryption” functionalities for encrypted MMS messages (EX-1004, 19)—using Ogawa’s decryption unit. *See* my discussion above in Sections VII.C.3-VII.C.4 above.

165. As discussed above in Sections VII.D.1.d.i-VII.D.1.d.ii for [1B2], each received message in the abstract messages subset (discussed above in Section VII.D.1.d.i) contains an “identifier” for routing the message to a destination application and “application data” that is routed. A POSA understood that the MMS User Agent “decrypt[s] each” encrypted “message in the subset” it receives “to obtain corresponding identifier and application data,” as claimed—which facilitates routing MMS-Ogawa’s application-specific data to the correct destination application.

5. Claim 6: The mobile end-user device of claim 5, wherein the secure Internet data messages are transported to the device messaging agent using one or more of encryption on a transport services stack, IP (Internet Protocol) layer encryption, and transport via a tunnel.

166. Claim 6 recites a list of alternatives (“one or more of”) for transporting the claimed “messages” to the device messaging agent, including “encryption on a transport services stack.” See EX-1002, 734 (noting “different alternatives called out in the claim”). In the ’403 specification, external communication with what PO identified as a “device message agent” (EX-1002, 731) occurs over “service control link 1653.” The ’403 specification describes including a “layer[] of encryption in the service control link” that is *“implemented in the transport services stack (2410, 2420)”* using *“standard secure or open Internet networking protocols, such as TLS or TCP.”* EX-1001, 87:40-62. As I discussed above in Section VII.C.2, in MMS-Ogawa, MM1 is implemented to utilize TLS. This TLS protocol provides secure data transmission between the MMS User Agent and the MMS Relay/Server “in architectural configurations that employ HTTP based protocol stacks” or WAP protocol stacks via the transport services stack layer. EX-1012, 21. The MM1 interface using TLS thus provides encryption at the transport services stack layer. A POSA thus understood that, just as in the ’403 specification, MMS-Ogawa’s use of the SSL/TLS protocol to secure

the MM1 interface is implemented in a transport services stack, and is “*encryption on a transport services stack*,” as claimed.

6. **Claim 11: The mobile end-user device of claim 1, wherein the device messaging agent is further to send secure upload Internet data messages to the network message server over the secure connection, wherein at least a subset of the secure upload Internet data messages are sent responsive to a corresponding request received on the secure interprocess communication channel from a corresponding one of the software applications, the device messaging agent constructing from each such request a secure upload Internet data message containing an identifier for a respective network application server corresponding to the requesting software application; and content received with the request.**

167. Claim 11 requires the “the device messaging agent” to “send secure *upload* Internet data messages to the network message server over the secure connection” in “respons[e] to a corresponding request received on the secure interprocess communication channel from a corresponding one of the software applications[.]” While “secure interprocess communication *channel*” has no antecedent basis, a POSA understood that it at least encompasses claim 1’s “secure interprocess communication *service*.”⁶ During prosecution, PO stated that “[c]laim 11 is intended to cover the reverse or upload channel that allows various device

⁶ Claims 14-15 and 17 also recite a “secure interprocess communication channel.”

applications to send data to a respective corresponding application server.” EX-1002, 735.

168. As I discussed above in Sections VII.D.1.c.i-VII.D.1.c.ii, MMS is used to transport application-specific data between MMS VAS Applications and applications on an MMS user device (other than the MMS User Agent application). EX-1004, 14, 54-55. As I discussed above in Section VII.D.1.c.iii, MMS-Ogawa’s MMS User Agent (“device messaging agent”) exchanges “secure Internet data messages” containing such application-specific data with MMS-Ogawa’s MMS Relay/Server (“network message server”) over a “secure connection” (MM1).

169. These messages go *both ways*—MMS can also be used to upload messages from the user device to a VASP (via the MMS Relay/Server). TS-23.140 expressly discloses MMS being “used to transport data specific to applications between... an MMS User Agent and an MMS VAS Application (*or vice versa*)” (EX-1004, 55)—for example, an “abstract message[]... sent *by an MMS User Agent... on behalf of an originating application*” on the user device (*id.*) that is “passed by the MMS Relay/Server *to a VASP*” (EX-1004, 116-117, FIG. 9). An example of this would be, e.g., an application on the device for chess using MMS to transport data regarding a user’s inputted chess move to a chess VAS Application.

170. TS-23.140 discloses that “an application may trigger an MMS User Agent... to submit” “abstract messages.” EX-1004, 55. “Upon” this “triggering” by an application to send an abstract message—i.e., *in response to* the application’s *request*—the MMS User Agent “insert[s]” information “receive[d]... from the application” into an “abstract message” and sends it. EX-1004, 55. As explained in paragraphs 171-174 below, TS-23.140 illustrates how such information is transmitted to the VASP (via the MMS Relay/Server) using its MM1_submit.REQ and MM7_deliver.REQ “abstract message” examples. See EX-1004, 62, 116, 58, 189-190. Specifically, as explained in paragraphs 171-174 below, TS-23.140 discloses how the MM1_submit.REQ message information from the MMS User Agent is sent by the MMS Relay/Server via an MM7_deliver.REQ message to the VAS Application.

171. TS-23.140 describes the MM1_submit.REQ “abstract message” in Table 4, copied below. EX-1004, 62 (Table 4).

Table 4: Abstract messages for submission of MM in MMS

Abstract messages	Type	Direction
MM1_submit.REQ	Request	MMS UA -> MMS Relay/Server
MM1_submit.RES	Response	MMS Relay/Server -> MMS UA

172. “The MMS Relay/Server will deliver messages to the VASP by supplying the MM as the payload of the MM7_deliver.REQ. The message originates, for example, from a MMS User Agent, an external application, or from

outside the MMSE. This delivery may include an identification of the request that may be used by the VASP to correlate a response to the message.” EX-1004, 116-117. TS-23.140 describes the MM7_deliver.REQ “abstract message” in Table 61, copied below. EX-1004, 116 (Table 61).

Table 61: Abstract messages for demanding a service from a VASP

Abstract messages	Type	Direction
MM7_deliver.REQ	Request	MMS Relay/Server -> VASP
MM7_deliver.RES	Response	VASP -> MMS Relay/Server

173. “The MMS Relay/Server shall translate recipient addresses that originate from the MM1 interface into the appropriate URL of the VASP, for example *when an MM7_deliver.REQ results from an MM1_submit.REQ from the MMS User Agent.*” EX-1004, 58 (Section 7.2.3); *see also* EX-1004, 189-190 (Annex K; mapping information elements in the MM1_Submit.REQ abstract message to information elements in the MM7_Deliver.REQ abstract message in Table K.1, including Applic-ID and Content; “There is a table for each MM1... abstract message that maps to a MM7 abstract message.”).

174. TS-23.140 discloses that, after the MMS Relay/Server receives an MM1_submit.REQ message from the MMS User Agent, it sends an MM7_deliver.REQ message to the MMS VAS Applications server. EX-1004, 116-117 (“The MMS Relay/Server will deliver messages” that “originate[]... from a

MMS User Agent” “to the VASP by supplying the MM as the payload of the MM7_deliver.REQ[.]”), FIG. 9 (annotated below):

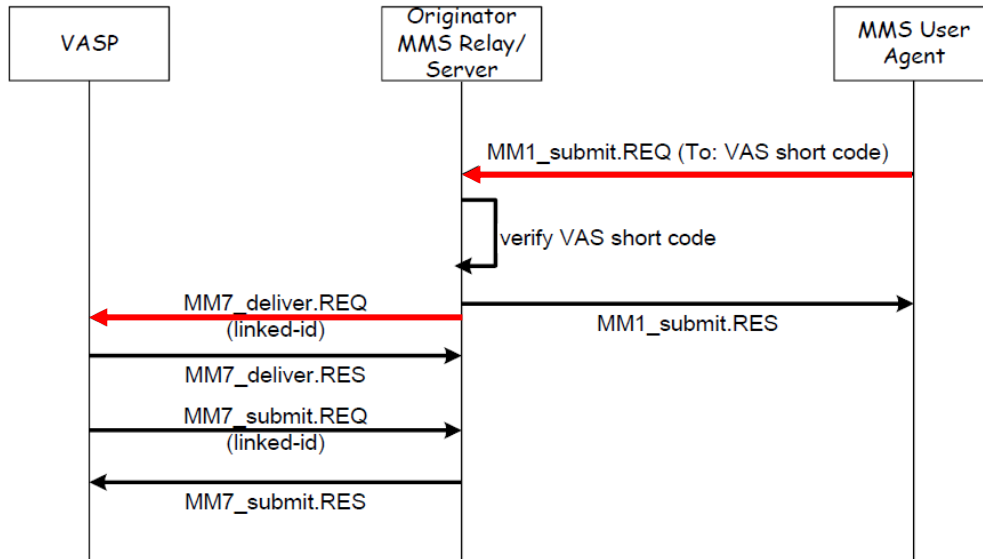


Figure 9: Use of MM7_deliver and subsequent response

EX-1004, 117, FIG. 9 (annotated)

175. As discussed above in Section VII.D.1.e, communication between MMS-Ogawa’s MMS User Agent application and other applications on the user device occur over MMS-Ogawa’s bus (“secure interprocess communication service”). A POSA thus understood that MMS-Ogawa’s MMS User Agent (the “device messaging agent,” *see* my discussion above in Section VII.D.1.c.i), “send[s] secure upload Internet data messages to the” MMS Relay/Server (the “network message server,” *see* my discussion above in Section VII.D.1.c.iii) “over the secure connection” in “respons[e] to” receiving a “corresponding request

received on the secure interprocess communication channel” from “*a corresponding one of the software applications*” on the user device (i.e., receiving a request on the bus from one of the applications downloaded on the device, per my discussion above in Section VII.D.1.d.i). A POSA understood that MMS-Ogawa’s abstract messages (e.g., MM1_submit.REQ) are “secure upload Internet data messages” because they are uploaded by the user device to the MMS Relay/Server for delivering information to the MMS VAS Application (the “*application server*” discussed in Section VII.D.1.d.ii).

176. Claim 11 further requires that “the device messaging agent” “construct[] from [the] request” “a secure upload Internet data message containing an identifier for a respective network application server corresponding to the requesting software application; and content received with the request.”

177. MMS-Ogawa’s User Agent does this. TS-23.140 describes that, “[u]pon triggering an MMS User Agent... to send an abstract message[,] the MMS User Agent... receive[s] information from the application” that the MMS User Agent “insert[s]... in both the information elements and/or payload... of the abstract message[.]” EX-1004, 55. TS-23.140 also says that the message “shall contain a destination application identifier” for use by the MMS VAS Application to “immediately route the received MMS information on to the destination

application that is referred to by the destination application identifier.” EX-1004, 55-56.

178. TS-23.140 also describes “MM recipients of a submitted MM shall be indicated in the addressing-relevant information field(s) of the MM1_submit.REQ[,]” and says “[t]he presence of this information element indicates that this abstract message shall be provided to an application residing on an... MMS VAS Application. It contains the identification of the destination application.” EX-1004, 63 (describing use of “Addressing” and “Applic-ID” in MM1 communications), 65 (showing “Information Elements” in MM1_submit.REQ); *see also id.*, 117-118 (showing that both MM1_submit.REQ and MM7_deliver.REQ messages include the identifier of the destination application: “[a]ll relevant address information for the delivery of the message to the VASP... should be included in the relevant information elements of MM7_deliver.REQ.”; and “This information [Applic-ID] element contains the identification of the destination application. Upon reception, the recipient MMS VAS Application shall provide this MM7_Deliver.REQ to the specified destination application.”).

179. A POSA therefore understood that MMS-Ogawa’s MMS User Agent “constructs from [the] request” a message containing the “identifier *for*” the “respective network application server corresponding to the requesting software

application”—because the identifier is *for* use by the MMS VAS Application to route the message to the correct destination application. EX-1004, 55-56.

180. TS-23.140 also discloses the MMS User Agent sending “content” from the originating application for delivery by the MMS Relay/Server to the VASP. EX-1004, 63, 118 (showing that MM1_submit.REQ and MM7_deliver.REQ messages include content for delivery to the VASP). A POSA therefore understood that the message the MMS User Agent “constructs from [the] request” *also* contains the “content,” as claimed.

7. **Claim 12: The mobile end-user device of claim 11, wherein at least one of the upload Internet data messages comprises a key for the network application server corresponding to the requesting software application.**

181. Claim 12 requires that “at least one of the upload Internet data messages of claim 11 ‘comprises a key for the network application server corresponding to the requesting software application.’” Neither the claims nor the ’403 specification require any specific *type* of “key,” and the specification explains that “various known security encryption techniques can be implemented in the encrypt functions (2408, 2428), with *public/private or completely private keys and/or signatures.*” EX-1001, 87:55-58. During prosecution, PO likewise did not narrow the key type that claim 12 is meant to cover—generally citing an alleged discussion of “secure device agent-to-corresponding network element

communication examples, including various types of keys and signing.” EX-1002, 735.

182. In MMS-Ogawa, messages are encrypted using symmetric MMS-Ogawa Message Encryption before they are transmitted to the MMS User Agent or the MMS VAS Applications. *See* my discussion above in Sections VII.C.3-VII.C.4. A POSA understood that when MMS-Ogawa is used as described for claim 11—where application-specific data is sent *by an MMS User Agent... on behalf of an originating application*” on the user device “*to a VASP*”—the recipient MMS VAS Application (which includes the destination application) needs to receive the shared encryption key that is used to encrypt the message to enable decryption of the message received from the MMS User Agent. *E.g.*, EX-1005, 5:62-65 (“...received shared key encrypted data will be decrypted using the shared encryption key...”).

183. Ogawa expressly teaches a method for distribution of the shared encryption key between network elements that require it for encryption/decryption. EX-1005, 6:64-7:24, FIG. 2. In particular, Ogawa teaches using a “key exchange algorithm” involving *both* the encrypting entity (the MMS User Agent on the MMS-Ogawa device, which sends the upload Internet data message) and the decrypting entity (the MMS VAS application that receives the upload Internet data message) generating keys, which are then “exchange[d] in accordance with Diffie-

Hellman protocol,” and used to generate the shared encryption key. EX-1005, 6:64-7:24 (describing steps S9 and S10), FIG. 2 (showing steps S9 and S10). A POSA understood that this exchange of keys between the MMS User Agent (“device messaging agent”) and the MMS VAS Application (“network application server”)—so that both entities can generate the shared symmetric key needed for MMS-Ogawa Message Encryption—would meet claim 12’s requirement that “at least one of” claim 11’s “upload Internet data messages comprises a key for the network application server corresponding to the requesting software application,” as claimed.

184. A POSA would have understood that such an “upload Internet data message” would still be “secure,” as required by claim 11—even though the communications are not yet secured using MMS-Ogawa Message Encryption—because MM1 and MM7 are secured using TLS/SSL. *See* my discussion above in Section VII.C.2; EX-1004, 41 (“MM7 should use transport layer security mechanisms to authenticate the VASP...”), *id.* (disclosing “HTTP over SSL or TLS” for MM7).

8. Claim 13: The mobile end-user device of claim 1, wherein the device messaging agent creates a log for the received secure Internet data messages.

185. Claim 13 requires that “the device messaging agent create[] a log for the received secure Internet data messages.”

186. During prosecution, PO indicated that the claimed “log” encompasses a “communication trace log for, e.g., service controller 122 to agent communications.” EX-1002, 735. According to the ’403 specification, “service controller 122 to agent communications... are monitored and logged so that a trace log of some... agent communications can be maintained[,]” (EX-1001, 44:42-47) and that “the agents can maintain their own communications or attempted communications log, which can then be reported to the service controller 122.” EX-1001, 45:12-15; EX-1002, 735. PO’s prosecution statements indicate that the claimed “log” encompasses a record of at least some of the communications received by the “device messaging agent” that is sent to a “network application server.” Creating such communication logs was a standard function in computing systems and it had been commonplace for applications to create them for decades. *See generally* EX-1057.

187. TS-23.140 discloses a VASP requesting a “read-reply report” for a MM received by the MMS User Agent. EX-1004, 34-35 (Section 7.1.6, “Read-Reply Report,” supported by MMS), 59 (“Read-reply reports are sent by the recipient MMS User Agent.”). The read-reply report “shall [contain/provide]” the MM sender’s and recipient’s addresses, the message ID, the read status of the MM, and a time stamp. EX-1004, 34-35. The MMS User Agent stores each read-reply report until it can reach the MMS Relay/Server, at which time it sends all stored

read-reply reports to the MMS Relay/Server, which are then sent to the VASP. EX-1004, 35. A POSA thus understood that MMS-Ogawa's MMS User Agent (the claimed "device messaging agent") creates the claimed "log for the received secure Internet data messages."

9. Claim 14: The mobile end-user device of claim 1, wherein the secure interprocess communication channel and the secure connection to the network message server are separately secured.

188. Claim 14 requires the "secure interprocess communication channel" and the "secure connection to the network message server" be "separately secured."

189. MMS-Ogawa's MMS User Agent receives messages from the MMS Relay/Server that are secured using both SSL/TLS and MMS-Ogawa Message Encryption, which are then decrypted by MMS-Ogawa's MMS User Agent. *See* my discussion above in Sections VII.C.2-VII.C.4 and VII.D.1.c.iii. Such messages are then re-encrypted by MMS-Ogawa's MMS User Agent using an inherent key before being communicated over MMS-Ogawa's "secure interprocess communication service." *See* my discussion above in Sections VII.C.5-VII.C.6 and VII.D.1.e. Thus, MMS-Ogawa's external secure connection between MMS User Agent and MMS Relay/Server is "*separately secured*" from MMS-Ogawa's internal secure interprocess communication channel, meeting claim 14.

10. Claim 15: The mobile end-user device of claim 1, wherein access by the software applications to the interprocess communication channel is subject to a security policy.

190. Claim 15 requires that “access by the software applications to the interprocess communication channel is subject to a security policy.” The ’403 specification uses “security policy” only once (EX-1001, 36:52), and never to describe access to an interprocess communication channel. The ’403 specification says the agent bus can be “secured” using “point[-]to[-]point” or “bus-level” “message exchange encryption using one or more keys that are partially shared or shared” (EX-1001, 41:62-42:4) and says that, by encrypting the agent bus, it “can only be accessed... as... permitted by *agent communication policies*” (EX-1001, 88:9-22). *See* my discussion above in Section VII.D.1.e. Based on the intrinsic record, a POSA understood that using shared key encryption is an example of subjecting access to the interprocess communication channel to a “security policy,” as claimed.

191. MMS-Ogawa’s interprocess communication service is secured using shared key encryption. *See* my discussion above in Sections VII.C.6 and VII.D.1.e. Thus, access by MMS-Ogawa’s applications are “subject to a security policy”—one that requires a specific encryption key—and meets claim 15.

11. Claim 16: The mobile end-user device of claim 1, wherein at least one of the secure Internet data messages comprises multiple identifier/data pairs.

192. Claim 16 requires that “at least one of the secure Internet data messages comprises multiple identifier/data pairs.” The ’403 specification does not describe a “secure Internet data message” that “comprises multiple identifier/*data* pairs.” During prosecution, PO identified “Figure 25 [as] illustrat[ing] such a... message structure, wherein... the data message... *is composed of* [multiple] agent *messages*, each formatted as an agent identifier with a paired message payload[.]” EX-1002, 736. FIG. 25 (annotated below) shows a “service controller communication *frame*” comprising multiple identifier/*message* pairs (respectively blue, green). EX-1001, 90:11-18 (describing FIG. 25).

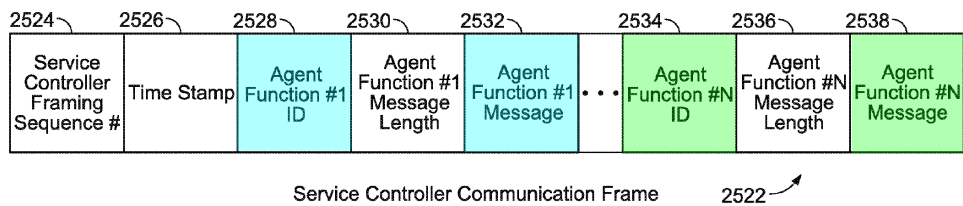


FIG. 25

EX-1001, FIG. 25 (annotated)

Additionally, the specification states that “service control server link 1638 can perform collection or buffering of server messages between transmissions[, and] once a transmission trigger has occurred, ...take all buffered agent communications and frame the communications.” EX-1001, 68:62-69:8. PO’s

prosecution statements indicate that claim 16's "message[] compris[ing] multiple identifier/data pairs" encompasses a network message server that collects and buffers messages (each comprising an agent identifier and payload), and then sends the collected messages to a "device messaging agent" as a group or frame of messages.

193. TS-23.140 describes, upon receiving a message, the MMS Relay/Server "stor[ing] the MM at least until: the associated time of expiry is reached, the MM is delivered, the recipient MMS User Agent requests the MM to be routed forward, [or] the MM is rejected." EX-1004, 28; *see generally id.*, 26-28 (Sections 7.1.1-7.1.2, Submission and Reception of Multimedia Messages). Messages may be "persistent[ly] stored" (EX-1004, 21) in a "Persistent Network-Based Storage" (MMBox) associated with "the MMS Relay/Server." EX-1004, 22 (Section 5.2.1, Persistent Network-based Storage (MMBoxes)).

194. When an unavailable recipient MMS User Agent becomes available, *e.g.*, by moving into coverage, the MMS Relay/Server delivers the group of collected messages to the MMS User Agent. EX-1004, 29 (§7.1.3). As explained above in Sections VII.D.1.d.i-VII.D.1.d.ii, each message in the group comprises a destination application identifier and application data. The group of collected messages delivered to the MMS User Agent is thus a "secure Internet data message[] compris[ing] multiple identifier/data pairs," as claimed.

195. To the extent claim 16 is read to require aggregating multiple messages in a frame, this would have been obvious. A POSA had reason to implement MMS-Ogawa such that the MMS Relay/Server transmits multiple collected MMs to a MMS User Agent in a frame comprising multiple MMs. TS-23.140 is part of a broader 3GPP suite of specifications—well-documented and well-known to POSAs, as discussed below—that define layered architecture and data transport mechanisms for MMS communications over 3G packet-switched bearers.

196. TS-23.140 contemplates each MM being conveyed in an HTTP message over TCP/IP, which is delivered to the radio access network through the 3G packet-switched bearer service. EX-1021 (MMS Encapsulation Protocol), 11, 13, 48. (EX-1021 is an OMA MMS protocol which was incorporated into EX-1004 at 12, 174-175.) Moreover, a POSA would have understood that multiple higher-layer data units—e.g., multiple HTTP messages, each carrying a distinct MM—would be aggregated within a single lower-layer frame for transmission to an MMS User Agent. Under the Release 6 bearer architecture, those same HTTP/TCP data units are carried by the Packet Data Convergence Protocol (PDCP, 3GPP TS 25.323), the Radio Link Control (RLC, 3GPP TS 25.322), and the Medium Access Control (MAC, 3GPP TS 25.321). The RLC layer in particular is expressly permitted to segment and/or concatenate multiple service data units to form a

single protocol data unit for transmission, while the MAC layer multiplexes multiple RLC PDUs within a single transport block. *See, e.g.*, TS-25.323 (PDCP) (EX-1053), 8-9, 23 (explaining PDCP performs transfer of upper-layer PDUs, e.g., MMS/HTTP/TCP/IP, containing user data, and “uses the services provided by the [RLC] sublayer”); TS-25.322 (RLC) (EX-1052), 11, 16, 22-23 (explaining RLC concatenates RLC SDUs (e.g., PLDC PDUs) into PDUs for the MAC layer); and TS-25.321 (MAC) (EX-1051), 12, 15, 28-29, 46-47 (explaining RLC PDUs are provided to the MAC layer, and concatenated into MAC PDUs before transmission). This was desirable because encapsulating multiple identifier/data pairs, or multiple MMs, within a single transmission frame would improve radio-interface efficiency and reduce protocol overhead.

197. Moreover, a POSA would have reasonably expected success implementing MMS-Ogawa’s MMS Relay/Server to transmit a group of messages using such a frame, especially considering the relevant 3GPP specifications (see my discussion above in ¶ 196) that expressly support encapsulation of multiple higher-layer data units into a single transport block for precisely this reason.

198. Thus, implementing MMS-Ogawa such that the MMS User Agent receives a message (e.g., a frame) containing “multiple ID/data pairs” was obvious.

12. Claim 17: The mobile end-user device of claim 1, the device messaging agent comprising an agent router to forward the application data on the secure interprocess communication

channel to the software process corresponding to the identified software application.

199. Claim 17 requires that the claimed “device messaging agent compris[e] an agent router” that “forward[s] the application data” “to the [claimed] software process” over “the secure interprocess communication channel.” The ’403 specification does not describe an “agent router.” During prosecution, PO asserted the term is met by FIG. 24’s “‘agent route’ function 2416.” EX-1002, 736. The specification does not describe any structural requirements for “agent route 2416,” and says only that it is “for routing... received... communications... to the appropriate agent...” EX-1001, FIG. 24, 89:15-21. Route 2416 is shown as part of service control device link 1691. *Id.*; *see also*, 42:10-11 (link 1691 is “equivalent to an agent”). Therefore, a portion of the device messaging agent (either software or hardware) that routes received messages to an appropriate application meets the claimed “agent router.”

200. As I discussed above in Sections VII.D.1.c-VII.D.1.d, TS-23.140 discloses that, upon receiving an application-specific message, the “*MMS User Agent... immediately route[s] the received MMS information on to the destination application that is referred to by the destination application identifier[.]*” EX-1004, 54-56.

201. As I explained above in Sections VII.D.1.e-VII.D.1.f, the MMS User Agent forwards the message’s application-specific data content (“application

data”) on the secure interprocess communication service to a software process corresponding to the destination application. MMS-Ogawa’s User Agent (“device messaging agent”) thus comprises an “agent router,” as claimed. EX-1004, 54-56.

13. Claim 18: The mobile end-user device of claim 1, wherein the secure interprocess communication service forwards the application data to at least one of the software processes in an encrypted format.

202. Claim 18 requires that “the secure interprocess communication service forwards the application data to at least one of the software processes in an encrypted format.” When MMS-Ogawa’s MMS User Agent receives a message comprising application-specific data over MM1, the message is decrypted by the MMS User Agent’s decryption unit, then re-encrypted by the MMS User Agent’s encryption unit and routed “to the destination application that is referred to by the destination application identifier” (EX-1004, 56) over MMS-Ogawa’s “secure interprocess communication service” (MMS-Ogawa’s bus) *See my discussion above in Sections VII.C.3-VII.C.6, VII.D.1.c.iii, VII.D.1.e, and VII.D.9.* Thus, the message is forwarded by the bus to a process of the destination application (“*at least one of the software processes*”) in an encrypted format, meeting claim 18.

14. Claim 19: The mobile end-user device of claim 1, the device messaging agent further to initiate the secure connection to the network message server.

203. Claim 19 requires that the “device messaging agent” “initiate the secure connection to the network message server.” As discussed for Claim 11, TS-

23.140 teaches the MMS User Agent sending an MM1_submit.REQ message to the MMS Relay/Server when another application on the user device wants to send a message to an MMS VAS Application. A POSA understood that in MMS-Ogawa, if the SSL/TLS-secured connection between MMS User Agent and MMS Relay/Server (MM1) is not already established—e.g., because the prior connection timed out, or because the user device was restarted—MMS-Ogawa’s MMS User Agent would initiate the secure connection with the MMS Relay/Server, for example by sending a TLS “Client Hello” message to the MMS Relay/Server to establish the connection. EX-1014, §7.4.1.2; *see also, e.g.*, EX-1026, [0275]-[0293] (describing TLS handshake phases and “ClientHello”), FIGS. 17-18.

15. Claim 20: The mobile end-user device of claim 1, further comprising a network stack in communication with the device messaging agent and the WWAN modem, the secure connection terminated within the network stack.

204. Claim 20 requires that claim 1’s “device” “compris[e] a network stack in communication with the device messaging agent and the WWAN modem[,]” and that the claimed “secure connection” be “terminated within the network stack.” The ’403 specification does not define requirements for a “network stack.” The specification uses “networking stack” and “device communications stack” to include a layered set of software components that implement network communication protocols. EX-1001, 92:62-94:9. During prosecution, PO stated that based on the ’403 specification’s alleged disclosure of a “network stack” that

uses “basic IP” and “TCP layer security,” POSAs understood the secure connection “can include security terminated in the network stack.” EX-1002, 736.

205. As I explained above in Sections VII.C.2 and VII.D.1.c.iii, MMS-Ogawa’s MM1 is secured using SSL/TLS. EX-1004, 24-25, FIG. 4 (below).

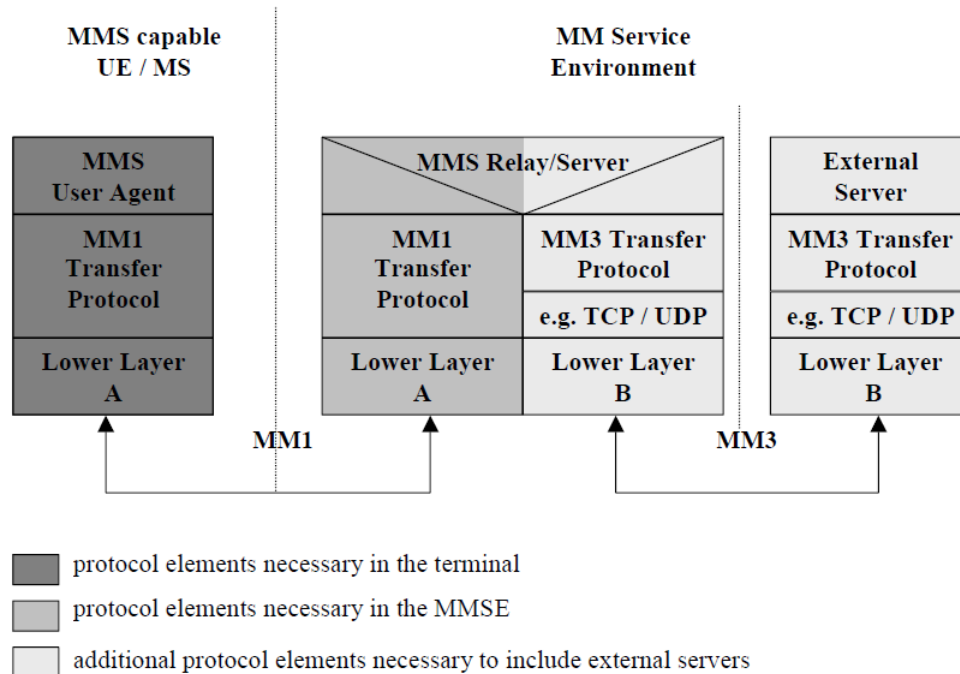


Figure 4: Protocol Framework to provide MMS

EX-1004, 24, FIG. 4

As I discussed previously in paragraph 130 above, the specification uses “TLS” and Secure Socket Layer (SSL) interchangeably. See, e.g., EX-1001, 94:3, 98:30; 99:15-16; 101:52-54 (providing “basic TCP setup, TLS/SSL” as example protocols used in “socket assignment and session management” layer in FIGS. 31-37).

206. As noted above in Section VII.C.2, TS.23-140 discloses MM1 implementing OMA MMS. “[A] device implementing OMA MMS must have...[a]

WAP WSP stack *or HTTP/TCP/IP stack.*” EX-1011, 11; *see also* EX-1004, 13 (incorporating-by-reference EX-1011), 162 (referencing EX-1011 for OMA implementation for MM1); EX-1011, 4-5 (incorporating-by-reference EX-1012, an “Architecture Overview”), 10 (same); EX-1012, 21 (“The TLS [WP-TLS] transport layer security protocol provides for secure data transmission between the MMS Client and the MMS Proxy-Relay in architectural configurations that employ HTTP based protocol *stacks* for MMSM implementation.”). A POSA thus understood MM1 comprises a “network stack,” as claimed.

207. Further, because MM1 utilizes the TCP/IP stack to provide the “lower layer” secure connection between MMS User Agent and MMS Relay-Server and is also secured via the TLS/SSL security layer of the TCP/IP stack, a POSA understood that the claimed secure connection (MM1, *see* Section VII.D.1.c.iii above) terminates within the TCP/IP stack in the same way PO says the claim can include, by using “basic IP” and “TCP layer security.” *See* EX-1004, 24 (showing and describing FIG. 4, which illustrates the protocol framework to provide MMS over MM1); EX-1012, 21 (discussing TLS “security protocol” that provides “secure data transmission between the MMS Client and the MMS Proxy-Relay in... HTTP based protocol stacks....”); EX-1002, 736 (discussing “basic IP” and “TCP layer security”). *See my discussion above in paragraphs 204-206.*

208. As I discussed above in Sections VII.D.1.b and VII.C.1, MMS-Ogawa's device comprises a WWAN modem to enable communications, via MMS User Agent, over TS-23.140's 3G mobile networks (the claimed "WWAN"). A POSA understood that MMS-Ogawa's MMS User Agent ("device messaging agent") communicated with MMS Relay/Server via WWAN modem over the MM1 interface, using the TCP/IP stack. EX-1004, 17 (showing a User Agent communicating with Relay/Server over 3G) 24-25 (Section 6.2, Protocol Framework to provide MMS), FIG. 4. A POSA also understood that MMS User Agent sits at the application layer, while MMS-Ogawa's WWAN modem sits below the TCP/IP stack. *See* EX-1004, 19 (describing "application layer functionalities" of MMS User Agent), 24 (showing a protocol framework to provide MMS in FIG. 4). A POSA thus understood that the TCP/IP stack used by MM1 is in communication with the user device's MMS User Agent and WWAN modem, as claimed.

16. **Claim 21: The mobile end-user device of claim 1, wherein at least one of the applications and the network application server corresponding to that application authenticate with each other prior to passing application data via the device messaging agent and network message server.**

209. Claim 21 requires that "at least one of the applications and the network application server corresponding to that application authenticate with each

other prior to passing application data via the device messaging agent and network message server.”

210. In MMS-Ogawa, applications on an MMS User Agent device or an MMS VAS Applications server “initially need to register with the appropriate MMS User Agent or MMS VAS Application” before exchanging messages using MMS. EX-1004, 54-55. “During this registration process the application provisions an MMS User Agent or an MMS VAS Application with its application identification value and may negotiate with the MMS User Agent or MMS VAS Application the details... of information to be exchanged between the two entities[,]” and “[b]ased on the negotiated details[,]... an application may trigger an MMS User Agent or an MMS VAS Application to submit certain abstract messages.” EX-1004, 55. A POSA understood that this required registration process with identifiers constitutes an “application[] and... network application server” “authentica[ing] with each other prior to passing application data” to one another via the MMS Relay/Server, as claimed, because without going through this registration process, provisioning the necessary identifiers, and negotiating the details of information to be exchanged – i.e., taking steps to establish that they are respectively permitted to use MMS to exchange data – the MMS Relay/Server will not relay messages. TS-23.140 also discloses the MMS-Relay Server “authentica[ing] the VASP during each session established for message

submission... before any transactions will be allowed by the MMS Relay/Server,” and “authoris[ing] the VAS to send MM to the MMS UA... during each session established by the VAS.” EX-1004, 41.

VIII. GROUND 2A-2C

211. I note that in IPR2024-00341, the Board agreed that it would have been obvious to implement TS-23.140 with a modem for network communications (discussed above in Section VII.C.1) and a software bus (e.g., a D-bus) for interprocess communications (discussed above in Section VII.C.5). EX-1022, 40-41, 45, 48-49.

212. As I discuss below in Grounds 2A-2C, POSAs would have also alternatively been motivated to modify the combination of TS-23.140 and Ogawa that I describe above in Ground 1 in view of (1) express teachings in Cole (of a modem, which I discuss below as Ground 2A), (2) express teachings in Sathish (of an interprocess communication bus, which I discuss below as Ground 2B), and (3) *both* Cole and Sathish (which I discuss below as Ground 2C).

213. As discussed below, the incorporation of Cole’s teachings (in Grounds 2A and 2C) also address claim 2, which recites a WLAN modem.

A. GROUND 2A: MMS-Ogawa in View of Cole (EX-1006)

1. Implementing a WWAN Modem in View of Cole (Claims 1, 3-6, 11-21)

214. As I discussed above in Section VII.C.1 (e.g., paragraphs 60 and 62), it was well-documented before the Critical Date to use a *modem* to enable communications over the 2G/3G WWAN networks described in TS-23.140. For example, for connecting a “mobile device” to a “2G”/“3G” “wireless wide area network (WWAN)” —which “may interface directly or indirectly with a public network 170 (e.g., the Internet)” —Cole expressly discloses implementing the mobile device’s various “communication interfaces” to include “*a WWAN modem.*” EX-1006, [0031] (describing a “mobile device” that interfaces with a “plurality of communication networks” including 2G and 3G wireless cellular technologies), [0034]-[0035] (describing interfacing with Cole’s disclosed networks using a communication interface that includes “a WWAN modem”), FIGS. 1-2 (illustrating same, below).

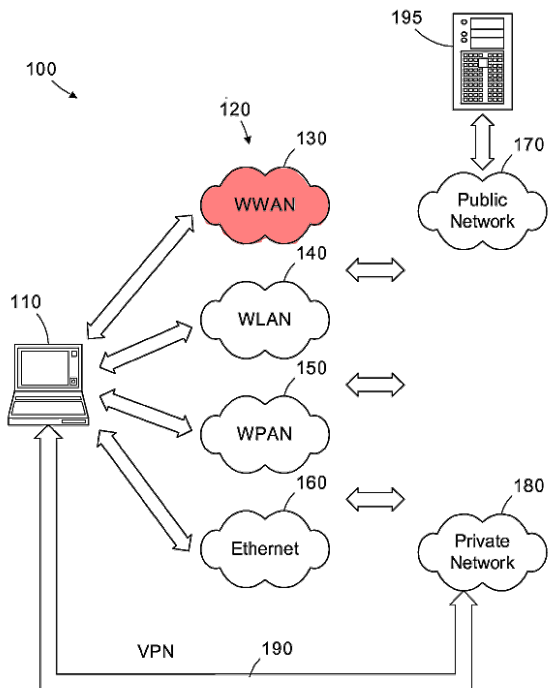


Figure 1

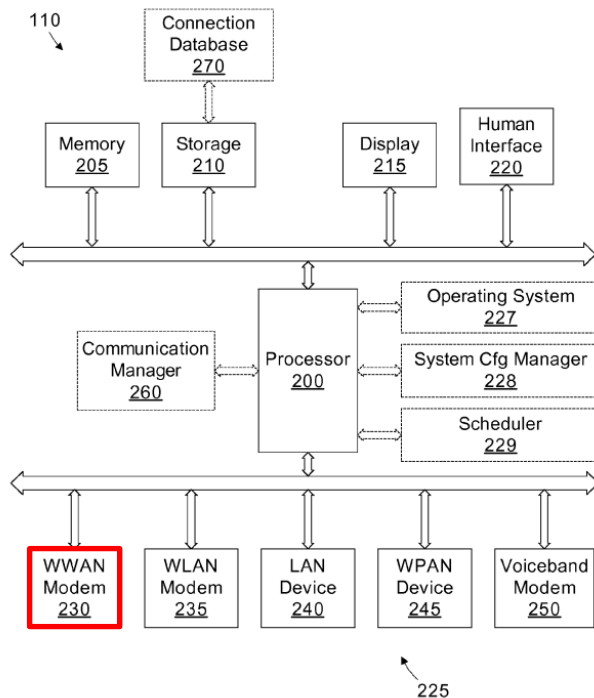


Figure 2

EX-1006, FIGS. 1-2 (annotated)

Cole confirms that “[t]he various protocols required to implement connections over [such] communication interfaces... to [such]... communication networks... ***are known to those [of] ordinary skill.***” EX-1006, [0035].

215. In view of Cole (e.g., the teachings I discuss above in paragraph 214 from EX-1006, [0003], [0031]-[0035], FIGS. 1-2), a POSA would have been motivated to use a “WWAN modem” to enable TS-23.140’s device to communicate over TS-23.140’s disclosed 2G/3G wireless networks, and would have reasonably expected success doing so. Such an implementation would have been nothing more than utilizing familiar, known components (Cole’s WWAN

modem in TS-23.140's device that communicates over 2G/3G networks) to achieve a predictable result of facilitating TS-23.140's disclosed communications.

216. This combination renders obvious claims 1, 3-6, and 11-21 for the same reasons I discussed above in Section VII for each of those respective claims in the context of MMS-Ogawa (Ground 1).

2. Adding a WLAN Modem in View of Cole (Claim 2)

217. TS-23.140 says the “[MMS] architecture... shall combine different networks and network types and shall integrate messaging systems already existent within these networks[,]” and that “[t]he basis of connectivity between these different networks shall be provided by the Internet protocol and its associated set of messaging protocols.” EX-1004, 16-17. TS-23.140 does not limit the ways in which its user device facilitates communications over such networks or what “network types” the MMS “architecture... shall combine.” *E.g.*, EX-1004, 16-17, 23-24.

218. For example, TS-23.140's FIG. 2 (below) shows, e.g., “Roaming MMS User Agent” that connects to MMS Relay/Server through generic “Mobile Network B.”

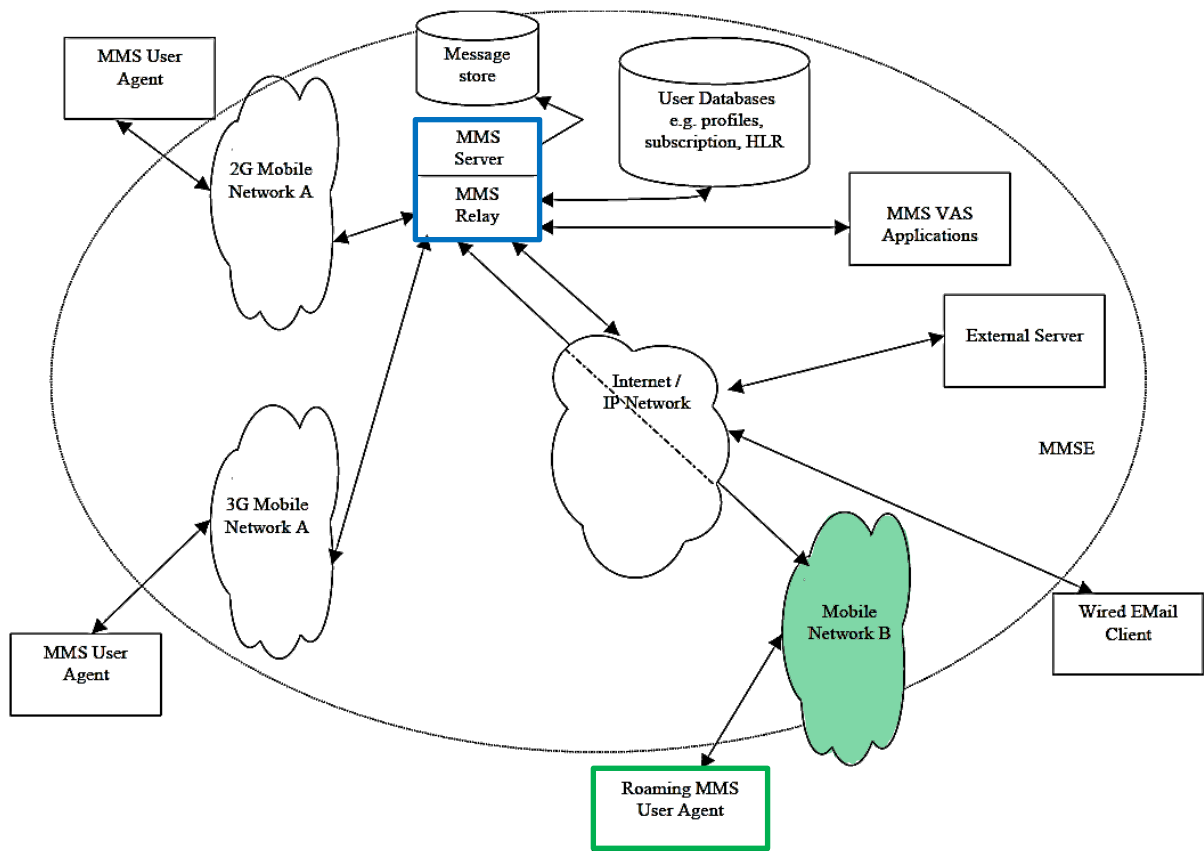


Figure 2: MMS Architectural Elements
EX-1004, 17, FIG. 2 (annotated)

219. Contemporaneous references, such as Trossen (EX-1007)—a 2004 publication that describes an “MMS system” and cites to the 3GPP Technical Specification (TS-23.140) (EX-1007, [0002]-[0003])—confirm that one way to implement TS-23.140’s disclosed “mobile network” was as a “WLAN,” and that it was well-known to implement MMS User Agents to be “capable of transmitting and receiving multimedia messages” to/from a messaging server (e.g., TS-23.140’s MMS Relay/Server) over “a mobile access network 126 such as a wireless local area network (WLAN)” while roaming. EX-1007, [0022], FIG. 1 (below).

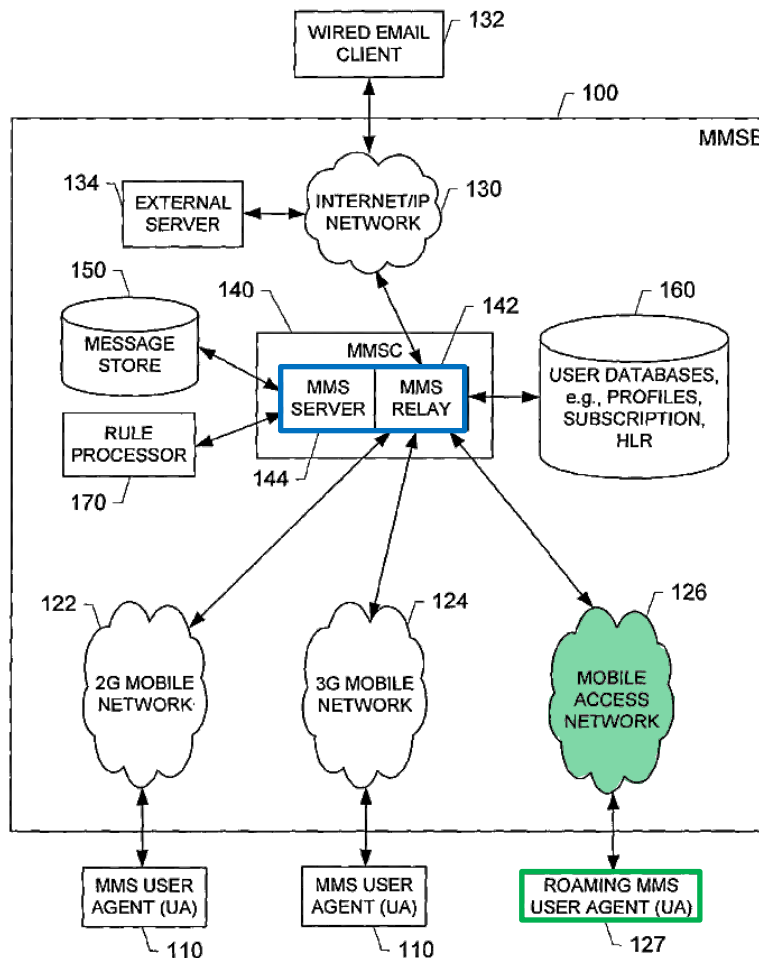


FIG. 1.

EX-1007, FIG. 1 (annotated)

Another 3GPP standard, TS-23.234, which describes “3GPP system *to Wireless Local Area Network (WLAN)* interworking” and is discussed further below, confirms same. *E.g.*, EX-1016, 8, 14, 26-28.

220. As demonstrated by Cole itself, it was well-documented before the Critical Date to implement such WLAN connectivity on mobile devices using a “WLAN modem.” EX-1006, [0003], [0031], [0035], FIG. 2. *See also* Ko (EX-

1042), [0050] (“Referring to FIG. 2, the mobile station (MS) or multimode mobile terminal 400 includes a main control part 410, a WLAN modem 420, a mobile network modem 430, and a user interface 440.”), FIG. 2.

221. In particular, in addition to the “WWAN modem” discussed above in Section VIII.A.1, Cole teaches including a “WLAN modem” among a mobile device’s “communication interfaces.” EX-1006, [0031], [0034]-[0035], FIGS. 1-2 (below).

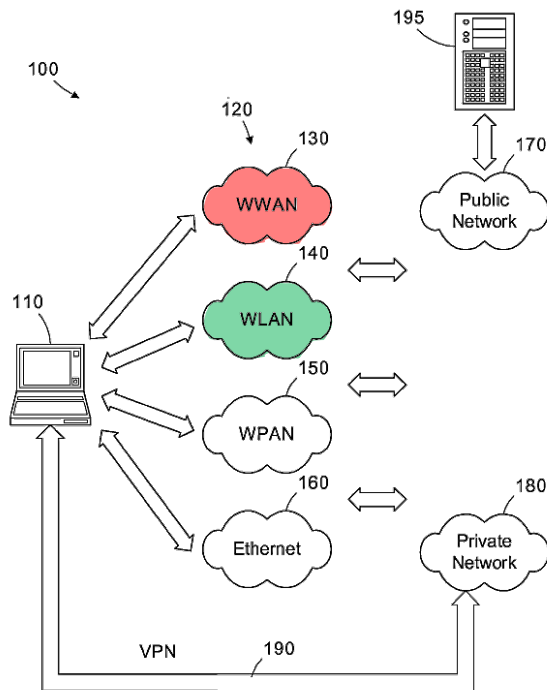


Figure 1

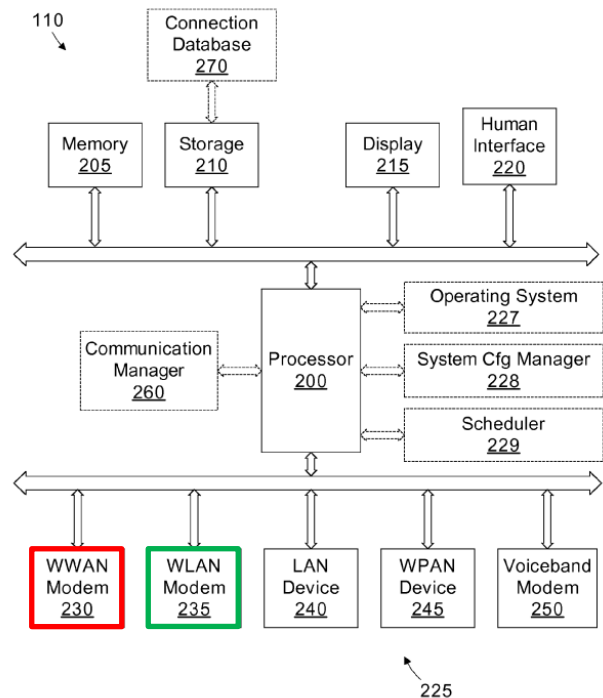


Figure 2

EX-1006, FIGS. 1-2 (annotated)

222. Cole teaches “determin[ing] which connections should be active...” through “evaluat[ion of] availability, bandwidth, user preferences, application requirements, mobile device resources, etc. to select a particular communication

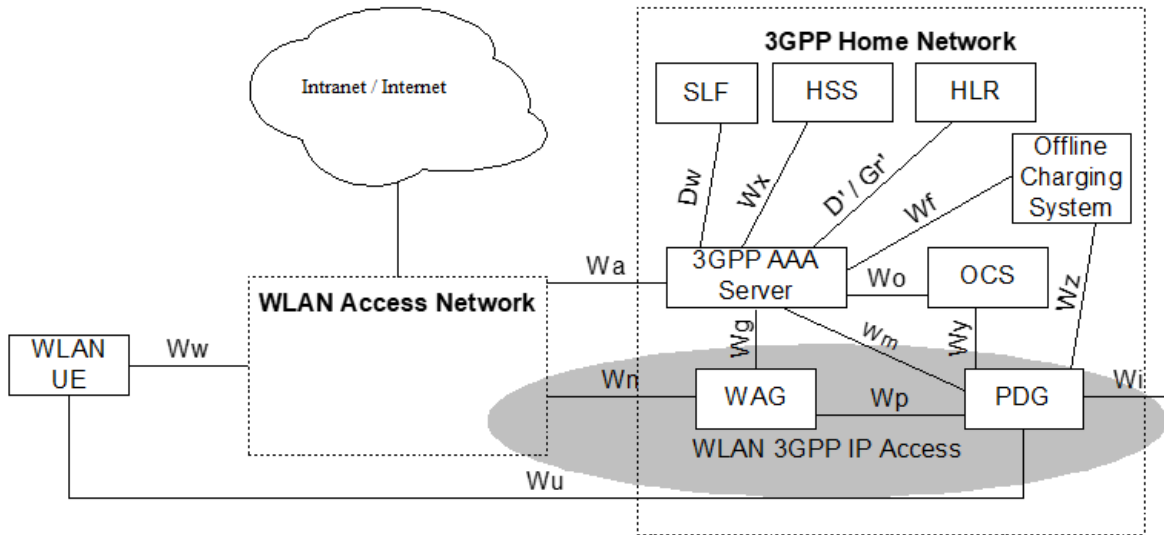
interface 225.” EX-1006, [0036]; *see also id.*, [0037]-[0040] (describing ways to choose which communication interface to use), [0003] (describing communication schemes for connecting to networks in multiple ways). As noted, Cole confirms that “[t]he various protocols required to implement [such] connections... **are known to those [of] ordinary skill.**” EX-1006, [0035].

223. A POSA had multiple reasons to implement Cole’s WLAN modem into in MMS-Ogawa’s user device:

224. **First**, TS-23.140 contemplates “multimedia messaging” that “encompass[es] **many different network types**” and explains that “[t]he basis of connectivity between these different networks shall be provided by the **Internet protocol** and its associated set of messaging protocols.” EX-1004, 17. TS-23-140 also discloses “mak[ing] use of [a] protocol framework” “[t]o provide implementation flexibility [and] integration of existing and new services together with interoperability across different networks and terminals[.]” EX-1004, 24. A POSA would have appreciated in view of such teachings in TS-23.140 the desirability of using MMS over various “different networks types,” including WLANs.

225. **Second**, other MMS-related references expressly contemplate and teach how to implement WLAN-enabled devices to access 3GPP Packet-Switched-based services like MMS. *E.g.*, EX-1007, [0001] (“Multimedia Messaging

Service”), [0022] (“FIG. 1... illustrates an overview of MMS system elements... The access networks are generally of different types and can include... a mobile access network... such as a wireless local area network (WLAN).”), FIG. 1; Shen (EX-1043), Abstract (discussing “multimedia messaging service system comprising a wireless LAN”), [0001]; Shaheen (EX-1044), Abstract (describing “wireless communication system for supporting multimedia services (MMS)” that “includes a third generation partnership program (3GPP) universal mobile telecommunications system (UMTS), a wireless local area network (WLAN) and an MMS server.”). For example, TS-23.234, another 3GPP specification, discloses a system description for interworking between 3GPP systems and WLANs “to extend 3GPP services and functionality to the WLAN access environment” in order to “allow[] a 3GPP subscriber to use a WLAN to access 3GPP PS based services.” EX-1016, 8, 14 (listing MMS as a “3GPP PS based service[]”), 26-28 (depicting architectures (e.g., Figure 6.1) for providing a WLAN UE with 3GPP Packet-Switched-based service access), 28 (“The WLAN UE may be capable of... of both WLAN and 3GPP radio access.”). Figure 6.1 is reproduced below:



NOTE: The shaded area refers to WLAN 3GPP IP Access functionality.

Figure 6.1: Non-roaming reference model

EX-1016, FIG. 6.1

Above, the PDG accesses 3GPP PS based services (e.g., MMS architecture/framework) over interface Wi (at right)—that is, the PDG passes the MMS message onto the MMS R/S at this point.

226. *Third*, Cole teaches that a “wide variety of connection options provides the user with flexibility and the ability to connect to a network in virtually any location.” EX-1006, [0003]-[0004]. A POSA understood that incorporating Cole’s WLAN modem into MMS-Ogawa’s user device would have desirably provided an additional access path to the MMS Relay/Server using existing IP-based MMS protocols, thereby improving the device’s accessibility to MMS in environments where 2G/3G cellular coverage was weak, unavailable, or costly—resulting in increased user flexibility and connectivity for MMS services.

227. *Fourth*, implementing Cole’s WLAN modem into MMS-Ogawa’s user device would have been nothing more than implementing known components/techniques (Cole’s WLAN modem for communicating over a WLAN) into known systems/devices (TS-23.140’s user device, implemented to use SSL/TLS and MMS-Ogawa Message Encryption for MM1 communications with the MMS Relay/Server) to achieve predictable results (secure MMS communications over a non-2G/3G mobile access network, as contemplated by TS-23.140).

228. A POSA would have reasonably expected success implementing Cole’s WLAN modem into MMS-Ogawa given Cole’s confirmation that “protocols required to implement [such] connections... are known to” POSAs (EX-1006, [0035]), the above-cited teachings regarding implementing MMS over WLAN networks, and the widespread use of WLAN for mobile device communications before the Critical Date. Moreover, POSAs understood that nothing in the teachings of MMS or Ogawa require communication over a specific network type (e.g. WWAN).

229. In such an implementation (“MMS-Ogawa-Cole”), the MMS User Agent receives secure messages over a secure connection with MMS Relay/Server in the same manner I described above in Section VII.D.1 for connections over WWAN, because in MMS-Ogawa-Cole, the WLAN is part of interface MM1

which facilitates “Internet protocol” communications between the MMS User Agent and the MMS Relay/Server. Messages sent over MMS-Ogawa-Cole’s secure connection over WLAN, as with MMS-Ogawa’s connection over WWAN, are secured using both MMS-Ogawa Message Encryption and SSL/TLS. *See* my discussion above in Sections VII.C.2-VII.C.4 and VII.D.1.c.iii.

- a. **Claim 2: The mobile end-user device of claim 1, further comprising a wireless local area network (WLAN) modem to exchange Internet data via a connection to a first WLAN, when configured for and connected to the first WLAN, the device messaging agent further to receive secure Internet data messages over a secure connection via the first WLAN to the network message server.**

230. Claim 2 requires that claim 1’s device “further” comprises “a wireless local area network (WLAN) modem to exchange Internet data via a connection to a first WLAN, when configured for and connected to the first WLAN,” which allows “the device messaging agent... to receive secure Internet data messages over a secure connection via” a “first WLAN to the network message server.” MMS-Ogawa-Cole meets claim 2. A POSA understood Cole’s WLAN modem to be for “exchang[ing] Internet data via a connection to a first WLAN, when configured for and connected to the first WLAN,” as claimed, because during prosecution, PO stated that “the term ‘first’ merely indicates one of several potential WLAN connections” (EX-1002, 737), and a POSA understood that in MMS-Ogawa-Cole,

the WLAN modem would allow the device to connect to whatever WLAN connection the modem was “configured for and connected to.”

231. Moreover, as consistent with my discussion of MMS-Ogawa-Cole above in this section, MMS-Ogawa-Cole works in the same way that MMS-Ogawa works, except that MMS-Ogawa-Cole also includes a WLAN modem, over which the device can connect to a WLAN, via which it can transmit and receive messages to the MMS Relay/Server. MMS-Ogawa-Cole’s “device messaging agent” thus “further... receive[s] secure Internet data messages over a secure connection via the first WLAN to the network message server,” as claimed for the reasons discussed above in Section VII.D.1.c.

B. GROUND 2B: MMS-Ogawa in View of Sathish (EX-1031)

232. As I discussed above in Section VII.C.5, it was well-documented before the Critical Date to use a software *bus* to enable the interprocess communications described in TS-23.140. EX-1004, 14, 54-56; EX-1028, 729-730, 732-733, FIGS. 4-5. For example, Sathish describes a “D-Bus” as “an example of a device inter-process communication channel used to send information between applications.” EX-1031, 10:56-62.

233. In view of Sathish, a POSA would have been motivated to implement the interface between TS-23.140’s MMS User Agent and the user device’s other applications using a software bus, e.g. D-Bus, and would have reasonably expected

success doing so. Using such a bus to enable TS-23.140's applications to interface would have been nothing more than utilizing familiar, known components to achieve a predictable result of facilitating TS-23.140's communications. For additional documentary evidence corroborating this teaching, *see also* EX-1048, [0001], [0003], [0004], [0019] (describing "communication" between "one or more processes" occurring over a "bus"); EX-1049, 907 (defining "software bus": "A programming interface that allows software modules to transfer data to each other..."); EX-1050, 2:64-3:6 (describing "software bus" that "interconnects" a "control software's subsystems"), 3:51-67, FIG. 5.

234. This combination ("MMS-Ogawa-Sathish"), with the bus secured as I discussed above in Section VII.C.6, renders obvious claims 1, 3-6, and 11-21 for the same reasons discussed above in Section VII for each of those respective claims in the context of MMS-Ogawa (Ground 1).

C. GROUND 2C: MMS-Ogawa-Cole-Sathish

235. A POSA understood that the modifications I describe above in Sections VIII.A and VIII.B are compatible with each other. Moreover, it was feasible and desirable to implement MMS-Ogawa's device to incorporate WWAN/WLAN modems in view of Cole (for the reasons I discussed above in Section VIII.A) *and* a software interprocess communications bus in view of

Sathish (for the reasons I discussed above in Section VIII.B, secured as I discussed above in Section VII.C.6).

236. Such a combination (MMS-Ogawa-Cole-Sathish) renders obvious claims 1-6 and 11-21 for the reasons respectively discussed in the analysis I provided above in Sections VII.D.1-VII.D.16 and VIII.A.2.

IX. GROUNDS 3A-3D

237. Claims 7-9 recite a “service downloader” for downloading, e.g. “updated version[s]” of “software applications” on claim 1’s device. As discussed below, claims 7-9 are rendered obvious by incorporating Papineau’s (EX-1017) teachings.

A. Papineau (EX-1017)

238. Papineau discloses “a method and system for downloading and managing portable applications on a mobile device[,]” and a “Java Application Manager (“JAM”)”—which “is an application that includes a set of functionality that downloads electronic content to the mobile information device... from a information network[.]” EX-1017, 1:33-35, 9:36-39. An application manager may, e.g., download newer versions of existing applications to upgrade them. EX-1017, 10:7-34, 25:39-50 (disclosing JAM loading “MIDlet[s]” (i.e., JAVA Applications) onto the device 12, and using the “version of the MIDlet application” to determine whether to load the application onto the phone). A POSA would have understood

that if the version of the application is new, the older application on the device is updated by the JAM downloading the new version. *Id.*, 10:7-34. The downloaded applications may be “protected.” EX-1017, 22:32-47.

239. Papineau states “[t]he [Java 2 Micro Edition (‘J2ME’)] Mobile Information Device Profile (‘MIDP’) is a set of Java Application Programming Interfaces (API) that provides the runtime environment for J2ME applications[.]” EX-1017, 2:53-56. Papineau incorporates-by-reference “version 1.0” of the MIDP specification. EX-1017, 7:58-59. In this Declaration, I reference Version 2.0 (EX-1025), which published Nov. 5, 2002 and is the version a POSA would have considered on the Critical Date (January 2009); I am informed and understand that the Petition also references this version. “J2ME applications that conform to the MIDP are called ‘MIDlets[.]’” EX-1017, 2:57-59. “MIDlets can be grouped together in a MIDlet Suite by creating a Java Archive (‘JAR’) file.” EX-1017, 8:25-27.

240. Papineau discloses downloading a “Java Application Descriptor (‘JAD’) file” that “provides information to an application manager about the contents of a JAR file[.]” and “[w]ith this information, decisions can be made as to whether or not a MIDlet is suitable for running on the device.” EX-1017, 9:11-14, 8:40-9:8 (Table 1, listing exemplary attributes that may be defined within the manifest file in a JAR). For example, “the JAM... will not download the JAR

file...” if it “determines that any of the Internet media types and/or URI schemes that the JAD file... is attempting to register cannot be registered (e.g., because they are protected media types or URI scheme)[.]” EX-1017, 20:41-48, *see also id.*, 9:15-17 (JAM determines if “the MIDlet requires more persistent memory than the device can provide.”).

241. If the JAM determines (from the JAD file) that the MIDlet is “suitable for running on the device,” the JAR file containing the MIDlet will be downloaded from the location specified in the JAD file’s “MIDlet-Jar-URL” attribute. EX-1017, 23:14-15, 10:7-26. The JAR file may be downloaded from a different server than the JAD file was downloaded from using an absolute URL in the “MIDlet-Jar-URL” attribute. EX-1017, 8:40-9:8 (Table 1, this attribute indicates the URL of the JAR file). The J2ME MIDP Specification (EX-1025) states that “[b]oth absolute and relative URLs MUST be supported” in the “MIDlet-Jar-URL” attribute. EX-1025, 434. V2 also states that “The context for a relative URL is the URL from which this application descriptor was loaded.” *Id.* This explains that the relative URL uses the same full URL path prefix of the .jad file. *Id.*

242. Papineau leaves to a POSA the details of how to ensure the correct JAR file was downloaded. One known option was to have the JAM check the JAR file for indicia (e.g., a signature) verifying the downloaded JAR file’s source. *See* EX-1025, 12 (“The term Application Management Software (AMS) is a generic

term used to describe the software on the device that manages the downloading and lifecycle of MIDlets. This term does not refer to any specific implementation and is used for convenience only. In some implementations, the term Java Application Manager (JAM) is used interchangeably.”). For example, it was known to “protect[]” a “MIDlet suite” “by signing the JAR” file containing the MIDlet. EX-1025, 29. For such “trusted” MIDlet suites, “[t]he signature and certificates are added to the application descriptor”—the JAD file—“as attributes,” which “[t]he device uses... to verify the signature” in the JAR, after it is downloaded, as part of an “authentication” process. *Id.*; *see also* EX-1017, 8:27-29. “The signer of the MIDlet suite may be the developer or some entity... responsible for distributing, supporting, and perhaps billing for its use.” EX-1025, 30. The “signer” has a “public key,” and this “public key is used to verify the signature on the MIDlet suite.” EX-1025, 30; *id.*, 31 (describing “authenticating a MIDlet Suite” using certificates).

B. GROUND 3A: Implementing Papineau’s JAM Teachings (Claims 7-10)

1. MMS-Ogawa-Papineau

243. TS-23.140 discloses applications may be “downloadable” to the user device. EX-1004, 54. TS-23.140 does not disclose details regarding how such downloaded applications, or their updates, are managed. EX-1004, 54-56. As I discuss above in Section IX.A, Papineau describes such details.

244. Papineau teaches using a JAM to manage the download and installation of applications/updates (MIDlets) on a mobile device. EX-1017, 1:33-35, 9:11-17, 9:36-39, 20:41-48, 22:32-47. POSAs would have been motivated to implement Papineau’s JAM in the MMS-Ogawa device because TS-23.140 discloses “the download of a downloadable application to a mobile phone[,]” and Papineau’s JAM was an established, well-documented way of managing TS-23.140’s contemplated application downloads (EX-1004, 54).

245. Papineau further teaches its JAM using a descriptor file (JAD) to verify the suitability of an application/update for a device prior to downloading the corresponding larger JAR file needed to install the application/update. EX-1017, 9:11-14, 10:7-26. POSAs would have been motivated to use JADs as Papineau teaches because this would have helped avoid wasting device resources and network bandwidth on large application installation files for applications not suitable for the device.

246. Moreover, POSAs would have been motivated to use MMS-Ogawa’s existing MM1 interface between the MMS Relay/Server and MMS User Agent (EX-1004, 23-24; *see also* my discussion above in Section VII.C.2) for transporting such JAD files to the device—and using MMS-Ogawa’s MMS User Agent to route the files to the JAM—as this would have leveraged the existing secure connection and existing MMSE infrastructure in MMS-Ogawa to facilitate

TS-23.140's disclosed application downloads. Papineau itself teaches such an implementation, noting that "the ability to invoke content, applications, services, downloads, etc. on the device from a push message is... supported," and "[t]he handling of these messages is the responsibility of the messaging client, which will invoke the JAM"—or otherwise "pass[]" the message to "the JAM... for handling." EX-1017, 22:19-21.

247. Papineau also teaches downloading applications/updates (JAR files) from a separate server, over a communication channel separate from the channel used to download JAD files used to initially determine whether a MIDlet is suitable for the device. EX-1017, 23:14-15; EX-1025, 434 (requiring support for absolute URLs in a JAD file's "MIDlet-Jar-URL" attribute). A POSA would have understood that connections to separate servers were separate data connections. Moreover, a POSA would have been motivated to implement allowing the download of larger application installation files (JAR files) over a separate, dedicated download channel from a separate server (identified in a JAD file)—because this would have helped over-burdening the MMS Relay/Server and the MMSE with transportation of all application data over MM1.

248. Additionally, in view of the specification governing MIDlets as of the Critical Date (see my discussion above in paragraph 239) and Papineau's disclosure of protected MIDlets (EX-1017, 22:32-47), POSAs would have been

motivated to implement the JAM to verify the signature/certificates of a downloaded application/update before installation, e.g. for trusted MIDlets as I described above in Section IX.A. *See also* EX-1025, 29-31. Ensuring that a downloaded MIDlet—e.g., a JAR file that arrived over a non-MM1 connection—is the correct file from a trusted server would have beneficially helped prevent any malicious or tampered software application from being installed on the device.

249. Implementing MMS-Ogawa to include Papineau’s JAM that uses JAD and protected/trusted JAR files as described above (“MMS-Ogawa-Papineau”) would have been nothing more than implementing known components/techniques (a JAM and associated application/update verification and authentication techniques) into known systems/devices (TS-23.140’s user device that uses MMS to transport application data) to achieve predictable results (securely downloading and updating applications).

250. A POSA would have reasonably expected success implementing the above-described techniques into the MMS-Ogawa system given that TS-23.140 expressly contemplates transporting application data for applications that are downloaded onto user devices (EX-1004, 54-56) and because the prior art components above (MMS User Agent, MMS Relay/Server, Papineau’s JAM files) would continue to perform functions they performed prior to the combination. For example, the MMS User Agents and the MMS Relay/Server would continue to

exchange data using secure communication links, while the JAM would manage the download and update of applications, including over a separate communication link from the existing communication link (MM1) to the MMS Relay/Server. Such a combination would have been well within a POSA's capability to implement.

251. In MMS-Ogawa-Papineau, Papineau's JAM is incorporated into MMS-Ogawa's device. To download/update an application (MIDlet), the JAM first downloads a JAD file for the MIDlet over MM1. The JAM determines—from information in the JAD file—whether the MIDlet is suitable for the device. If the JAM determines the MIDlet is suitable for the UE, the JAM downloads a JAR file for the MIDlet from the location specified in the MIDlet-Jar-URL attribute of the JAD file, comprising an absolute URL to a server other than the MMS Relay/Server. After the JAR file is downloaded, the JAM authenticates the JAR file (which is signed) using the signature and certificates from the JAD file. MMS-Ogawa-Papineau renders obvious the claims respectively identified above in Section VII.D for the same reasons described above in that section for Ground 1 in Sections VII.D.1-VII.D.16. In addition, MMS-Ogawa-Papineau renders obvious claims 7-9 as discussed below in Sections IX.B.2-IX.B.4.

2. **Claim 7: The mobile end-user device of claim 1, wherein at least one of the applications comprises a service downloader, the service downloader authenticating a**

downloaded software application based on application data from at least one of the secure Internet data messages.

252. Claim 7 requires that “at least one of the applications comprises a service downloader” that “authenticat[es] a downloaded software application based on application data from at least one of the secure Internet data messages.”

253. When describing a “service downloader,” the ’403 specification describes it as an agent that “provides a download function to install or update service software elements on the device.” EX-1001, 66:46-48. The specification never describes a service downloader “authenticating a downloaded software application *based on application data from at least one of the secure Internet data messages*[.]” and instead leaves to a POSA the details of such authentication, stating that “[a POSA] will appreciate that there are a variety of other security techniques that can be used to ensure the integrity of the service downloader....” EX-1001, 66:55-58. A POSA thus understood that the “authenticat[ion]” of “a downloaded software application based on application data” could be accomplished using “a variety of... techniques[.]” *Id.*

254. MMS-Ogawa-Papineau’s JAM “is an application that includes... functionality that downloads” other applications “to the mobile information device[.]” EX-1017, 9:36-39. A POSA thus understood that MMS-Ogawa-Papineau’s JAM is a “*service downloader*,” as claimed.

255. As I discussed above in Section IX.B.1, MMS-Ogawa-Papineau’s JAM receives a message comprising a JAD file for a MIDlet (“*downloaded software application*”) over the secure connection between the MMS User Agent and the MMS Relay/Server (MM1). Using the information contained in the JAD file, the JAM verifies that the MIDlet is suitable for installation on MMS-Ogawa-Papineau’s user device. EX-1017, 9:11-14, 10:7-26. If the MIDlet is suitable, the JAM downloads a JAR file and authenticates it using the certificates and signature in the JAD file (application data), as I discussed above in paragraphs 242 and 248. A POSA thus understood that MMS-Ogawa-Papineau’s JAM “*authentica[es]*” the MIDlet “*based on application data from at least one of the secure Internet data messages*” (i.e., based on information contained in the JAD file in a message received over MM1), as claimed.

3. **Claim 8: The mobile end-user device of claim 7, wherein the service downloader is to download the downloaded software application using an Internet data connection other than the secure connection used for the secure Internet data messages.**

256. Claim 8 requires that claim 7’s “service downloader” be for “download[ing] the downloaded software application using an Internet data connection other than the secure connection....”

257. In MMS-Ogawa-Papineau, the JAD file (comprising application data used for authenticating the JAR file) is received over MM1, and the JAR file

(comprising the downloaded software application) for a MIDlet is received from a separate server, over a separate Internet data connection, as I discussed above in Section IX.B.1. A POSA thus understood that the JAR file is downloaded over an “Internet data connection other than the secure connection used for the secure Internet data messages” as claimed.

4. Claim 9: The mobile end-user device of claim 7, wherein the downloaded software application comprises an updated version of the device messaging agent.

258. Claim 9 depends from claim 7, and further requires that “the downloaded software application comprises an updated version of the device messaging agent.”

259. In MMS-Ogawa-Papineau, the JAM may upgrade existing applications by downloading newer versions of those applications. EX-1017, 25:39-50, 10:20-34. TS-23.140 discloses that its “MMS User Agent *[is an] application residing on a UE...* that performs MMS-specific operations on a user’s behalf and/or on another application’s behalf.” EX-1004, 14. Papineau expressly discloses implementing its JAM teachings in conjunction with such “messaging client[s].” EX-1017, 22:19-31.

260. A POSA understood that the MMS User Agent of MMS-Ogawa-Papineau could be implemented using JAVA, a well-known, versatile programming language that is not limited in any way that would prevent a POSA

from being able to implement the functionality described for the MMS User Agent in TS-23.140 or contemplated in the combinations I have described herein that incorporate teachings of Ogawa and Papineau.

261. Because MMS-Ogawa-Papineau’s JAM is used to update applications on MMS-Ogawa-Papineau’s device, and because the MMS User Agent is an application on the device, an obvious implementation of MMS-Ogawa-Papineau with which a POSA would have reasonably expected success would have been to implement updates to the MMS User Agent using MMS-Ogawa-Papineau’s JAM, such that the MMS User Agent delivers to the JAM a JAD file (received over MM1) that results in the JAM downloading a JAR file (the claimed “*downloaded software application*”) for later updating the MMS User Agent itself (the claimed “*updated version of the device messaging agent*”), as claimed. An application downloading files for the purpose of ultimately updating itself was little different from downloading files for the purpose of updating another application, and required no special technical skill beyond the level of a POSA to implement.

C. GROUNDS 3B-3D

262. The same reasons I described above in Section IX.B.1 for incorporating Papineau’s teachings into MMS-Ogawa also respectively apply to the MMS-Ogawa-Cole, MMS-Ogawa-Sathish, and MMS-Ogawa-Cole-Sathish combinations that I described above in Sections VIII.A-VIII.C, because Papineau’s

teachings do not affect how and why a POSA would have combined those references.

263. Incorporating Papineau’s teachings into those three combinations render obvious the claims respectively identified above in Sections VIII.A-VIII.C *in addition to* claims 7-9 for the reasons I discuss above in Sections IX.B.2-IX.B.4.

X. GROUNDS 4A-4D

264. Claim 10 recites requirements related to a “secure execution environment” on claim 1’s device. As discussed below, claim 10 is rendered obvious by incorporating the Ellison’s (EX-1019) teachings.

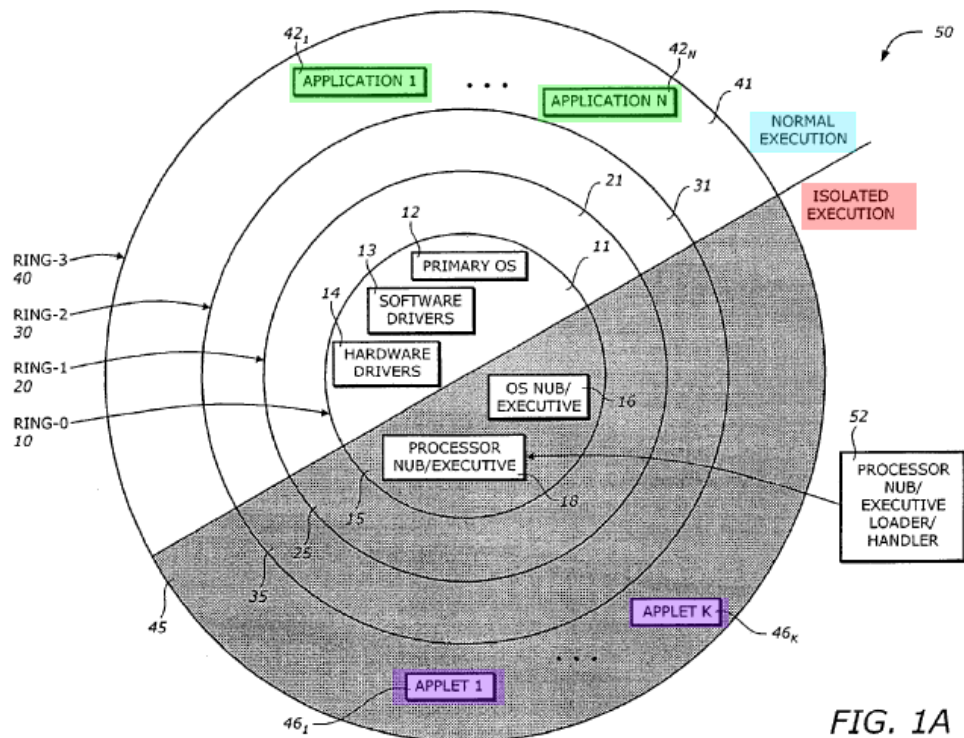
A. Ellison (EX-1019)

265. Ellison discloses techniques for “protect[ing] a subset of a software environment.” EX-1019, Abstract. This includes multiple “operating system nub key[s] (OSNK)” “unique to an operating system (OS) nub.” *Id.* A “usage protector” uses the OSNK to “protect usage of” a software environment’s subset. *Id.*

266. Ellison FIG. 1A (annotated below) “illustrat[es] a logical operating architecture according to” Ellison’s invention. EX-1019, 1:58-59. As shown below, Ellison discloses an “isolated execution mode” (red) where access “is restricted” and a “normal execution mode” (blue) that “operates in a non-secure... or normal

environment” without isolated execution mode’s security features. EX-1019, 1:58-59, 3:4-6; 4:65-5:1, 6:1-26, 8:25-32, FIG. 1A-1C.

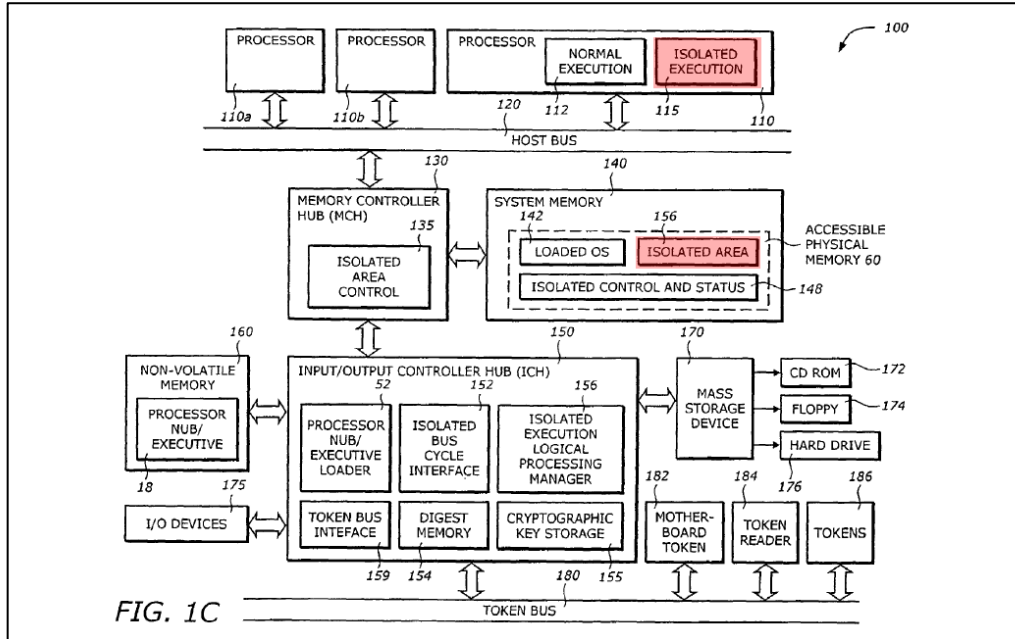
267. Ellison allows “normal” execution for applications (green) outside the device’s secure platform, while “[t]he isolated mode applets [(purple)]... are tamper-resistant and monitor-resistant from all software attacks from... non-isolated space applications[.]” EX-1019, 4:1-3, 4:65-5:1.



EX-1019, FIG. 1A (annotated)

In particular, while elements *outside* Ellison’s isolated area “*cannot* access the isolated area,” elements within the isolated area “*can* access”—e.g., transport data to/from—the “non-isolated area.” EX-1019, 4:30-37.

268. “[W]hen operating in isolated execution mode,” Ellison describes processor 110 defining an “isolated area 70” in memory. EX-1019, 6:1-26, 8:25-32, FIGS. 1A-1C, 2. FIG. 1C, showing the isolated area, is reproduced below:



EX-1019, FIG. 1C (annotated)

269. FIG. 2 discloses a “secure platform” implementing Ellison’s security techniques, which may wholly “operat[e] within the isolated execution environment[.]” EX-1019, 6:1-26, 8:25-32, FIGS. 1A-1C, FIG. 2.

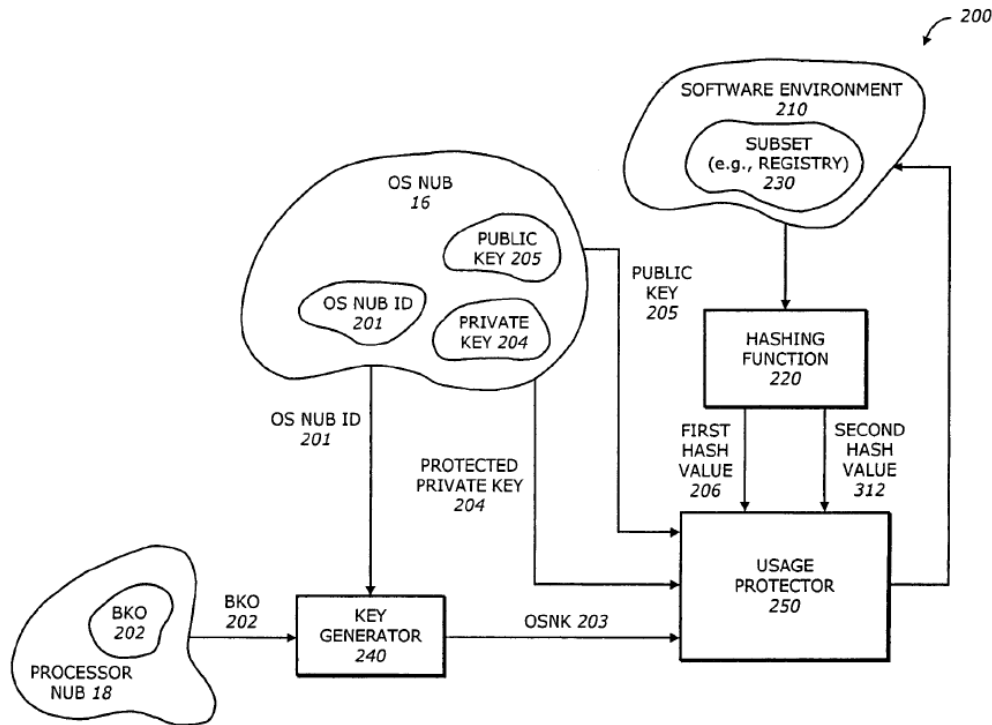


FIG. 2

270. FIG. 2's secure platform includes "key generator 240" that "generates a key operating system nub key (OSNK) 203" supplied to "usage protector 250" that uses OSNK 203 to protect subset 230's usage and OSNK 203 is supplied to "trusted agents." *Id.*, 8:66-9:40, FIG. 2. One example of a trusted agent is a "usage protector 250" that "uses the OSNK 203 to protect the usage of [a] subset 230." *Id.*, 9:28-40, FIG. 2. Usage protector 250 uses a hashing function with subset 230's OSNK 203 to determine if it has been altered (e.g., reads/writes to the subset 230), thereby protecting against "unauthorized reads" and detecting "intrusion, tampering or unauthorized modification." *Id.*, 9:41-62.

B. GROUND 4A: Implementing Ellison’s Isolated Execution Environment (Claim 10)

1. MMS-Ogawa-Ellison

271. A POSA had multiple reasons to implement MMS-Ogawa’s UEs in view of Ellison’s above-described teachings of a secure platform which operates within a secure, isolated area of the user device.

272. *First*, implementing Ellison’s secure platform (EX-1019, 8:25-9:62, FIG. 2) would have desirably improved device security in the MMS-Ogawa system, which transports data on behalf of applications downloaded to the device, e.g., third-party VASP applications that should not be trusted with access to sensitive device data/functions. EX-1004, 41, 54-56. The use of such secure platforms for improving device security was well-known and well-documented. *See, e.g.*, Needham (EX-1030) (describing protocol that formed basis of security protocols (e.g., Kerberos) included in many operating systems from the late 1980s and beyond); Schroeder (EX-1036) (describing “A Hardware Architecture for Implementing Protection Rings,” where operating systems could allow not only for the separation of different processes depending on their security needs, but also allow for communication across such security boundaries through highly-controlled mechanisms); Saltzer (EX-1032) (describing process separation and controlled communication between processes), Li (EX-1033), 25-26 (describing the Symbian mobile operating system, which separated processes into three “Trust

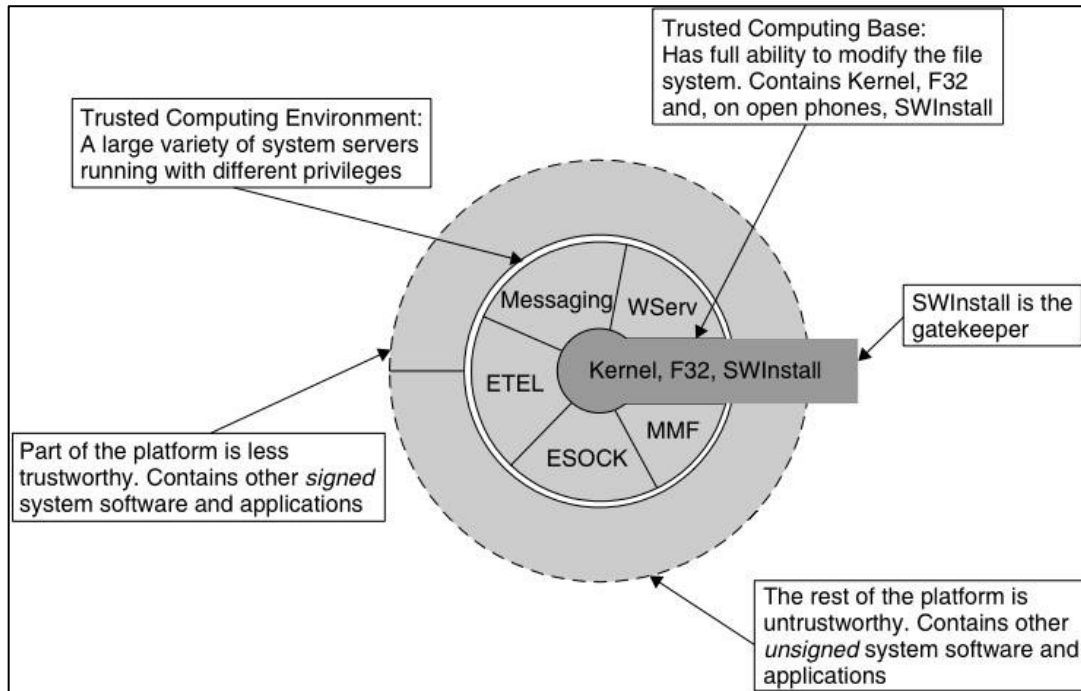
Tiers”: The Trusted Computing Base (TCB), The Trusted Computing Environment (TCE), and Applications, and thereby implemented protection rings/tiers around different device components,), FIG. 1. A POSA would have been motivated to incorporate Ellison’s teachings because that would have enabled intra-device message transmission with increased security and helped guard against and prevent malicious actions (malware, unauthorized access). EX-1019, 8:25-9:62, FIG. 2.

273. **Second**, combining TS-23.140 and Ellison’s teachings merely involved combining known prior art elements (Ellison’s isolated software environment in MMS-Ogawa’s device) according to known methods to yield predictable results.

274. A POSA would have reasonably expected success incorporating Ellison’s techniques into MMS-Ogawa’s user devices because Ellison expressly describes that its techniques can be implemented in “computer system[s]” (like MMS-Ogawa’s UEs) with “processor[s].” EX-1019, 5:11-16.

275. Moreover, it was well-known for computing systems to include the messaging services and clients/agents responsible for providing those services, within a trusted/secure computing environment, and for applications (e.g., downloaded third-party destination applications to which messages are directed) to be tiered/organized outside this environment. This is confirmed, for example, by Li (EX-1033). Li establishes that in the early 2000s, the Symbian mobile operating

system separated processes into three “Trust Tiers”: The Trusted Computing Base (TCB), The Trusted Computing Environment (TCE), and Applications, as illustrated below:



As shown above, the first two layers/tiers shown were primarily dedicated to components of the Operating System with relatively higher operating privileged, followed by the Applications tier with applications having relatively lower privileges. The first two tiers included the Trusted Computing Base and the Trusted Computing Environment. As shown, the Trusted Computing Environment includes, among other services, a Messaging service. The Application Tier could be further subdivided into Signed and Unsigned applications. Signed applications could interact with the TCE and have this layer execute higher privileged requests, whereas Unsigned applications could not access that same functionality.

276. Given (1) the fact that TS-23.140 expressly contemplates a user device with an MMS User Agent that communicates with downloaded applications on the user device (e.g., a downloaded application trying to upload messages transporting application data to the MMS Relay/Server via the MMS User Agent), (2) the well-known need to protect computing environments (including components providing essential services such as messaging) from other applications/components of the environment, and (3) Ellison’s own disclosures confirming the same, a POSA would have been motivated to implement Ellison’s teaching of a secure execution environment that enables such improved security within the MMS environment, such that the MMS User Agent is organized as part of the trusted computing environment (i.e., the secure execution environment) and enables communications with applications residing outside this trusted/secure environment. As noted above, Ellison expressly discloses elements *outside* of Ellison’s isolated area not being able to “access the isolated area,” while elements within the isolated area “can access”—e.g., transport data to/from—the “non-isolated area.” EX-1019, 4:30-37.

277. In such an implementation (“MMS-Ogawa-Ellison”), Ellison’s secure platform is incorporated into TS-23.140’s user devices to secure the MMS User Agent, with a usage protector protecting messaging operations of the MMS User Agent. EX-1019, 8:25-32, 8:66-9:62, FIG. 2. The MMS User Agent runs within

subset 230 of software environment 210 in “isolated execution mode,” while at least one device applications runs in “normal execution mode.” EX-1019, 3:4-6. As noted (e.g., paragraphs 268-270), Ellison allows for “normal” execution for applications outside of the secure execution environment on that device. EX-1019, 4:65-67, 5:1, 6:1-26, FIGS. 1A, 1C. Ellison’s usage protector provides secure access to applications residing outside the secure execution environment. EX-1019, 8:25-32, 8:66-9:62, FIG. 2.

2. Claim 10: The mobile end-user device of claim 1, wherein the device messaging agent runs in a secure execution environment on the device, and at least one of the applications runs outside of the secure execution environment on the device.

278. Claim 10 requires that the “device messaging agent” runs in a “secure execution environment on the device,” while “at least one of the applications runs outside of the secure execution environment on the device.”

279. MMS-Ogawa-Ellison, which incorporates Ellison’s secure platform teachings into MMS-Ogawa’s device, meets claim 10. A POSA understood that Ellison’s secure, isolated area is a “*secure execution environment*” as claimed. A POSA further understood that MMS-Ogawa-Ellison’s MMS User Agent runs in the “secure execution environment on the device,” while at least one un-trusted third-party device application “*runs outside of the secure execution environment on the device*” (see my discussion above in Section X.B.1). As discussed, the

MMS User Agent, which handles potentially sensitive user data in transit over the network, is run in the isolated environment of MMS-Ogawa-Ellison. By contrast, destination applications on the user device—which do not directly transmit such data outside of the device or establish the connection between the User Agent and the Relay/Server—are run outside the isolated environment because they require lower security due to handling lower sensitivity information (e.g., chess game handling chess moves). EX-1004, 55.

280. As required by Element [1C1], MMS-Ogawa-Ellison’s “device messaging agent” still has authorized communications with the “application[] run[ning] outside of the secure execution environment” over the “secure interprocess communication service,” as claimed. *E.g.*, EX-1019, 4:30-37.

C. GROUNDS 4B-4D

281. For claims 1-6 and 13-21, the same reasons I described above in Section X.B.1 for incorporating Ellison’s teachings into MMS-Ogawa also apply to the MMS-Ogawa-Cole, MMS-Ogawa-Sathish, and MMS-Ogawa-Cole-Sathish combinations, which I described above in Sections VIII.A-VIII.C), because a POSA understood that Ellison’s teachings do not affect how and why a POSA would have combined those references for those claims.

282. Incorporating Ellison’s teachings into those three combinations render obvious claims 1-6 and 13-21 for the reasons I respectively discussed above in

Sections VIII.A-VIII.C—in addition to claim 10, for the reasons I discussed above in Section X.B.2.

XI. CLAIM LISTING APPENDIX

Claim 1
[1PRE] A mobile end-user-area device comprising:
[1A] wireless wide-area network (WWAN) modem to exchange Internet data via a connection to a first WWAN, when configured for and connected to the first WWAN;
[1B1] a device messaging agent to receive secure Internet data messages, on behalf of a plurality of software applications capable of execution on the device, and over a secure connection to a network message server reachable via the WWAN,
[1B2] wherein at least a subset of the secure Internet data messages contain an identifier for a corresponding one of the software applications and application data from a respective network application server corresponding to that application; and
[1C1] a secure interprocess communication service,
[1C2] wherein the device messaging agent, for each message in the subset of the secure Internet data messages, maps the identifier to the corresponding one of the software applications in order to forward the application data on the secure interprocess communication service to a software process corresponding to the identified software application.
Claim 2
The mobile end-user device of claim 1, further comprising a wireless local area network (WLAN) modem to exchange Internet data via a connection to a first WLAN, when configured for and connected to the first WLAN, the device messaging agent further to receive secure Internet data messages over a secure connection via the first WLAN to the network message server.
Claim 3
The mobile end-user device of claim 1, further comprising the plurality of software applications.

Claim 4

The mobile end-user device of claim 3, wherein the plurality of applications include a first application that receives the application data in a first format, and a second application that receives the application data in a second format different than the first format.

Claim 5

The mobile end-user device of claim 1, wherein the secure Internet data messages are received encrypted, the device messaging agent decrypting each message in the subset to obtain the corresponding identifier and application data.

Claim 6

The mobile end-user device of claim 5, wherein the secure Internet data messages are transported to the device messaging agent using one or more of encryption on a transport services stack, IP (Internet Protocol) layer encryption, and transport via a tunnel.

Claim 7

The mobile end-user device of claim 1, wherein at least one of the applications comprises a service downloader, the service downloader authenticating a downloaded software application based on application data from at least one of the secure Internet data messages.

Claim 8

The mobile end-user device of claim 7, wherein the service downloader is to download the downloaded software application using an Internet data connection other than the secure connection used for the secure Internet data messages.

Claim 9

The mobile end-user device of claim 7, wherein the downloaded software application comprises an updated version of the device messaging agent.

Claim 10

The mobile end-user device of claim 1, wherein the device messaging agent runs in a secure execution environment on the device, and at least one of the applications runs outside of the secure execution environment on the device.

Claim 11

[11A] The mobile end-user device of claim 1, wherein the device messaging agent is further to send secure upload Internet data messages to the network message server over the secure connection, wherein at least a subset of the secure upload Internet data messages are sent responsive to a corresponding request received on the secure interprocess communication channel from a corresponding one of the software applications, the device messaging agent constructing from each such request a secure upload Internet data message containing

[11B] an identifier for a respective network application server corresponding to the requesting software application; and

[11C] content received with the request.

Claim 12

The mobile end-user device of claim 11, wherein at least one of the upload Internet data messages comprises a key for the network application server corresponding to the requesting software application.

Claim 13

The mobile end-user device of claim 1, wherein the device messaging agent creates a log for the received secure Internet data messages.

Claim 14

The mobile end-user device of claim 1, wherein the secure interprocess communication channel and the secure connection to the network message server are separately secured.

Claim 15

The mobile end-user device of claim 1, wherein access by the software applications to the interprocess communication channel is subject to a security policy

Claim 16

The mobile end-user device of claim 1, wherein at least one of the secure Internet data messages comprises multiple identifier/data pairs.

Claim 17

The mobile end-user device of claim 1, the device messaging agent comprising an agent router to forward the application data on the secure interprocess communication channel to the software process corresponding to the identified software application.

Claim 18

The mobile end-user device of claim 1, wherein the secure interprocess communication service forwards the application data to at least one of the software processes in an encrypted format.

Claim 19

The mobile end-user device of claim 1, the device messaging agent further to initiate the secure connection to the network message server.

Claim 20

The mobile end-user device of claim 1, further comprising a network stack in communication with the device messaging agent and the WWAN modem, the secure connection terminated within the network stack.

Claim 21

The mobile end-user device of claim 1, wherein at least one of the applications and the network application server corresponding to that application authenticate with each other prior to passing application data via the device messaging agent and network message server.

I declare that all statements made herein of my own knowledge are true, that all statements made on information and belief are believed to be true, and that these statements were made with the knowledge that willful false statements and the like are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code.

I declare under penalty of perjury that the foregoing is true and correct.

Dated: 20 January 2026



Dr Patrick G. Traynor

APPENDIX A

Patrick Gerard Traynor

Professor

Associate Chair for Research in CISE

John and Mary Lou Dasburgh Preeminent Chair in Engineering

Department of Computer & Information Science & Engineering (CISE)

University of Florida

1889 Museum Rd,

Gainesville, FL 32611 USA

`traynor@cise.ufl.edu`

`http://www.cise.ufl.edu/~traynor`

Table of Contents

EDUCATIONAL BACKGROUND	4
EMPLOYMENT HISTORY	4
CURRENT FIELDS OF INTEREST	4
I. TEACHING	6
A. Courses Taught	6
B. Continuing Education	6
C. Curriculum Development	6
D. Individual Student Guidance	7
E. Teaching Honors and Awards	12
II. RESEARCH AND CREATIVE SCHOLARSHIP	13
A. Thesis	13
B. Published Journal Papers (Refereed)	13
C. Published Books and Parts of Books	15
D. Edited Proceedings	15
E. Conference Presentations	15
E.1. Conference Papers with Proceedings (Refereed)	15
E.2. Conference Presentations with Proceedings (Non-Refereed)	23
E.3. Conference Presentations without Proceedings	23
F. Other	23
F.1. Submitted Journal Papers	23
F.2. Refereed Research Reports	23
F.3. Software	23
F.4. Published Papers (Non-Refereed)	23
F.5. Books in Preparation	23
F.6. Workshops and External Courses	24
G. Research Proposals and Grants (Principal Investigator)	24
H. Research Proposals and Grants (Contributor)	27
I. Research Honors and Awards	28
III. SERVICE	29
A. Professional Activities	29
A.1. Memberships and Activities in Professional Societies	29
A.2. Conference Committee Activities	29
B. On-Campus Committees	30
B.1. University of Florida	30
B.2. Georgia Tech	31
C. Special Assignments	31
D. Ph.D. Examining Committees	31
E. External Member of M.S. Examining Committee	35
F. Consulting and Advisory Appointments	35
G. Civic Activities	35
IV. NATIONAL AND INTERNATIONAL PROFESSIONAL RECOGNITION	36
A. Honors and Awards	36
B. Invited Conference Session Chairmanships	36
C. Professional Registration	36
D. Patents	36
E. Editorial and Reviewer Work for Technical Journals and Publishers	37
F. Expert Witness Services	39
V. OTHER CONTRIBUTIONS	41
A. Seminar Presentations (Invited Papers and Talks at Meetings and Symposia)	41

B. Special Activities 45

EDUCATIONAL BACKGROUND

Degree	Year	University	Field
Ph.D.	2008	Pennsylvania State University State College, PA <i>Dissertation:</i> Characterizing the Impact of Ridigity on the Security of Cellular Telecommunications Networks <i>Advisors:</i> Thomas F. La Porta and Patrick D. McDaniel	Computer Science & Engineering
M.S.	2004	Pennsylvania State University State College, PA	Computer Science & Engineering
B.S.	2002	University of Richmond Richmond, VA <i>Minors:</i> Biology, Business Admin	Computer Science

EMPLOYMENT HISTORY

Title	Organization	Years
Interim Department Chair	University of Florida	July 2025–Present
Associate Chair for Research	University of Florida	August 2018–Present
Professor	University of Florida	August 2018–Present
Associate Professor	University of Florida	August 2014–July 2018
Associate Professor	Georgia Institute of Technology	March 2014–August 2014
Assistant Professor	Georgia Institute of Technology	2008–March 2014
Research Assistant	Pennsylvania State University	2004–2008
Teaching Assistant	Pennsylvania State University	2004

CURRENT FIELDS OF INTEREST

My research focuses on the security of cellular/telephony networks and mobile systems. The security of these systems generally relies on their closed nature and trust in the honest behavior of users. However, with the recent disintegration of these assumptions and with over than six billion subscribers around the world, cellular and mobile systems represent the next great expansion in global critical infrastructure and, because of their unique characteristics, require new and different approaches to security.

Recognizing this, my research focuses on three specific themes: (1) developing efficient techniques to allow telephony providers and customers to authenticate the origin of incoming calls; (2) measuring and

improving the security of emerging mobile financial systems and (3) efficient and strong privacy-preserving techniques for mobile devices. Additionally, I have significant expertise in fraud detection, particularly for payment systems.

I have a strong interest in solutions that can be deployed in both the short and long terms, and am actively engaging both industry and government in this capacity. My research, if successful, will help to not only improve the general security of networked devices, but also to maintain the historical reliability of telephony networks as they become the dominant digital access technology.

I. TEACHING

A. Courses Taught

Semester/Year	Course Number & Title	Number of Students	Comments
Fall 2024	CNT 4007 Computer Networks 1	310	Revamped Course
Fall 2023	CIS 6930 Cellular and Mobile Network Security	19	New Topics
Fall 2022	CNT 5410 Computer and Network Security	75	New Topics
Fall 2021	CNT 5410 Computer and Network Security	45	New Topics
Fall 2019	CNT 5410 Computer and Network Security	28	New Topics
Fall 2018	CIS 6930 Cellular and Mobile Network Security	16	New Course
Fall 2017	CNT 5410 Computer and Network Security	27	New Topics
Fall 2016	CNT 5410 Computer and Network Security	60	New Topics
Spring 2016	CNT 5410 Computer and Network Security	13	New Topics
Spring 2015	CNT 5410 Computer and Network Security	12	New Topics
Fall 2014	CNT 5410 Computer and Network Security	30	New Course
Spring 2014	CS 6262 Network Security	55	New Projects
Fall 2013	CS 3251 Computer Networks I	73	Expanded Syllabus
Spring 2013	CS 6262 Network Security	65	All New Projects
	CS 8001 Information Security Seminar	20	New Speakers
Fall 2012	CS 8803 Cellular & Mobile Network Security	17	New Topics
	CS 8001 Information Security Seminar	20	New Speakers
Spring 2011	CS 8001 Information Security Seminar	20	New Speakers
Fall 2011	CS 6262 Network Security	27	Expanded Syllabus
	CS 8001 Information Security Seminar	35	New Speakers
Spring 2011	CS 3251 Computer Networks I	61	Expanded Syllabus
	CS 8001 Information Security Seminar	20	New Speakers
Fall 2010	CS 8803/4803 Cellular & Mobile Network Security	16	New Course
	CS 8001 Information Security Seminar	31	New Speakers
Fall 2009	CS 6262 Network Security	55	Expanded Syllabus
Spring 2009	CS 3251 Computer Networks I	45	Expanded Syllabus
Fall 2008	CS 8003 Destructive Research	10	New Course

Guest lecturer for CS 4235 (Introduction to Information Security) and CS 8803 (e-Democracy) in Fall 2008.

Advised ECE 4811/CS 4802 (Vertically Integrated Project) with Ed Coyle

B. Continuing Education

None.

C. Curriculum Development

University of Florida

CNT 4007 Computer Networks 1: *Fall 2024.* Provided the first major overhaul to this course in a number of years. While I have relied on the same book used by other faculty, I have created new homeworks, projects, and slides to better represent the current state of computer networks. I have also significantly expanded the discussion of security in this course.

CIS 6930 Cellular and Mobile Network Security: *Fall 2018, 2023.* Developed an entirely new course around security issues facing cellular and mobile networks. Students learned about wireless basics, spectrum issues, core network architectures (GSM, ISDN, IMS, SIP), air interfaces (2G-5G), mobility management, authentication, mobile phone operating systems (Android, iPhone), Android security, congestion and denial of service, privacy and eavesdropping. Students also complete a research project and aim towards publishing this work at a major venue. My aim is for this class to become part of the regular offering of security courses. Semester projects were also judged and encouraged using a “venture capital” model, in which students had to pretend as if they were pitching their ideas for a start-up company to potential investors.

CNT 5410 Computer and Network Security: *Fall 2014-2022.* Totally rewrote the syllabus and slide material, giving the class its first major overhaul in a number of years. While many old themes remain, new lecture blocks including Web Security, Cellular Security and Social Engineering were developed from scratch. This new course material was made available to all other faculty members teaching this class, who have since used my slides and syllabus.

Georgia Tech In addition to the above courses, I also developed the following course while serving as a faculty member at Georgia Tech.

CS 3251 Computer Networks I: *Spring 2009.* Modified undergraduate networking course to include a persistent focus on security at all layers of the protocol stack. I have also created new lectures focusing on the physical layer and cellular networks and new exams to include all of the abovementioned changes.

CS 8803 Destructive Research: *Fall 2008.* Developed course based around understanding how so-called secure systems have been defeated by attackers. With such knowledge, students would have the context to develop the next generation of more secure systems. I delivered more than 1/3 of the lectures in this seminar course and paid special focus on vulnerabilities in cellular networks, analog telecommunications and electronic voting. Students were also instructed on techniques for performing research, writing technical papers and making conference and lecture-style presentations. I have offered these slides to future 7001 classes to help impact a wider audience.

D. Individual Student Guidance

1. Research Scientists Supervised

None.

2. Ph.D. Students Graduated

Hadi Abdullah University of Florida

Fall 2016–Summer 2022

Evaluating the security of ML-driven voice interfaces. Now: Research Scientist at Visa Research

Chaitrali Amrutkar Georgia Institute of Technology

Fall 2009–Fall 2013

Her research discovered vulnerabilities in mobile web browsers and developed techniques to detect malicious mobile web pages. Joined Oracle in Spring 2014.

- Logan Blue** University of Florida
Fall 2016–Summer 2022
Investigated biological feature reconstruction from voice recordings. Now: Research Scientist at Harbor Labs
- Jasmine Bowers** University of Florida
Fall 2015–Summer 2020
Her research focuses on mobile applications, and the development of tools for building secure systems. Now: Research Scientist, MITRE
- Henry “Hank” Carter** Georgia Institute of Technology
Fall 2010–Spring 2016
Developing techniques for secure function evaluation for privacy-preserving applications on constrained mobile devices. Now: Assistant Professor, Villanova University
- Italo Dacosta** Georgia Institute of Technology
Fall 2008–Summer 2012
Co-advised with Mustaque Ahamad. Research on scaling performance of SIP network components. Graduated Summer 2012, currently research scientist at EPFL.
- David Dewey** Georgia Institute of Technology
Fall 2011–Summer 2015
Investigated compiler techniques to remove software vulnerabilities from code. Now CTO of MailChimp.
- Cassidy Gibson** University of Florida
Fall 2019–Spring 2025
Investigated the use of AI in generating non-consensual imagery. Now Research Scientist at ActiveFence.
- Seth Layton** University of Florida
Fall 2020–Summer 2025
Detecting deepfakes in audio samples. Now Research Scientist at PinDrop.
- Christian Peeters** University of Florida
Fall 2016–Summer 2022
Develop techniques to detect and defend against call and message interception attacks in cellular networks. Now: Research Scientist at Harbor Labs
- Brad Reaves** University of Florida
Fall 2014–Spring 2017
Develop strong authentication techniques for cellular networks. Now: Assistant Professor at North Carolina State University.
- Nolen Scaife** University of Florida
Fall 2014–Spring 2019
Developed techniques to detect credit card skimming. First: Assistant Professor at the University of Colorado Boulder. Now: Director, Global Cyber Intelligence at Walmart
- Imani Sherman** University of Florida
Fall 2018–Summer 2021
Developing usable interfaces against robocalls. Co-advised with Juan Gilbert. Now: Assistant Professor at the University of California, San Diego
- Tyler Tucker** University of Florida
Fall 2021–Summer 2025
Evaluating the security of Bluetooth/cellular radios. Now: Associate at the Analysis Group.

Luis Vargas University of Florida
Fall 2016–Summer 2021
Developing techniques for network-based detection and mitigation of malware in a healthcare environment. Now: Data Scientist at the Alethia Group

Kevin Warren University of Florida
Fall 2019–Summer 2025
Detecting deepfake audio through linguistic information. Now: Associate at the Analysis Group.

2. Ph.D. Students Supervised

Nathaniel Bennett University of Florida
Fall 2022–Present
Finding vulnerabilities in cellular core networks via fuzzing.

Jordan Greene University of Florida
Fall 2025–Present
Cellular network security.

Allison Lu University of Florida
Fall 2022–Present
Measuring repeatability in computer security.

Daniel Olszewski University of Florida
Fall 2019–Present
Removing unwanted/insecure features from software.

Aviva Smith University of Florida
Fall 2025–Present
Detecting Deepfakes.

3. Ph.D. Students - Other

Saurabh Chakradeo Georgia Institute of Technology
Fall 2010–Spring 2013
Research exploring malicious mobile applications. Left to join Facebook.

Brendan Dolan-Gavitt Georgia Institute of Technology
Spring 2009
Research project on using kernel type graphs to detect dummy structures.

Ryon Kennedy University of Florida
Fall 2020–Spring 2023
Finding vulnerabilities in cellular core networks via fuzzing. Left to join UFIT.

Eric (Yu) Liu Georgia Institute of Technology
Fall 2008
Research on the spread of malcode through cellular infrastructure.

Chaz Lever Georgia Institute of Technology
Fall 2011–Spring 2014
Developing techniques to measure the spread of malware in cellular networks. Left Georgia Tech to create a startup.

Frank Park Georgia Institute of Technology

Fall 2008–Spring 2010

Research on multi-factor authentication using cellular phones. Left program after failing comprehensive exam to join startup.

Ferdinand Schober Georgia Institute of Technology

Fall 2009–Summer 2010

Developed mechanisms for smart networks and smart mobile devices to fight infection and provide remote remediation. Returned to Microsoft.

4. M.S. Students Supervised

Chaitrali Amrutkar Georgia Institute of Technology

Fall 2008–Spring 2009

Research on improving performance of security critical functions in IMS cellular core. Completed her Ph.D with me at GT.

Logan Blue University of Florida

Fall 2015–Spring 2016

Investigated problems of cellular and network security.

David Dewey Georgia Institute of Technology

Fall 2009–Spring 2010

Research on security issues caused by transitive trust assumptions in the Windows COM infrastructure. Completed his Ph.D. with me at GT.

Christopher Grayson Georgia Institute of Technology

Fall 2012–Fall 2013

Developed continuous authentication mechanisms using the multitude of sensors available on a mobile phone. Now at Bishop Fox Consulting (industry).

Young Seuk Kim Georgia Institute of Technology

Fall 2012–Fall 2013

Performed research that compared the security vulnerabilities found in the traditional and mobile web.

Daniel Komaromy Georgia Institute of Technology

Fall 2008–Summer 2009

Research on building a real-time streaming audio system using attribute-based crypto for broadcast encryption.

Nigel Lawrence Georgia Institute of Technology

Fall 2011–Spring 2012

Discovered hijacking attacks in SNMPv3, a widely used and thought to be secure network management protocol. Now at Solute (industry).

Philip Marquardt Georgia Institute of Technology

Fall 2009–Present

Research on developing an iPhone application to prevent individuals from being profiled by Shopper Loyalty Programs. First with MIT Lincoln Labs, now Raytheon

Rishikesh Naik Georgia Institute of Technology

Fall 2008–Spring 2010

Research on converting expensive cryptographic primitives (e.g., Secure Function Evaluation) into efficient applications for mobile phones. Now with Cisco Systems.

Ashish Nautiyal University of Florida

Fall 2015–Spring 2016

Research on connecting telephone calls to the larger authentication infrastructure.

Nilesh Nipane Georgia Institute of Technology

Fall 2008–Spring 2010

Research on creating provably anonymous networks on a base of secure function evaluation. Now with VMWare.

Walter “Nolen” Sciafe Georgia Institute of Technology

Spring 2012–Spring 2014

Developed the OnionDNS architecture, which prevents domain delisting attacks by leveraging a Tor hidden service. Joined Ph.D. program at UF.

Tyler Tucker University of Florida

Fall 2018–Spring 2021

Evaluating the security of Bluetooth radios.

5. M.S. Special Problems Students

Siddhant Deshmukh University of Florida

Fall 2016–Present

Developed tools for analysis of mobile digital financial services.

Chinmay Gangakhedkar Georgia Institute of Technology

Spring 2009

Research on multi-factor authentication using mobile phones.

Christopher Grayson Georgia Institute of Technology

Spring 2013

Research on continuous authentication using mobile phones.

Aarushi Karnany University of Florida

Fall 2016–Present

Developed tools for analysis of mobile digital financial services.

Rohit Matthews Georgia Institute of Technology

Spring 2011

Developed mobile phone-based tools for measuring performance and reachability throughout the Internet.

Ashwin Narasimhan Georgia Institute of Technology

Spring 2009

Research on developing efficient security mechanisms for the IMS cellular core.

Aamir Poonawalla Georgia Institute of Technology

Spring 2010

Helped develop a call provenance infrastructure, which included both networking and machine learning components.

Erin Reddick Georgia Institute of Technology

Fall 2008–Fall 2009

Research on IPTV security with GTRI.

Lalanthika Vasudevan Georgia Institute of Technology

Spring 2009

Research on developing efficient security mechanisms for the IMS cellular core.

6. Undergraduate Special Problems Students

Ethan Shernan Georgia Institute of Technology

Spring 2014

Developed an infrastructure for detecting billing bypass fraud attacks.

Young Seuk Kim Georgia Institute of Technology

Fall 2011–Spring 2012

Developed a mobile phone application for taking measurements of cellular networks.

Dane Van Dyck Georgia Institute of Technology

Summer 2009

Research on virtualization support for mobile phones.

E. Teaching Honors and Awards

1. Georgia Institute of Technology, CETL “Thanks For Being A Great Teacher” Award, Fall 2013.
2. Georgia Institute of Technology, CETL “Thanks For Being A Great Teacher” Award, Fall 2012.
3. United State Army Signal Corps, “Helmet” Award, 2010.
4. Georgia Institute of Technology, CETL “Thanks For Being A Great Teacher” Award, Spring 2009.
5. Pennsylvania State University CSE Graduate Student Teaching Award, 2005

II. RESEARCH AND CREATIVE SCHOLARSHIP

A. Thesis

1. Patrick Gerard Traynor. *Characterizing the Impact of Rigidity on the Security of Cellular Telecommunications Networks*. PhD thesis, The Pennsylvania State University, May 2008.

B. Published Journal Papers (Refereed)

1. Cassidy Gibson, Vanessa Frost, Katie Platt, Washington Garcia, Luis Vargas, Sara Rampazzi, Vincent Bindschaedler, Patrick Traynor, and Kevin Butler. Analyzing the Monetization Ecosystem of Stalkerware. *Proceedings on Privacy Enhancing Technologies (PoPETs)*, 2022. (Acceptance rate: 24%).
2. Bradley Reaves, Luis Vargas, Nolen Scaife, Dave Tian, Logan Blue, Patrick Traynor, and Kevin Butler. Characterizing the Security of the SMS Ecosystem with Public Gateways. *ACM Transactions on Privacy and Security (TOPS)*, 22(1), 2018.
3. Patrick Traynor, Kevin Butler, Jasmine Bowers, and Bradley Reaves. FinTechSec: Addressing the Security Challenges of Digital Financial Services. *IEEE S&P Magazine*, 15(5):85–89, 2017.
4. Nolen Scaife, Henry Carter, Rachel Jones, Lyrissa Lidsky, and Patrick Traynor. OnionDNS: A Seizure-Resistant Top-level Domain. *International Journal of Information Security (IJIS)*, 2017.
5. Bradley Reaves, Jasmine Bowers, Nolen Scaife, Adam Bates, Arnav Bharatiya, Patrick Traynor, and Kevin Butler. Mo(bile) Money, Mo(bile) Problems: Analysis of Branchless Banking Applications in the Developing World. *ACM Transactions on Privacy and Security (TOPS)*, 2017.
6. Henry Carter and Patrick Traynor. OPFE: Outsourcing Computation for Private Function Evaluation. *International Journal of Information and Computer Security (IJICS)*, 2017.
7. Stephan Heuser, Bradley Reaves, Praveen Kumar Pendyala, Henry Carter, Alexandra Dmitrienko, William Enck, Negar Kiyavash, Ahmad-Reza Sadeghi, and Patrick Traynor. Phonion: Practical Protection of Metadata in Telephony Networks. *Proceedings on Privacy Enhancing Technologies (PoPETs)*, 2017.
8. Bradley Reaves, Jasmine Bowers, Sigmond A. Gorski III, Olabode Anise, Rahul Bobhate, Raymond Cho, Hiranava Das, Sharique Hussain, Hamza Karachiwala, Nolen Scaife, Byron Wright, Kevin Butler, William Enck, and Patrick Traynor. *droid: Assessment and Evaluation of Android Application Analysis Tools. *ACM Computing Surveys (CSUR)*, 2016.
9. Chaitrali Amrutkar, Young Seuk Kim, and Patrick Traynor. Detecting Mobile Malicious Webpages in Real Time. *IEEE Transactions on Mobile Computing (TMC)*, To Appear 2016.
10. Henry Carter, Benjamin Mood, Patrick Traynor, and Kevin Butler. Outsourcing Secure Two-Party Computation as a Black Box. *Journal of Security and Communication Networks (SCN)*, To Appear 2016.
11. Henry Carter, Benjamin Mood, Patrick Traynor, and Kevin Butler. Secure Outsourced Garbled Circuit Evaluation for Mobile Devices. *Journal of Computer Security (JCS)*, 24(2):137–180, 2016.
12. Adam Bates, Kevin Butler, Micah Sherr, Clay Shields, Patrick Traynor, and Dan Wallach. Accountable Wiretapping -or- I Know They Can Hear You Now. *Journal of Computer Security (JCS)*, 23(2):167–195, 2015.
13. Henry Carter, Chaitrali Amrutkar, Italo Dacosta, and Patrick Traynor. For Your Phone Only: Custom Protocols for Efficient Secure Function Evaluation on Mobile Devices. *Journal of Security and Communication Networks (SCN)*, 7(7):1165–1176, 2014.

14. Chaitrali Amrutkar, Patrick Traynor, and Paul van Oorschot. An Empirical Evaluation of Security Indicators in Mobile Web Browsers. *IEEE Transactions on Mobile Computing (TMC)*, 14(5), 2015.
15. Andrew Harris, Seymour Goodman, and Patrick Traynor. Privacy and Security Concerns Associated with Mobile Money Applications in Africa. *Washington Journal of Law, Technology & Arts*, 8(3), 2013.
16. Italo Dacosta, Saurabh Chakradeo, Mustaque Ahamad, and Patrick Traynor. One-Time Cookies: Preventing Session Hijacking Attacks with Stateless Authentication Tokens. *ACM Transactions on Internet Technology (TOIT)*, 12(1), 2012.
17. Cong Shi, Xiapu Luo, Patrick Traynor, Mostafa Ammar, and Ellen Zegura. ARDEN: Anonymous netwoRking in Delay tolErant Networks. *Journal of Ad Hoc Networks*, 10(6):918–930, 2012.
18. Patrick Traynor. Characterizing the Security Implications of Third-Party EAS Over Cellular Text Messaging Services. *IEEE Transactions on Mobile Computing (TMC)*, 11(6):983–994, 2012.
19. Italo Dacosta, Vijay Balasubramaniyan, Mustaque Ahamad, and Patrick Traynor. Improving Authentication Performance of Distributed SIP Proxies. *IEEE Transactions on Parallel and Distributed Systems (TPDS)*, 22(11):1804–1812, 2011.
20. Patrick Traynor, Chaitrali Amrutkar, Vikhyath Rao, Trent Jaeger, Patrick McDaniel, and Thomas La Porta. From Mobile Phones to Responsible Devices. *Journal of Security and Communication Networks (SCN)*, 4(6):719 – 726, 2011.
21. Matthew Pirretti, Patrick Traynor, Patrick McDaniel, and Brent Waters. Secure Attribute-Based Systems. *Journal of Computer Security (JCS)*, 18(5):799–837, 2010.
22. Patrick Traynor, Kevin Butler, William Enck, Kevin Borders, and Patrick McDaniel. malnets: Large-Scale Malicious Networks via Compromised Wireless Access Points. *Journal of Security and Communication Networks (SCN)*, 2(3):102–113, 2010.
23. Patrick Traynor. Securing Cellular Infrastructure: Challenges and Opportunities. *IEEE Security & Privacy Magazine*, 7(4), 2009.
24. Kevin Butler, Sunam Ryu, Patrick Traynor, and Patrick McDaniel. Leveraging Identity-based Cryptography for Node ID Assignment in Structured P2P Systems. *IEEE Transactions on Parallel and Distributed Systems (TPDS)*, 20(12):1803–1815, 2009.
25. Patrick Traynor, William Enck, Patrick McDaniel, and Thomas La Porta. Mitigating Attacks On Open Functionality in SMS-Capable Cellular Networks. *IEEE/ACM Transactions on Networking (TON)*, 17(1), 2009.
26. Patrick Traynor, Michael Chien, Scott Weaver, Boniface Hicks, and Patrick McDaniel. Non-Invasive Methods for Host Certification. *ACM Transactions on Information and System Security (TISSEC)*, 2008.
27. Patrick Traynor, William Enck, Patrick McDaniel, and Thomas La Porta. Exploiting Open Functionality in SMS-Capable Cellular Networks. *Journal of Computer Security (JCS)*, 16(6):713–742, 2008.
28. Patrick Traynor, Raju Kumar, Heesook Choi, Sencun Zhu, Guohong Cao, and Thomas La Porta. Efficient Hybrid Security Mechanisms for Heterogeneous Sensor Networks. *IEEE Transactions on Mobile Computing (TMC)*, 6(6), 2007.

C. Published Books and Parts of Books

1. Andrew Harris, Frank S. Park, Seymour Goodman, and Patrick Traynor. *Emerging Privacy and Security Concerns for Digital Wallet Deployment*. Privacy in America: Interdisciplinary Perspectives. Scarecrow Press, July 2011.
2. Kevin Butler, William Enck, Patrick Traynor, Jennifer Plasterr, and Patrick McDaniel. *Privacy Preserving Web-Based Email*. Algorithms, Architectures and Information Systems Security, Statistical Science and Interdisciplinary Research. World Scientific Computing, November 2008.
3. Patrick Traynor, Patrick McDaniel, and Thomas La Porta. *Security for Telecommunications Networks*. Number 978-0-387-72441-6 in Advances in Information Security Series. Springer, August 2008.

D. Edited Proceedings

None.

E. Conference Presentations

E.1. Conference Papers with Proceedings (Refereed)

1. Daniel Olszewski, Tyler Tucker, Kevin Butler, and Patrick Traynor. SoK: Towards a Unified Approach to Applied Replicability for Computer Security. In *Proceedings of the USENIX Security Symposium (Security)*, 2025.
2. Daniel Olszewski, Allison Lu, Anna Crowder, Nathaniel Bennett, Seth Layton, Sri Hrushikesh Varma Bhupathiraju, Tyler Tucker, Siddhant Kalgutkar, Hunter Ver Helst, Carson Stillman, Kevin Butler, Sara Rampazzi, and Patrick Traynor. "Raise Your Hand If You've Been Personally Victimized By A Lack Of Reproducibility": On Reproducibility in Tier 2 Security Conferences. In *Proceedings of ACM Conference on Reproducibility and Replicability (REP)*.
3. Cassidy Gibson and Daniel Olszewski and Natalie Grace Brigham and Anna Crowder and Kevin R. B. Butler and Patrick Traynor and Elissa M. Redmiles and Tadayoshi Kohno. Analyzing the AI Nudification Application Ecosystem. In *Proceedings of the USENIX Security Symposium (Security)*, 2025.
4. Tyler Tucker, Nathaniel Bennett, Martin Kotuliak, Simon Erni, Srdjan Capkun, Kevin Butler, and Patrick Traynor. Detecting IMSI-Catchers by Characterizing Identity Exposing Messages in Cellular Traffic. In *Symposium on Network and Distributed System Security (NDSS)*, 2025. (Acceptance Rate 16.1%).
5. Anna Crowder, Allison Lu, Kevin Childs, Carson Stillman, Patrick Traynor, and Kevin Butler. Data to Infinity and Beyond: Examining Data Sharing and Reuse Practices in the Computer Security Community. In *Proceedings of the IEEE Symposium on Security and Privacy (S&P)*, 2025.
6. Magdalena Pasternak, Kevin Warren, Daniel Olszewski, Susan Nittroueri, Patrick Traynor, and Kevin Butler. Characterizing the Impact of Audio Deepfakes in the Presence of Cochlear Implant Simulated Audio. In *Symposium on Network and Distributed System Security (NDSS)*, 2025. (Acceptance Rate 16.1%).
7. Anna Crowder, Daniel Olszewski, Patrick Traynor, and Kevin R. B. Butler. I Can Show You the World (of Censorship): Extracting Insights from Censorship Measurement Data Using Statistical Techniques. In *Proceedings of the Annual Computer Security Applications Conference (ACSAC)*, December 2024. (Acceptance Rate 21.8%).

8. Kevin Childs, Cassidy Gibson, Anna Crowder, Kevin Warren, Carson Stillman, Elissa Redmiles, Eakta Jain, Patrick Traynor, and Kevin Butler. "I Had Sort of a Sense that I Was Always Being Watched... Since I Was": Examining Interpersonal Discomfort From Continuous Location-Sharing Applications. In *Proceedings of the ACM Conference on Computer and Communications Security (CCS)*, 2024. (Acceptance Rate 16.9%).
9. Nathaniel Bennett, Weidong Zhu, Benjamin Simon, Ryon Kennedy, William Enck, Patrick Traynor, and Kevin Butler. RANsacked: A Domain-Informed Approach for Fuzzing LTE and 5G RAN-Core Interfaces. In *Proceedings of the ACM Conference on Computer and Communications Security (CCS)*, 2024. (Acceptance Rate 16.9%).
10. Kevin Warren, Tyler Tucker, Anna Crowder, Daniel Olszewski, Allison Lu, Caroline Fedele, Magdalena Pasternak, Seth Layton, Kevin Butler, Carrie Gates, and Patrick Traynor. Better Be Computer or I'm Dumb": A Large-Scale Evaluation of Humans as Audio Deepfake Detectors. In *Proceedings of the ACM Conference on Computer and Communications Security (CCS)*, 2024. (Acceptance Rate 16.9%).
11. K. Virgil English, Nathaniel Bennett, Seaver Thorn, Kevin Butler, William Enck, and Patrick Traynor. Examining Cryptography and Randomness Failures in Open-Source Cellular Cores. In *Proceedings of the ACM Conference on Data and Application Security and Privacy (CODASPY)*, 2024. (Acceptance rate: 21.3%)(Best Paper).
12. Seth Layton, Tyler Tucker, Daniel Olszewski, Kevin Warren, Carrie Gates, Kevin Butler, and Patrick Traynor. SoK: The Good, The Bad, and The Unbalanced: Measuring Structural Limitations of Current Deepfake Datasets. In *Proceedings of the USENIX Security Symposium (Security)*, 2024. (Acceptance Rate 18.3%).
13. Imani Munyaka, Daniel Delgado, Juan Gilbert, Jaime Ruiz, and Patrick Traynor. "I used to live in Florida": Exploring the Impact of Spam Call Warning Accuracy on Callee Decision-Making. In *Symposium on Usable Security and Privacy (USEC)*, 2024.
14. Jianliang Wu, Patrick Traynor, Dongyan Xu, Dave (Jing) Tian, and Antonio Bianchi. Finding Traceability Attacks in the Bluetooth Low Energy Specification and Its Implementations. In *Proceedings of the USENIX Security Symposium (Security)*, 2024. (Acceptance Rate 18.3%).
15. Daniel Olszewski, Allison Lu, Carson Stillman, Kevin Warren, Cole Kitroser, Alejandro Pascual, Divyajyoti Ukirde, Kevin Butler, and Patrick Traynor. "Get in Researchers; We're Measuring Reproducibility": A Reproducibility Study of Machine Learning Papers in Tier 1 Security Conferences. In *Proceedings of the ACM Conference on Computer and Communications Security (CCS)*, 2023. (Acceptance rate: 19.8%).
16. Christian Peeters, Tyler Tucker, Anushri Jain, Kevin Butler, and Patrick Traynor. LeopardSeal: Detecting Call Interception via Audio Rogue Base Stations. In *Proceedings of the ACM International Conference on Mobile Systems, Applications and Services (MobiSys)*, 2023. (Acceptance rate: 21%).
17. Tyler Tucker, Hunter Searle, Kevin Butler, and Patrick Traynor. Blue's Clues: Practical Discovery of Non-Discoverable Bluetooth Devices. In *Proceedings of the IEEE Symposium on Security and Privacy (S&P)*, 2023. (Acceptance rate: 14%).
18. Hadi Abdullah, Aditya Karlekar, Saurabh Prasad, Muhammad Sajidur Rahman, Logan Blue, Luke Bauer, Vincent Bindschaedler, and Patrick Traynor. Attacks as Defenses: Designing Robust Audio CAPTCHAs Using Attacks on Automatic Speech Recognition Systems. In *Symposium on Network and Distributed System Security (NDSS)*, 2023. (Acceptance rate: 16%).
19. Daniel Olszewski, Sandeep Sathyanarayana, Weidong Zhu, Kevin Butler, and Patrick Traynor. HallMonitor: A Framework for Identifying Network Policy Violations in Software. In *IEEE Conference on Communications and Network Security (CNS)*, 2022.

20. Hadi Abdullah, Aditya Karlekar, Vincent Bindschaedler, and Patrick Traynor. Demystifying Limited Adversarial Transferability in Automatic Speech Recognition Systems. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2022. (Acceptance rate: 32%).
21. Logan Blue, Kevin Warren, Hadi Abdullah, Cassidy Gibson, Luis Vargas, Jessica O'Dell, Kevin Butler, and Patrick Traynor. Who Are You (I Really Wanna Know)? Detecting Audio DeepFakes Through Vocal Tract Reconstruction. In *Proceedings of the USENIX Security Symposium (SECURITY)*, 2022. (Acceptance rate: 17.2%).
22. Grant Hernandez, Marius Muench, Dominik Maier, Alyssa Milburn, Shinjo Park, Tobias Scharnowski, Tyler Tucker, Patrick Traynor, and Kevin R. B. Butler. FirmWire: Transparent Dynamic Analysis for Cellular Baseband Firmware. In *Symposium on Network and Distributed System Security (NDSS)*, 2022. (Acceptance rate: 16.2%).
23. Christian Peeters, Christopher Patton, Imani N. Sherman, Daniel Olszewski, Thomas Shrimpton, and Patrick Traynor. SMS OTP Security (SOS): Hardening SMS-Based Two Factor Authentication. In *ACM Asia Conference on Computer and Communications Security (ASIACCS)*, 2022. (Acceptance rate: 18.2%).
24. Hadi Abdullah, Muhammad Sajidur Rahman, Christian Peeters, Cassidy Gibson, Washington Garcia, Vincent Bindschaedler, Thomas Shrimpton, and Patrick Traynor. Beyond L_p Clipping: Equalization based Psychoacoustic Attacks against ASRs. In *The Asian Conference on Machine Learning (ACML)*, 2021.
25. Imani Sherman and Daniel Delgado and Juan Gilbert and Jaime Ruiz and Patrick Traynor. Characterizing User Comprehension in the STIR/SHAKEN Anti-Robocall Standard. In *Proceedings of the Annual Research Conference on Communications Information and Internet Policy (TPRC 49)*, 2021.
26. Hadi Abdullah, Kevin Warren, Vincent Bindschaedler, Nicolas Papernot, and Patrick Traynor. The Faults in our ASRs: An Overview of Attacks against Automatic Speech Recognition and Speaker Identification Systems. In *Proceedings of the IEEE Symposium on Security and Privacy (S&P)*, 2021. (Acceptance rate: 12.1%).
27. Hadi Abdullah, Muhammad Sajidur Rahman, Washington Garcia, Logan Blue, Kevin Warren, Anurag Swarnim Yadav, Tom Shrimpton, and Patrick Traynor. Hear “No Evil”, See “Kenansville”: Efficient and Transferable Black-Box Attacks on Speech Recognition and Voice Identification Systems. In *Proceedings of the IEEE Symposium on Security and Privacy (S&P)*, 2021. (Acceptance rate: 12.1%).
28. Imani Sherman, Jasmine Bowers, Liz-Laure Laborde, Juan E. Gilbert, Jaime Ruiz, and Patrick Traynor. Truly Visual Caller ID? An Analysis of Anti-Robocall Applications and their Accessibility to Visually Impaired Users. In *IEEE International Symposium on Technology and Society (IEEE ISTAS)*, 2020.
29. Imani Sherman, Jasmine Bowers, Keith McNamara, Juan Gilbert, Jaime Ruiz, and Patrick Traynor. Are You Going to Answer That? Measuring User Responses to Anti-Robocall Application Indicators. In *Proceedings of the ISOC Network & Distributed Systems Security Symposium (NDSS)*, 2020. (Acceptance rate: 17.4%).
30. Joseph Choi, Dave Tian, Grant Hernandez, Christopher Patton, Benjamin Mood, Thomas Shrimpton, Patrick Traynor, and Kevin Butler. A Hybrid Approach to Secure Function Evaluation Using SGX. In *Proceedings of the ACM ASIA Conference on Computer and Communications Security (ASIACCS'19)*, 2019. (Acceptance Rate: 17.0% for full papers).
31. Vanessa Frost, Dave Tian, Christie Ruales, Patrick Traynor, and Kevin Butler. Examining DES-based Cipher Suite Support within the TLS Ecosystem. In *Proceedings of the ACM ASIA Conference*

on *Computer and Communications Security (ASIACCS'19)*, 2019. (Acceptance Rate: 22.0% for all papers).

32. Dave Tian, Joseph Choi, Grant Hernandez, Patrick Traynor, and Kevin Butler. A Practical Intel SGX Setting for Linux Containers in the Cloud. In *Proceedings of the ACM Conference on Data and Application Security and Privacy (CODASPY'19)*, 2019. (Acceptance rate: 23.5%).
33. Nolen Scaife, Jasmine Bowers, Christian Peeters, Grant Hernandez, Imani Sherman, Lisa Anthony, and Patrick Traynor. Kiss from a Rogue: Evaluating Detectability of Pay-at-the-Pump Card Skimmers. In *Proceedings of the IEEE Symposium on Security and Privacy (S&P)*, 2019. (Acceptance rate: 12.0%).
34. Jasmine Bowers, Imani Sherman, Kevin Butler, and Patrick Traynor. Characterizing Security and Privacy Practices in Emerging Digital Credit Applications. In *Proceedings of the ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec)*, 2019. (Acceptance rate: 25.6%).
35. Hadi Abdullah, Washington Garcia, Christian Peeters, P. Traynor, K. Butler, and J. Wilson. Practical Hidden Voice Attacks against Speech and Speaker Recognition Systems. In *Proceedings of the ISOC Network & Distributed Systems Security Symposium (NDSS)*, 2019. (Acceptance Rate: 17.1%).
36. Lius Vargas, Logan Blue, Vanessa Frost, Christopher Patton, N. Scaife, K. Butler, and P. Traynor. Digital Healthcare-Associated Infection Analysis of a Major Multi-Campus Hospital System. In *Proceedings of the ISOC Network & Distributed Systems Security Symposium (NDSS)*, 2019. (Acceptance Rate: 17.1%).
37. Dominik Wermke, Nicolas Huaman, Yasemin Acar, Bradley Reaves, Patrick Traynor, and Sascha Fahl. A Large Scale Investigation of Obfuscation Use in Google Play. In *Proceedings of the Annual Computer Security Applications Conference (ACSAC)*, 2018. Acceptance Rate: 20.1%.
38. Nolen Scaife, Christian Peeters, and Patrick Traynor. Fear the Reaper: Characterization and Fast Detection of Card Skimmers. In *Proceedings of the USENIX Security Symposium (SECURITY)*, 2018. Acceptance Rate: 19.0%.
39. Dave (Jing) Tian, Grant Hernandez, Joseph Choi, Vanessa Frost, Christie Raules, Kevin Butler, Patrick Traynor, Hayawardh Vijayakumar, Lee Harrison, Amir Rahmati, and Mike Grace. Attention Spanned: Comprehensive Vulnerability Analysis of AT Commands Within the Android Ecosystem. In *Proceedings of the USENIX Security Symposium (SECURITY)*, 2018. Acceptance Rate: 19.0%.
40. Luis Vargas, Gyan Hazarika, Rachel Culpepper, Kevin Butler, Thomas Shrimpton, Doug Szajda, and Patrick Traynor. Mitigating Risk while Complying with Data Retention Laws. In *Proceedings of the ACM Conference on Computer and Communications Security (CCS)*, 2018.
41. Logan Blue, Luis Vargas, and Patrick Traynor. Hello, Is It Me You're Looking For? Differentiating Between Human and Electronic Speakers for Voice Interface Security. In *Proceedings of the ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec)*, 2018.
42. Logan Blue, Hadi Abdullah, Luis Vargas, and Patrick Traynor. 2MA: Verifying Voice Commands via Two Microphone Authentication. In *ACM Asia Conference on Computer and Communications Security (ASIACCS)*, 2018. (Acceptance Rate: 20.0%).
43. Nolen Scaife, Christian Peeters, Camilo Velez, Hanqing Zhao, Patrick Traynor, and David Arnold. The Cards Aren't Alright: Detecting Counterfeit Gift Cards Using Encoding Jitter. In *Proceedings of the IEEE Symposium on Security and Privacy (S&P)*, 2018. (Acceptance Rate: 10.4%).

44. Christian Peeters, Hadi Abdullah, Nolen Scaife, Jasmine Bowers, Patrick Traynor, Bradley Reaves, and Kevin Butler. Sonar: Detecting SS7 Redirection Attacks Via Call Audio-Based Distance Bounding. In *Proceedings of the IEEE Symposium on Security and Privacy (S&P)*, 2018. (Acceptance Rate: 10.4%).
45. Tyler Ward, Joseph Choi, Kevin Butler, John M. Shea, Patrick Traynor, and Tan Wong. Privacy Preserving Localization Using a Distributed Particle Filtering Protocol. In *IEEE MILCOM*, 2017. (Acceptance Rate: 56%).
46. Bradley Reaves and Logan Blue and Hadi Abdullah and Luis Vargas and Patrick Traynor and Thomas Shrimpton. AuthentiCall: Efficient Identity and Content Authentication for Phone Calls. In *Proceedings of the USENIX Security Symposium (SECURITY)*, 2017. (Acceptance Rate: 16.3%).
47. Jasmine Bowers and Bradley Reaves and Imani N. Sherman and Patrick Traynor and Kevin Butler. Regulators, Mount Up! Analysis of Privacy Policies for Mobile Money Applications. In *Proceedings of the USENIX Symposium on Usable Privacy and Security (SOUPS)*, 2017. (Acceptance Rate: 26.5%).
48. Bradley Reaves, Logan Blue, and Patrick Traynor. AuthLoop: End-to-End Cryptographic Authentication for Telephony over Voice Channels. In *Proceedings of the USENIX Security Symposium (SECURITY)*, 2016. (Acceptance Rate: 15.5%).
49. Dave Tian, Nolen Scaife, Adam Bates, Kevin Butler, and Patrick Traynor. Making USB Great Again with USBFILTER. In *Proceedings of the USENIX Security Symposium (SECURITY)*, 2016. (Acceptance Rate: 15.5%).
50. Bradley Reaves, Dave Tian, Logan Blue, Patrick Traynor, and Kevin Butler. Detecting SMS Spam in the Age of Legitimate Bulk Messaging. In *Proceedings of the ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec)*, 2016. (Acceptance Rate: 35.0%).
51. Nolen Scaife, Henry Carter, Patrick Traynor, and Kevin Butler. CryptoLock (and Drop It): Stopping Ransomware Attacks on User Data. In *IEEE International Conference on Distributed Computing Systems (ICDCS)*, 2016. (Acceptance Rate: 17.6%).
52. Bradley Reaves, Nolen Scaife, Dave Tian, Logan Blue, Patrick Traynor, and Kevin Butler. Sending out an SMS: Characterizing the Security of the SMS Ecosystem with Public Gateways. In *Proceedings of the IEEE Symposium on Security and Privacy (S&P)*, 2016. (Acceptance Rate: 13.0%).
53. Benjamin Mood, Debayan Gupta, Henry Carter, Kevin Butler, and Patrick Traynor. Frigate: A Validated, Extensible, and Efficient Compiler and Interpreter for Secure Computation. In *Proceedings of the IEEE European Symposium on Security and Privacy*, 2016. (Acceptance Rate: 17.3%).
54. Henry Carter, Benjamin Mood, Patrick Traynor, and Kevin Butler. Outsourcing Secure Two-Party Computation as a Black Box. In *Proceedings of the International Conference on Cryptology and Network Security*, 2015. (Acceptance Rate: 52.9%).
55. Nolen Scaife, Henry Carter, and Patrick Traynor. OnionDNS: A Seizure-Resistant Top-level Domain. In *Proceedings of the IEEE Conference on Communications and Network Security (CNS)*, 2015. (Acceptance Rate: 28.1%).
56. Bradley Reaves, Nolen Scaife, Adam Bates, Patrick Traynor, and Kevin Butler. Mo(bile) Money, Mo(bile) Problems: Analysis of Branchless Banking Applications in the Developing World. In *Proceedings of the USENIX Security Symposium (SECURITY)*, 2015. (Acceptance Rate: 15.7%).
57. Bradley Reaves, Ethan Shernan, Adam Bates, Henry Carter, and Patrick Traynor. Boxed Out: Blocking Cellular Interconnect Bypass Fraud at the Network Edge. In *Proceedings of the USENIX Security Symposium (SECURITY)*, 2015. (Acceptance Rate: 15.7%).

58. David Dewey, Bradley Reaves, and Patrick Traynor. Uncovering Use-After-Free Conditions In Compiled Code. In *Proceedings of the International Conference on Availability, Reliability and Security (ARES)*, 2015. (Acceptance Rate: 22%).
59. Ethan Sherman, Henry Carter, Dave Tian, Patrick Traynor, and Kevin Butler. More Guidelines Than Rules: CSRF Vulnerabilities from Noncompliant OAuth 2.0 Implementations. In *Proceedings of the International Conference on Detection of Intrusions & Malware, and Vulnerability Assessment (DIMVA)*, 2015. (Acceptance Rate: 22.7%).
60. Henry Carter, Charles Lever, and Patrick Traynor. Whitewash: Outsourcing Garbled Circuit Generation for Mobile Devices. In *Proceedings of the Annual Computer Security Applications Conference (ACSAC)*, 2014. (Acceptance Rate: 19.9%).
61. Henry Carter, Benjamin Mood, Patrick Traynor, and Kevin Butler. Secure Outsourced Garbled Circuit Evaluation for Mobile Devices. In *Proceedings of the USENIX Security Symposium (SECURITY)*, 2013. (Acceptance Rate: 16.2%).
62. Chaitrali Amrutkar, Matti Hiltunen, Shobha Venkataraman, Kaustubh Joshi, Patrick Traynor, Trevor Jim, and Oliver Spatscheck. Why is My Smartphone Slow? On The Fly Diagnosis of Poor Performance on the Mobile Internet. In *Proceedings of The 43rd Annual IEEE/IFIP International Conference on Dependable Systems and Networks*, 2013. (Acceptance Rate: 19.6%).
63. Saurabh Chakradeo, Brad Reaves, Patrick Traynor, and William Enck. MAST: Triage for Market-scale Mobile Malware Analysis. In *Proceedings of the ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec)*, 2013. (Acceptance Rate: 15.0%)(Best Paper).
64. Charles Lever, Manos Antonakakis, Brad Reaves, Patrick Traynor, and Wenke Lee. The Core of the Matter: Analyzing Malicious Traffic in Cellular Carriers. In *Proceedings of the ISOC Network & Distributed System Security Symposium (NDSS)*, 2013. (Acceptance rate: 18.8%).
65. Chaitrali Amrutkar, Kapil Singh, Arunabh Verma, and Patrick Traynor. VulnerableMe: Measuring Systemic Weaknesses in Mobile Browser Security. In *Proceedings of the International Conference on Information Systems Security (ICISS)*, 2012. (Acceptance rate: 25%) (Best Paper - SAIC Student Paper Competition (GT)) (Finalist - CSAW AT&T Applied Security Research Best Paper Competition 2012).
66. Chaitrali Amrutkar, Patrick Traynor, and Paul van Oorschot. A Measurement Study of SSL Indicators on Mobile Browsers: Extended Life, or End of the Road? In *Proceedings of the Information Security Conference (ISC)*, 2012. (Acceptance rate: 32%) (Best Student Paper).
67. Italo Dacosta, Mustaque Ahamad, and Patrick Traynor. Trust No One Else: Detecting MITM Attacks Against SSL/TLS Without Third-Parties. In *Proceedings of the European Symposium on Research in Computer Security (ESORICS)*, 2012. (Acceptance Rate: 20.2%).
68. Adam Bates, Kevin Butler, Micah Sherr, Clay Shields, Patrick Traynor, and Dan Wallach. Accountable Wiretapping -or- I Know They Can Hear You Now. In *Proceedings of the ISOC Network & Distributed System Security Symposium (NDSS)*, 2012. (Acceptance Rate: 17.8%).
69. Yacin Nadji, Jon Giffin, and Patrick Traynor. Automated Remote Repair for Mobile Malware. In *Proceedings of the Annual Computer Security Applications Conference (ACSAC)*, 2011. (Acceptance Rate: 18.5%).
70. Nilesh Nipane, Italo Dacosta, and Patrick Traynor. "Mix-In-Place" Anonymous Networking Using Secure Function Evaluation. In *Proceedings of the Annual Computer Security Applications Conference (ACSAC)*, 2011. (Acceptance Rate: 18.5%).

85. Sunam Ryu, Kevin Butler, Patrick Traynor, and Patrick McDaniel. Leveraging Identity-based Cryptography for Node ID Assignment in Structured P2P Systems. In *Proceedings of the IEEE International Symposium on Security in Networks and Distributed Systems (SSNDS)*, 2007. (Acceptance Rate: 40%).
86. Luke St. Clair, Lisa Johansen, William Enck, Matthew Pirretti, Patrick Traynor, Patrick McDaniel, and Trent Jaeger. Password Exhaustion: Predicting the End of Password Usefulness. In *Proceedings of the International Conference on Information Systems Security (ICISS)*, 2006. (Invited Paper).
87. Kevin Butler, William Enck, Jennifer Plasterr, Patrick Traynor, and P. McDaniel. Privacy-Preserving Web-Based Email. In *Proceedings of the International Conference on Information Systems Security (ICISS)*, December 2006. (Acceptance Rate: 30.4%).
88. Matthew Pirretti, Patrick Traynor, Patrick McDaniel, and Brent Waters. Secure Attribute-Based Systems. In *Proceedings of the Thirteenth ACM Conference on Computer and Communications Security (CCS)*, November 2006. (Acceptance Rate: 14.8%).
89. Patrick Traynor, William Enck, Patrick McDaniel, and Thomas La Porta. Mitigating Attacks on Open Functionality in SMS-Capable Cellular Networks. In *Proceedings of the Twelfth Annual ACM International Conference on Mobile Computing and Networking (MobiCom)*, September 2006. (Acceptance Rate: 11.7%).
90. Patrick Traynor, Michael Chien, Scott Weaver, Boniface Hicks, and Patrick McDaniel. Non-Invasive Methods for Host Certification. In *Proceedings of the Second IEEE International Conference on Security and Privacy in Communication Networks (SecureComm)*, August 2006. (Acceptance Rate: 25.4%).
91. Patrick Traynor, JaeShung Shin, Barat Madan, Shashi Phoha, and Thomas La Porta. Efficient Group Mobility for Heterogeneous Sensor Networks. In *Proceedings of the IEEE Vehicular Technology Conference (VTC Fall)*, September 2006. (Acceptance Rate: 58%).
92. Patrick Traynor, Raju Kumar, Hussain Bin Saad, Guohong Cao, and Thomas La Porta. LIGER: Implementing Efficient Hybrid Security Mechanisms for Heterogeneous Sensor Networks. In *Proceedings of the 4th ACM International Conference on Mobile Systems, Applications and Services (MobiSys)*, June 2006. (Acceptance Rate: 15.4%).
93. Patrick Traynor, Guohong Cao, and Thomas La Porta. The Effects of Probabilistic Key Management on Secure Routing in Sensor Networks. In *Proceedings of the 2006 IEEE Wireless Communications and Networking Conference (WCNC)*, April 2006. (Acceptance Rate: 38.8%).
94. Patrick Traynor, Heesook Choi, Guohong Cao, Sencun Zhu, and Thomas La Porta. Establishing Pair-Wise Keys In Heterogeneous Sensor Networks. In *Proceedings of the 25th Annual IEEE Conference on Computer Communications (INFOCOM)*, April 2006. (Acceptance Rate: 18%).
95. William Enck, Patrick Traynor, Patrick McDaniel, and Thomas La Porta. Exploiting Open Functionality in SMS-Capable Cellular Networks. In *Proceedings of the Twelfth ACM Conference on Computer and Communications Security (CCS)*, November 2005. (Acceptance Rate: 15%).
96. Patrick Traynor. Work in Progress Presentations: Fine-Grained Secure Localization for 802.11 Networks. 15th USENIX Security Symposium (SECURITY), August 2006.
97. Patrick Traynor. Work in Progress Presentations: Fundamental Limitations of Sensor Network Security. ACM/USENIX Fourth International Conference on Mobile Systems Applications and Services (MobiSys), June 2006. (Award: Most Entertaining WIP).
98. Patrick Traynor, Heesook Choi, Guohong Cao, and Thomas La Porta. Poster Session: Probabilistic Unbalanced Key Distribution and Its Effects on Distributed Sensor Networks. Workshop on Wireless Security (WiSe), October 2004.

Removed for external version.

E.2. Conference Presentations with Proceedings (Non-Refereed)

None.

E.3. Conference Presentations without Proceedings

1. Patrick Traynor. Characterizing the Limitations of Third-Party EAS Over Cellular Text Messaging Services. Technical report, 3G Americas Whitepaper, 2008.
2. Lisa Johansen, Kevin Butler, William Enck, Patrick Traynor, and Patrick McDaniel. Grains of SANs: Building Storage Area Networks from Memory Spots. Technical Report NAS-TR-0060-2006, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, 2007.
3. Luis Vargas, Patrick Emami, and Patrick Traynor. On the Detection of Disinformation Campaign Activity with Network Analysis. In *Proceedings of the 2020 ACM SIGSAC Cloud Computing Security Workshop, CCSW '20*, 2020.

F. Other

F.1. Submitted Journal Papers

None.

F.2. Refereed Research Reports

None.

F.3. Software

1. *GSM Air Interface Simulator*: Developed a full voice, data and SMS capable simulator for the wireless portion of a GSM network. Models communications down to the timeslot for highest possible accuracy. Used in the majority of our work on cellular security.
2. *Malicious Telephony Load Tester*: Built a system on top of the TM1 Telecom Database testing suite to allow for a comparison of malicious traffic of varying composition.

F.4. Published Papers (Non-Refereed)

1. Siddhant Deshmukh, Henry Carter, Grant Hernandez, Patrick Traynor, and Kevin Butler. Efficient and Secure Template Blinding for Biometric Authentication. In *IEEE Workshop on Security and Privacy in the Cloud (SPC)*, 2016.
2. Debayan Gupta, Benjamin Mood, Joan Feigenbaum, Kevin Butler, and Patrick Traynor. Using Intel Software Guard Extensions for Efficient Two-Party Secure Function Evaluation. In *Workshop on Encrypted Computing and Applied Homomorphic Cryptography (WAHC)*, 2016.

F.5. Books in Preparation

None.

F.6. Workshops and External Courses

1. Chaitrali Amrutkar and Patrick Traynor. Rethinking Permissions for Mobile Web Apps: Barriers and the Road Ahead. In *Proceedings of the ACM CCS Workshop on Security and Privacy in Smartphones and Mobile Devices (SPSM)*, 2012.
2. Nigel Lawrence and Patrick Traynor. Under New Management: Practical Attacks on SNMPv3. In *Proceedings of the USENIX Workshop on Offensive Technologies (WOOT)*, 2012.
3. Andrew Harris, Frank S. Park, Seymour Goodman, and Patrick Traynor. Emerging Privacy Concerns for Digital Wallet Deployment. In *Proceedings of the Workshop on Making Privacy in America*, 2009.
4. Patrick Traynor. Privacy and Security Concerns for Personal and Mobile Health Devices. In *Proceedings of the Workshop to Set A Research Agenda for Privacy and Security of Healthcare Technologies*, 2009.
5. Kevin Butler, William Enck, Harri Hursti, Stephen McLaughlin, Patrick Traynor, and Patrick McDaniel. Systemic Issues in the Hart InterCivic and Premier Voting System: Reflections Following Project EVEREST. In *Proceedings of the USENIX/ACCURATE Electronic Voting Technology (EVT) Workshop*, 2008.
6. Keynote: Well, It Worked on My Computer: Reproducibility in Computer Security Research. Learning from Authoritative Security Experiment Results (LASER) Workshop, December 2024. École Polytechnique Fédérale de Lausanne (EPFL, Switzerland).
7. Social Engineering and Two-Factor Authentication. Cyber Security Training Eastern Indonesia, August 2024. Financial Innovation Lab (EIFIL) (Denpasar, Bali, Indonesia).
8. DFS Security and Mobile Money Analysis. Cyber Security Training Eastern Indonesia, August 2024. Financial Innovation Lab (EIFIL) (Denpasar, Bali, Indonesia).

G. Research Proposals and Grants (Principal Investigator)

1. Approved and Funded

1. **Artus Protocol STTR Phase II - Extension**
Sponsor: Office of Naval Research
Investigator(s): Patrick Traynor (PI), Kevin Butler
Amount: \$300,000 over 2 years
Awarded: August 2023
2. **Testing Audio Deep Fake Detectors**
Sponsor: Bank of America
Investigator(s): Patrick Traynor (PI), Kevin Butler
Amount: \$150,000 over 1 year
Awarded: August 2023
3. **Testing Audio Deep Fake Detectors**
Sponsor: Bank of America
Investigator(s): Patrick Traynor (PI), Kevin Butler
Amount: \$274,000 over 2 years
Awarded: August 2021
4. **Deploying Defenses for Cellular Networks Using the AWARE Testbed**
Sponsor: Department of Homeland Security: CISA:
Investigator(s): Patrick Traynor (PI), Kevin Butler, Guofei Gu, Radu Stoleru, Walter Magnussen, P.

R. Kumar

Amount: \$3,100,000 over 4 years

Awarded: October 2019

5. **SaTC:CORE:Medium: Securing the Voice Processing Pipeline Against Adversarial Audio**
Sponsor: NSF Secure and Trustworthy Cyberspace
Investigator(s): Patrick Traynor (PI), Thomas Shrimpton, Vincent Bindschaedler
Amount: \$1,199,999 over 4 years
Awarded: October 2019
6. **Artus Protocol STTR Phase II**
Sponsor: Office of Naval Research
Investigator(s): Patrick Traynor (PI), Kevin Butler
Amount: \$800,000 over 4 years
Awarded: August 2019
7. **Evaluating the Security of QR Code-Based Payments**
Sponsor: Discover Financial
Investigator(s): Patrick Traynor (PI)
Amount: \$50,000 over 1 year
Awarded: September 2018
8. **Workshop: Addressing the Technical Security Challenges of Emerging Digital Financial Services**
Sponsor: NSF Secure and Trustworthy Cyberspace
Investigator(s): Patrick Traynor (PI), Kevin Butler
Amount: \$50,000 over 1 year
Awarded: September 2017
9. **Designing Strong End-to-End Authentication Mechanisms for Modern Telephony Systems**
Sponsor: NSF Secure and Trustworthy Cyberspace
Investigator(s): Patrick Traynor (PI)
Amount: \$500,000 over 3 years
Awarded: July 2016
10. **Digital Healthcare-Associated Infection: Measurement, Defense and Prevention in a Modern Digital Healthcare Ecosystem**
Sponsor: National Science Foundation
Investigator(s): Patrick Traynor (PI), Kevin Butler, Shigang Chen
Amount: \$1,200,000 over 4 years
Awarded: June 2016
11. **Evaluating and Improving Security in Emerging Branchless Banking Systems**
Sponsor: NSF Secure and Trustworthy Cyberspace
Investigator(s): Patrick Traynor (PI)
Amount: \$500,000 over 3 years
Awarded July 2015
12. **Prevention and Detection of Disallowed Connections in Mobile and Pervasive Systems**
Sponsor: CISE-ECE Harris Endowed Seed Fund Program
Investigator(s): Patrick Traynor (PI), Renato Figueiredo (PI)
Amount: \$40,000 over 1 year
Awarded December 2014
13. **Mobile Excursion Study Support**
Sponsor: Hanscom AFB Electronic Systems Command Development Planning Division (ESC/XR)

Investigator(s): Patrick Traynor (PI), Mustaque Ahamad, Jeff Evans, Chuck Bokath
Amount: \$280,000 over 3 months
Awarded July 2012

14. **Characterizing the Security Limitations of Accessing the Mobile Web**
Sponsor: NSF Secure and Trustworthy Cyberspace
Investigator(s): Patrick Traynor (PI) and William Enck (NC State)
Amount: \$334,000 over 3 years
Awarded July 2012
15. **Mitigating Attacks on Mobile Devices and Critical Cellular Infrastructure**
Sponsor: US Department of Defense - Defense University Research Instrumentation Program (DURIP)
Investigator(s): Patrick Traynor (PI), Jon Giffin, Mustaque Ahamad
Amount: \$210,081 over 1 year
Awarded June 2011
16. **Characterizing and Implementing Efficient Primitives for Privacy-Preserving Computation**
Sponsor: DARPA PROgramming Computation on EncryptEd Data (PROCEED) – Broad Agency Announcement
Investigator(s): Patrick Traynor (PI) and Kevin Butler (UOregon)
Amount: \$580,000 over 4 years
Awarded May 2011
17. **Security for Converged IMS Networks**
Sponsor: US Department of Defense
Investigator(s): Patrick Traynor (PI), Mustaque Ahamad and Russ Clark
Amount: \$242,401 over 1 year
Awarded August 2010
18. **CAREER: Protecting User Data on Lost, Stolen and Damaged Mobile Phones**
Sponsor: NSF Trustworthy Computing
Investigator(s): Patrick Traynor (PI)
Amount: \$400,000 over 5 years
Awarded: May 2010
19. **Provably Anonymous Networking Through Secure Function Evaluation**
Sponsor: NSF Trustworthy Computing
Investigator(s): Patrick Traynor (PI)
Amount: \$200,000 over 2 years
Awarded: July 2009
20. **Characterizing and Mitigating Device-Based Attacks in Cellular Telecommunications Networks**
Sponsor: NSF Trustworthy Computing
Investigator(s): Patrick Traynor (PI) and Jonathon Giffin
Amount: \$450,000 over 3 years
Awarded: July 2009

2. Pending

Removed for external version.

H. Research Proposals and Grants (Contributor)

1. Approved and Funded

- 1. SaTC: Frontier: Securing the Future of Computing for Marginalized and Vulnerable Populations**
Sponsor: NSF SaTC
Investigator(s): Kevin Butler (PI), Patrick Traynor, Tadayoshi Kohno, Franzi Roesner, Apu Kapadia, Eakta Jain.
Amount: \$7,500,000 for 5 years
Awarded October 2022
- 2. ROCKY: Reliable Obfuscated Communications Kit for everYone**
Sponsor: DARPA Resilient Anonymous Communication for Everyone (RACE) – Broad Agency Announcement
Investigator(s): Thomas Shrimpton (PI), Patrick Traynor, Kevin Butler, Vincent Bindschaedler, Nadia Heninger
Amount: \$1,600,000 over 4 years
Awarded May 2019
- 3. WiFiUS: Collaborative Research: SELIOT: Securing Lifecycle of Internet-of-Things**
Sponsor: NSF CNS WiFiUS
Investigator(s): Gene Tsudik (PI), Patrick Traynor
Amount: \$300,000 for 2 years
Submitted December 2016
- 4. Cloud-based Oblivious Spectrum Mapping and Allocation**
Sponsor: NSF CNS EARS
Investigator(s): John Shea (PI), Tan Wong, Patrick Traynor
Amount: \$532,952 for 2 years
Submitted May 2016
- 5. DURIP: Developing Research Capability in Cyber-Physical Systems at the University of Florida**
Sponsor: Small
Investigator(s): Kevin Butler (PI), Patrick Traynor, My Thai
Amount: \$200,000 for 2 years
Submitted: June 2015
- 6. Securing the New Converged Telephony Landscape**
Sponsor: NSF TWC: Small
Investigator(s): Mustaque Ahamad (PI) and Patrick Traynor
Amount: \$500,000 for 3 years
Submitted: December 2012
- 7. Facilitating Free and Open Access to Information on the Internet**
Sponsor: NSF Trustworthy Computing
Investigator(s): Nick Feamster (PI), Wenke Lee, Patrick Traynor, Hans Klein, Roger Dingledine, Michael Freedman and Edward W. Felten
Amount: \$1,500,000 for 4 years
Awarded: June 2011
- 8. Monitoring Free and Open Access to Information on the Internet**
Sponsor: Google Focus Program
Investigator(s): Nick Feamster (PI), Wenke Lee, Mustaque Ahamad, Patrick Traynor, Henry Owen, Ellen Zegura, Zvi Galil

Amount: \$1,000,000 for 2 years
Awarded: November 2011

9. **Dynamic-attribute-based Disclosure of Health Information in Emergency Care Scenarios**
Sponsor: Health Systems Institute (HSI) Seed Grant Program
Investigator(s): Doug Blough (PI), Mustaque Ahamad, Patrick Traynor and Jim Jose
Amount: \$50,000 over 1 year
Awarded: August 2009
10. **Federal Cyber Service Scholarships at Georgia Tech**
Sponsor: NSF SFS Scholarships
Investigator(s): Seymour Goodman (PI), Patrick Traynor
Amount: \$1,250,682 over 5 years
Awarded: June 2009
11. **Security for IMS-Enabled Converged Applications**
Sponsor: US Department of Defense
Investigator(s): Mustaque Ahamad (PI), Patrick Traynor (PI), Michael Hunter, Russ Clark
Amount: \$146,121 for 1 year
Awarded: August 2008

2. Pending

Removed for external version.

I. Research Honors and Awards

1. Fellow, Center for Financial Inclusion at Accion, 2017.
2. Sloan Research Fellow, Alfred P. Sloan Foundation, 2014.
3. Best Paper, The ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec); Budapest, Hungary, 2013.
4. Best Student Paper, The Information Security Conference (ISC); Passau, Germany, 2012
5. Lockheed Inspirational Young Faculty Award, 2012
6. Best Demo, "Is Browsing the Internet on Your Mobile Phone Secure?" Chaitrali Amrutkar (Ph.D Advisee), CoC Research Day, 2011
7. Best Poster, "(sp)iPhone: Decoding Vibrations From Nearby Keyboards Using Mobile Phone Accelerometers" Arunabh Verma, Henry Carter (MS, Ph.D Advisees), CoC Research Day, 2011
8. National Science Foundation CAREER Award, 2010
9. Pennsylvania State University Alumni Association Dissertation Award, 2007
10. Pennsylvania State University CSE Graduate Research Assistant Award, 2007
11. AT&T Wireless Fellowship, 2005

III. SERVICE

A. Professional Activities

A.1. Memberships and Activities in Professional Societies

1. Senior Member, Association for Computing Machinery (ACM)
2. Senior Member, Institute of Electrical and Electronics Engineers (IEEE)
3. Member, USENIX Advanced Computing Systems Association (USENIX)

A.2. Conference Committee Activities

1. Program co-Chair, *IEEE Symposium on Security and Privacy (OAKLAND)*: 2023, 2024
2. Program co-Chair, *USENIX Security Symposium (SECURITY)*: 2019
3. Program co-Chair, *Network and Distributed System Security Symposium (NDSS)*: 2017, 2018
4. Program Chair, *USENIX Workshop on Offensive Technologies (WOOT)*: 2016
5. Program Chair, *ACM Conference on Wireless Network Security (WiSec)*: 2014
6. Program Co-Chair, *Annual Computer Security Applications Conference (ACSAC)*: 2012, 2013
7. Program Chair, *USENIX Workshop on Hot Topics in Security (HotSec)*: 2012
8. Chair Invited Talks Committee, *USENIX Security Symposium (SECURITY)*: 2014
9. Workshops Chair, *IEEE Conference on Communications and Network Security (CNS)*: 2016
10. Program Committee, *USENIX Security Symposium (SECURITY)*: 2008, 2009, 2010, 2013, 2015-2018, 2020-2022
11. Program Committee, *IEEE Symposium on Security and Privacy (OAKLAND)*: 2009-2014, 2022.
12. Program Committee, *ACM Conference On Computer and Communications Security (CCS)*: 2009, 2013-2015, 2017
13. Program Committee, *Network and Distributed System Security Symposium (NDSS)*: 2010, 2013-2016, 2020-2021
14. Program Committee, *IEEE European Symposium on Security and Privacy (Euro S&P)*: 2016
15. Program Committee, *Annual Computer Security Applications Conference (ACSAC)*: 2008, 2009, 2010, 2011, 2015
16. Program Committee, *ACM Conference on Wireless Network Security (WiSec)*: 2009, 2010, 2013, 2015-2021
17. Program Committee, *International Conference on Financial Cryptography and Data Security (FC)*: 2010, 2013
18. Program Committee, *USENIX Symposium on Operating Systems Design and Implementation (OSDI)*: 2016.
19. Program Committee, *ICST Conference on Security and Privacy in Communication Networks (SecureComm)*: 2009, 2010
20. Program Committee, *Privacy Enhancing Technologies Symposium (PETS)*: 2015, 2016

21. Program Committee, *International World Wide Web Conference (WWW)*: 2016
22. Program Committee, *USENIX Workshop on Hot Topics in Security (HotSec)*: 2011
23. Program Committee, *ACM SIGCOMM Workshop on Networking, Systems, and Applications on Mobile Handhelds (MOBIHELD)*: 2010
24. Program Committee, *International Workshop on Mobile Security (WMS)*: 2010
25. Program Committee, *European Symposium on Research in Computer Security (ESORICS)*: 2009, 2011
26. Program Committee, *IEEE Conference on Mobile Ad-hoc and Sensor Systems (MASS)*: 2009, 2010
27. Program Committee, *Information Security Conference (ISC)*: 2010
28. Program Committee, *IEEE International Conference on Information Privacy, Security, Risk and Trust (PASSAT)*: 2009
29. Program Committee, *Computer Security Architecture Workshop (CSAW)*: 2008
30. Program Committee, *IWCMC Computer and Network Security Symposium*: 2009
31. Program Committee, *IARIA International Conference on Internet Monitoring and Protection (ICIMP)*: 2009
32. Program Committee, *IEEE Workshop on Network Security and Privacy (NSP)*: 2008
33. Program Committee, *IEEE International Workshop on Wireless and Sensor Networks Security (WSNS)*: 2008, 2009
34. Program Committee, *IEEE Conference on Sensor Networks, Ubiquitous and Trustworthy Computing (SUTC)*: 2008
35. Program Committee, *ACM Conference on Computer and Communications Security, Industry and Government Track (CCS I&G)*: 2006, 2007
36. Program Committee, *Workshop on Secure Network Protocols (NPSec)*: 2006
37. Program Committee, *International Conference on Information Systems Security (ICISS)*: 2006, 2009, 2010
38. Program Committee, *IEEE LCN Workshop on Network Security (WNS)*: 2006, 2007, 2008

B. On-Campus Committees

B.1. University of Florida

1. Member, Computer and Information Science and Engineering Steering Committee, 2015-2017.
2. Member, Graduate Recruiting Committee, 2015-2017.
3. Chair, Computer and Information Science and Engineering Industrial Advisory Board, 2014-2015.

B.2. Georgia Tech

1. Member, Massive Open Online Master's (MOOMS) Investigation Committee, 2012-2013.
2. Chair, School of Computer Science Ph.D. Review Committee, 2012.
3. Member, School of Computer Science Ph.D Review Committee, 2011.
4. Faculty Advisor, Grey H@T - Georgia Tech Undergraduate Security Club, 2011-2014.
5. Member, School of Computer Science Ph.D. Review Committee, 2011.
6. Member, School Advisory Committee, School of Computer Science, 2011-2013.
7. Member, School of Computer Science Chair Recruiting Committee, 2011.
8. Member, School of Computer Science Faculty Recruiting Committee, 2010, 2011.
9. Chair, College of Computing Ph.D. Welcome Weekend Committee, 2009, 2010, 2011 (co-chair).
10. Member, College of Computing Ph.D. Recruiting Committee, 2009.
11. Member, Georgia Tech Computer and Network Usage Security Policy (CNUSP) Evaluation Group, 2009.

C. Special Assignments

None.

D. Ph.D. Examining Committees

Ph.D. Examining Committees

1. Bradley Reaves, Department of Computer and Information Science and Engineering, University of Florida, Summer 2017.
Advisor: Professor Patrick Traynor.
2. Adam Bates, Department of Computer and Information Science and Engineering, University of Florida, Spring 2016.
Advisor: Professor Kevin Butler.
3. Benjamin Mood, Department of Computer and Information Science and Engineering, University of Florida, Spring 2016.
Advisor: Professor Kevin Butler.
4. Henry Carter, College of Computing, Georgia Tech, Fall 2015.
Advisor: Professor Patrick Traynor.
5. David Dewey, College of Computing, Georgia Tech, Fall 2015.
Advisor: Professor Patrick Traynor.
6. Lateef Yusuf, College of Computing, Georgia Tech, Spring 2014.
Advisor: Professor Umakishore Ramachandran.
7. Chaitrali Amrutkar, College of Computing, Georgia Tech, Summer 2013.
Advisor: Professor Patrick Traynor.
8. Long Lu, College of Computing, Georgia Tech, Summer 2013.
Advisor: Professor Wenke Lee.

9. Manos Antonakakis, College of Computing, Georgia Tech, Summer 2012.
Advisor: Professor Wenke Lee.
10. Junjie Zhang, College of Computing, Georgia Tech, Summer 2012.
Advisor: Professor Wenke Lee.
11. Italo Dacosta, College of Computing, Georgia Tech, Summer 2012.
Advisor: Professor Patrick Traynor.
12. Virendra Kumar, College of Computing, Georgia Tech, Summer 2012.
Advisor: Professor Alexandra Boldyreva.
13. Anirudh Ramachandran, College of Computing, Georgia Tech, Summer 2011.
Advisor: Professor Nick Feamster.
14. Vijay Balasubramaniyan, College of Computing, Georgia Tech, Summer 2011.
Advisor: Professor Mustaque Ahamad.
15. Kapil Singh, College of Computing, Georgia Tech, Summer 2011.
Advisor: Professor Wenke Lee.
16. Abhinav Srivastava, College of Computing, Georgia Tech, Summer 2011.
Advisor: Professor Jon Giffin.
17. Adam O'Neill, College of Computing, Georgia Tech, Summer 2010.
Advisor: Professor Alexandra Boldyreva.
18. David Cash, College of Computing, Georgia Tech, Fall 2009.
Advisor: Professor Alexandra Boldyreva.

External Member of Ph.D. Research Committee

None.

External Member of Ph.D. Examining Committee

1. Shannon Eggers, Department of Materials Sciences and Engineering - Nuclear Engineering Program, University of Florida, Fall 2016.
Advisor: Professor Kelly Jordan.
2. Ed Carlisle, Department of Electrical and Computer Engineering, University of Florida, Summer 2016.
Advisor: Professor Alan George.
3. Claudio Marforio, Department of Computer Science, Swiss Federal Institute of Technology Zurich (ETH Zurich), Fall 2015.
Advisor: Professor Srdjan Capkun.
4. Nils Ole Tippenhauer, Department of Computer Science, Swiss Federal Institute of Technology Zurich (ETH Zurich), Spring 2012.
Advisor: Professor Srdjan Capkun.
5. Bongkyoung Kwon, School of Electrical and Computer Engineering, Georgia Tech, Summer 2009.
Advisor: Professor John Copeland.

Ph.D. Thesis Proposal Committees

1. Bradley Reaves, Department of Computer and Information Science and Engineering, Spring 2016.
Advisor: Professor Patrick Traynor.
2. Maliheh Shirvanian, University of Alabama, Birmingham, Spring 2016.
Advisor: Professor Nitesh Saxena.
3. Benjamin Mood, Department of Computer and Information Science and Engineering, Fall 2015.
Advisor: Professor Kevin Butler.
4. Adam Bates, Department of Computer and Information Science and Engineering, Fall 2015.
Advisor: Professor Kevin Butler.
5. Lateef Yusuf, College of Computing, Georgia Tech, Spring 2013.
Advisor: Professor Umakishore Ramachandran.
6. Long Lu, College of Computing, Georgia Tech, Spring 2013.
Advisor: Professor Wenke Lee.
7. Chaitrali Amrutkar, College of Computing, Georgia Tech, Fall 2012.
Advisor: Professor Patrick Traynor.
8. Junjie Zhang, College of Computing, Georgia Tech, Fall 2011.
Advisor: Professor Wenke Lee.
9. Italo Dacosta, College of Computing, Georgia Tech, Fall 2011.
Advisor: Professor Patrick Traynor.
10. Manos Antonakakis, College of Computing, Georgia Tech, Fall 2011.
Advisor: Professor Wenke Lee.
11. Abhinav Srivastava, College of Computing, Georgia Tech, Spring 2011.
Advisor: Professor Jon Giffin.
12. Vijay Balasubramaniyan, College of Computing, Georgia Tech, Fall 2010.
Advisor: Professor Mustaque Ahamad.
13. Kapil Singh, College of Computing, Georgia Tech, Fall 2010.
Advisor: Professor Wenke Lee.
14. Anirudh Ramachandran, College of Computing, Georgia Tech, Fall 2010.
Advisor: Professor Nick Feamster.
15. Adam O'Neill, College of Computing, Georgia Tech, Spring 2010.
Advisor: Professor Alexandra Boldyreva.
16. David Cash, College of Computing, Georgia Tech, Spring 2009.
Advisor: Professor Alexandra Boldyreva.

Ph.D. Qualifying Exam Committees—Georgia Tech

1. Byoungyoung Lee, College of Computing, Georgia Tech, Spring 2013.
Advisor: Professor Wenke Lee.
2. Yizheng Chen, College of Computing, Georgia Tech, Spring 2013.
Advisor: Professor Wenke Lee.

3. Xinyu Xing, College of Computing, Georgia Tech, Spring 2013.
Advisor: Professor Wenke Lee.
4. Brad Reaves, College of Computing, Georgia Tech, Spring 2013.
Advisor: Professor Patrick Traynor.
5. Chaz Lever, College of Computing, Georgia Tech, Spring 2013.
Advisor: Professor Patrick Traynor.
6. Terry Nelms, College of Computing, Georgia Tech, Spring 2012.
Advisor: Professors Mustaque Ahamad and Roberto Perdesci.
7. Saurabh Chakradeo, College of Computing, Georgia Tech, Spring 2012.
Advisor: Professor Patrick Traynor.
8. Henry Carter, College of Computing, Georgia Tech, Spring 2012.
Advisor: Professor Patrick Traynor.
9. David Dewey, College of Computing, Georgia Tech, Spring 2012.
Advisor: Professor Jon Giffin.
10. Chaitrali Amrutkar, College of Computing, Georgia Tech, Fall 2011.
Advisor: Professor Patrick Traynor.
11. Yacin Nadji, College of Computing, Georgia Tech, Fall 2011.
Advisor: Professor Wenke Lee.
12. Yogesh Mundada, College of Computing, Georgia Tech, Fall 2011.
Advisor: Professor Nick Feamster.
13. Hyojoon Kim, College of Computing, Georgia Tech, Fall 2011.
Advisor: Professor Nick Feamster.
14. Ikpeme Erete, College of Computing, Georgia Tech, Spring 2011.
Advisor: Professor Alex Orso.
15. Chaitrali Amrutkar, College of Computing, Georgia Tech, Spring 2011.
Advisor: Professor Patrick Traynor.
16. Brendan Dolan-Gavitt, College of Computing, Georgia Tech, Spring 2011.
Advisor: Professor Wenke Lee and Professor Jon Giffin.
17. Sam Burnett, College of Computing, Georgia Tech, Fall 2010.
Advisor: Professor Nick Feamster.
18. Cong Shi, College of Computing, Georgia Tech, Fall 2010.
Advisor: Professor Mostafa Ammar and Professor Ellen Zegura.
19. Partha Kanuparth, College of Computing, Georgia Tech, Fall 2010.
Advisor: Professor Constantine Dorvolis.
20. Long Lu, College of Computing, Georgia Tech, Spring 2010.
Advisor: Professor Wenke Lee.
21. Virendra Kumar, College of Computing, Georgia Tech, Spring 2009.
Advisor: Professor Alexandra Boldyreva.
22. Frank Park, College of Computing, Georgia Tech, Spring 2009.
Advisor: Professor Patrick Traynor.

23. Italo Dacosta, College of Computing, Georgia Tech, Fall 2008.
Advisor: Professor Mustaque Ahamad and Professor Patrick Traynor.
24. Adam O'Neill, College of Computing, Georgia Tech, Fall 2008.
Advisor: Professor Alexandra Boldyreva.

E. External Member of M.S. Examining Committee

M.S. Thesis Defense Committees None.

F. Consulting and Advisory Appointments

1. Skim Reaper, *Co-Founder and CEO*, 2019-Present.
2. CryptoDrop Anti-Ransomware, *Co-Founder and CEO*, 2017-2018.
3. Pindrop Security, *Research Fellow and Co-Founder*, Spring 2012 - Spring 2014.
4. United States Army (via US Falcon), *Information Assurance Officer Training Program*, Spring 2010.
5. 3G Americas, *Characterizing the Limitations of Third-Party EAS over Cellular Text Messaging Systems*, Fall 2008.

G. Civic Activities

None.

IV. NATIONAL AND INTERNATIONAL PROFESSIONAL RECOGNITION

A. Honors and Awards

1. Fellow, Kavli Foundation, 2017.
2. Fellow, Center for Financial Inclusion at Accion, 2016.
3. Sloan Research Fellow, Alfred P. Sloan Foundation, 2014.

B. Invited Conference Session Chairmanships

1. Session Chair, *Work-in-Progress* at the *USENIX Security Symposium (SECURITY)*, 2016.
2. Session Chair, *Mobile Security* at the *USENIX Security Symposium (SECURITY)*, 2013.
3. Poster Chair, *USENIX Security Symposium (SECURITY)*, 2010, 2011.
4. Session Chair, *Privacy and Anonymity* at the *USENIX Workshop on Hot Topics in Security (HotSec)*, 2011.
5. Session Chair, *Security of Authentication and Protection Mechanisms* at the *IEEE Symposium on Security & Privacy (OAKLAND)*, 2011.
6. Session Chair, *Information Abuse* at the *IEEE Symposium on Security & Privacy (OAKLAND)*, 2010.
7. Session Chair, *RFID Security* at the *ACM Conference on Computer and Communications Security (CCS)*, 2009.
8. Session Chair, *Browser Security Session* at the *USENIX Security Symposium (SECURITY)*, 2009.
9. Session Chair, *Information Security Session* at the *IEEE Symposium on Security and Privacy (OAKLAND)*, 2009.
10. Session Chair, *Work-in-Progress* at the *IEEE Symposium on Security and Privacy (OAKLAND)*, 2009.
11. Session Chair, *Work/Opinions-in-Progress* at the *ISOC Network and Distributed Systems Security (NDSS) Symposium*, 2009.
12. Session Chair, *Privacy Session* at the *USENIX Security Symposium (SECURITY)*, 2008.

C. Professional Registration

None.

D. Patents

1. Patrick G. Traynor, Christian Peeters, Bradley G. Reaves, Hadi Abdullah, Kevin Butler, Jasmine Bowers, Walter N. Scaife, "Detecting SS7 Redirection Attacks With Audio-Based Distance Bounding", United State Patent # 11,265,717, Filed March 2019, Issued March 2022.
2. Patrick G. Traynor, Logan E. Blue, Luis Vargas, "Method and Apparatus for Differentiating Between Human and Electronic Speaker for Voice Interface Security", United State Patent # 11,176,960, Filed June 2019, Issued November 2021.
3. Patrick G. Traynor, Bradley G. Reaves, Logan E. Blue Practical End-to-End Cryptographic Authentication for Telephony Over Voice Channels, United State Patent # 11,329,831, Filed November 2018, Issued May 2022.

4. Walter Nolen Scaife, Patrick G. Traynor and Christian Peeters, "Payment Card Overlay Skimmer Detection", United States Patent # 10,496,914, Filed October 2017, Issued December 2019. (See also # 10,936,928)
5. Patrick G. Traynor, David P. Arnold, Walter Nolen Scaife, Christian Peeters, and Camilo Valez Cuervo, "Detecting counterfeit magnetic stripe cards using encoding jitter", United States Patent # 10,803,261, Filed May 2017, Issued October 2020.
6. Patrick G. Traynor, Bradley Reaves, Logan Blue, Luis Vargas, Hadi Abdullah, and Thomas Shrimpton, "Identity and content authentication for phone calls", United States Patent # 10,764,043, Filed Apr 2017, Issued September 2020.
7. Walter Nolen Scaife, Henry Carter, Patrick G. Traynor and Kevin R. B. Butler. "Malware Detection Through User Data Transformation Monitoring", United States Patent # 10,685,114. Filed September 2015, Issued June 2020.
8. Vijay A. Balasubramaniyan, Mustaque Ahamad, Patrick G. Traynor. "Using Single-Ended Audio Features to Automatically Determine Voice Call Provenance", United States Patent, #9,037,113 June 2010, Issued May 2015. (See also #9,516,497 and #10,523,809)
9. Patrick G. Traynor, Byungsuk Kim and Farooq Anjum. "Secure Localization for 802.11 Networks with Fine Granularity", United States Patent, #8,107,400, Filed July 2008, Issued January 2012.

E. Editorial and Reviewer Work for Technical Journals and Publishers

Associate Editor:

- *ACM Transactions on Information and System Security (TISSEC)* 2015-present

Guest Editor:

Journals

- *IEEE Security and Privacy Magazine (S&P)* 2013

Reviewer for:

Journals

- *ACM Transactions on Information and System Security (TISSEC)* 2008, 2009, 2010, 2011, 2012, 2013
- *IEEE Transactions on Dependable and Secure Computing (TDSC)* 2012, 2013
- *IEEE Security and Privacy Magazine (S&P)* 2010, 2011
- *Communications of the ACM (CACM)* 2010
- *Journal of Anesthesia & Analgesia* 2009
- *IEEE Transactions on Mobile Computing (TMC)* 2008, 2010, 2011, 2012, 2013
- *IEEE Transactions on Internet Technology (TOIT)* 2009, 2010
- *ACM Mobile Computing and Communications Review (MC2R)* 2008
- *IEEE/ACM Transactions on Networking (TON)* 2007, 2008
- *Journal of Pervasive and Mobile Computing (PMC)* 2009, 2010

- *IEEE Transactions on Parallel and Distributed Systems (TPDS)* 2005, 2009, 2010
- *IEEE Transactions on Computers (TOC)* 2010
- *Journal of Security and Communication Networks (SCN)* 2008
- *IEEE Communications Letters (CL)* 2007, 2009
- *IEEE Transactions on Wireless Communications (TWC)* 2007
- *Pervasive and Mobile Computing (PMC)* 2007
- *IEEE Transactions on Software Engineering (TSE)* 2007, 2008
- *Journal of Wireless Networks (WiNet)* 2006, 2007, 2008, 2009
- *Journal of Wireless Communications and Mobile Computing* 2006
- *ACM Computing Surveys (ACMCS)* 2006
- *Information Processing Letters (IPL)* 2006
- *IEEE Transactions on Very Large Scale Integration Systems (TVLSIS)* 2006

Conferences and Workshops

- *ACM International Conference on Mobile Computing and Networking (MobiCom)*, 2011
- *ACM Conference on Computer and Communications Security (CCS)*, 2008, 2011
- *IEEE Symposium on Security and Privacy (OAKLAND)* 2007, 2008
- *Computer Security Foundations (CSF)*, 2011
- *IFIP Conference on Data and Applications Security (DBSec)* 2008
- *Financial Cryptography (FC)* 2007, 2008
- *International Conference on VLSI Design (VLSI)* 2007
- *Annual Computer Security Applications Conference (ACSAC)* 2005, 2006, 2007
- *USENIX Workshop on Hot Topics in Security (HotSec)* 2007
- *International Conference on Information Systems Security (ICISS)* 2007
- *IEEE International Conference on Computer Engineering & Systems (ICCES)* 2007
- *International Workshop on Security (IWSec)* 2006, 2007
- *USENIX Security Symposium (SECURITY)* 2006, 2007
- *IEEE Sarnoff Symposium (SARNOFF)* 2007
- *International Conference on New Technologies, Mobility and Security (NTMS)* 2007
- *IEEE Infocom (INFOCOM)* 2007
- *Network and Distributed System Security Symposium (NDSS)* 2007
- *International Workshop on Storage Security and Survivability (IWSSS)* 2006
- *ACM Conference on Computer and Communications Security (CCS)* 2006

- *IEEE GLOBECOM (GLOBECOM) 2006*
- *International Conference on Mobile and Ubiquitous Systems: Networks and Services (MOBIQUITOUS) 2006*
- *IFIP Conference on Data and Applications Security (DBSec) 2006*
- *Emerging Trends in Information and Communications Security (ETRICS) 2006*
- *International Conference on Applied Cryptography and Network Security (ACNS) 2006*
- *ACM Symposium on Access Control Models and Technology (SACMAT) 2006*
- *IEEE Conference on Communication Systems Software & Middleware (COMSWARE) 2006*
- *International Conference on Cryptology in India (IndoCrypt) 2005*
- *IEEE Symposium on New Frontiers in Dynamic Spectrum Access (DySPAN) 2005*
- *European Symposium on Research in Computer Security (ESORICS) 2005*

F. Expert Witness Services

1. *Natalie Delgado, et al. v. Meta Platforms Inc., Case No.: 23-cv-04181 (N.D. Cal.):* Expert witness for the Defense (via Gibson, Dunn & Crutcher, LLP). *October 2025 - Present.*
2. *Google LLC v Headwater Research LLC:* Expert witness for the Plaintiff for Inter Partes Review (via Wolf, Greenfield & Sacks, P.C.). *August 2025 - Present.*
3. *Amazon v Headwater Research LLC:* Expert witness for the Plaintiff for Inter Partes Review (via Perkins Coie, LLP). *August 2025 - Present.*
4. *HBCU Messaging US LP v. Apple, Inc. et al., Civil No. 1:24-cv-1199,:* Expert witness for the Defense (via Fish & Richardson, LLP). *June 2025 - Present.*
5. *PACid v. Citibank, N.A., 1:24-cv-00272-DAE (W.D. Tex.):* Expert witness for the Defense (via Troutman Pepper Locke LLP). *April 2025 - Present.*
6. *Facebook Inc. Derivative Litigation., Consolidated C.A. No. 2018-0307-JTL (Del. Ch.):* Expert witness for the Defense (via Wachtell, Lipton, Rosen & Katz, LLP) *January 2025 - July 2025.*
7. *Averon US, Inc. vs AT&T Inc., Case No. 1:22-cv-01341-TMH:* Expert witness for the Defense (via O'Melveny & Myers, LLP). *November 2024 - September 2025.*
8. *RightQuestion, LLC v. AT&T Inc. et al.,, Case No. 2-24-cv-00094 -JRG:* Expert witness for the Defense (via Duane Morris, LLP). *September 2024 - Present.*
9. *ByteDance Ltd. vs CellSpin Soft, Inc.:* Expert witness for the Plaintiff for Inter Partes Review (via Fish & Richardson, LLP). *February 2024 - October 2024.*
10. *Bank of America, N.A., vs PACid:* Expert witness for the Plaintiff for Inter Partes Review (via McNish PLLC). *December 2023 - March 2024.*
11. *Microsoft vs Proxense, LLC:* Expert witness for the Plaintiff for Inter Partes Review (via Fish & Richardson, LLP). *November 2023 - July 2025.*
12. *Samsung vs Headwater Research LLC:* Expert witness for the Plaintiff for Inter Partes Review (via Fish & Richardson, LLP). *August 2023 - October 2025.*

13. *Advanced Coding Technologies LLC v. ByteDance PTE. Ltd., and TikTok PTE. Ltd.*: Expert witness for the Defense for Non-Infringement (via Fish & Richardson, LLP). July 2023 - September 2023.
14. *Epic Games, Inc. & Anor v Google LLC & Ors - Federal Court of Australia Proceeding NSD 190 of 2021*: Expert witness for the Defendant (via Corrs Chambers Westgarth). January 2023 - Present.
15. *Rubin vs KAHOOT! ASA and KAHOOT! EDU*: Expert witness for the Defendant for Inter Partes Review (via Vasquez Benisek & Lindgren, LLP). December 2022 - June 2024.
16. *Telefonaktiebolaget LM Ericsson vs Apple, Inc.*: Expert witness for the Defendant, Non-Infringement and Invalidity (via WilmerHale LLP). February 2022 - December 2022.
17. *Wepay Global Payments, LLC v. Bank of America N. A.*: Expert Witness for the Defendant (via WilmerHale LLP) September 2022 - November 2022.
18. *Apple vs. R.N Nehushtan Trust Ltd.*: Expert Witness for the Plaintiff for Inter Partes Review (via Fish & Richardson, LLP). August 2022 - June 2023.
19. *Apple/Microsoft vs. Zipit Wireless*: Expert Witness for the Plaintiff for Inter Partes Review (via Fish & Richardson, LLP). May 2021 - November 2022.
20. *Blackberry Inc v MobileIron, Inc.*: Expert Witness for the Plaintiff for Inter Partes Review (via Fish & Richardson, LLP). January 2021 - March 2021.
21. *Apple Inc v Seven Networks, LLC*: Expert Witness for the Plaintiff for Inter Partes Review (via Fish & Richardson, LLP). August 2019 - November 2020.
22. *mSIGNIA, Inc. v. InAuth, Inc.*: Expert Witness for the Defendant for Inter Partes Review, Non-Infringement and Invalidity, Trade Secrets (via Quinn Emanuel Urquhart and Sullivan, LLP). October 2017 - December 2018.
23. *Huawei v. T-Mobile*: Expert Witness for the Defendant for Non-Infringement (via WilmerHale LLP, Alston & Bird LLP) June 2016 - December 2017.
24. *Telefonaktiebolaget LM Ericsson v Apple*: Expert Witness for the Defendant for Non-Infringement, Invalidity (via WilmerHale LLP). June 2015 - December 2015.
25. *Mayfonk v Nike*: Expert Witness for the Plaintiff for Infringement/Trade Secrets (via Paul Hastings). June 2015 - November 2015.
26. *Maxim Integrated Products v Bank of the West*: Expert Witness for the Defendant for Non-Infringement (via Paul Hastings LLP). January 2014 - August 2014.
27. *Maxim Integrated Products v Comerica Inc, et al.*: Expert Witness for the Defendant for Non-Infringement (via McKenna, Long & Aldridge LLP). June 2014 - August 2014.
28. *William Grecia v. Apple Inc. et al.*: Expert Consultant for the Defendant for Invalidity (via Kirkland & Ellis LLP). July 2014 - August 2014.
29. *Intertrust Technologies Corp. v. Apple Inc.*: Expert Consultant for Defendant for Invalidity and Non-Infringement (via Kirkland & Ellis LLP). October 2013 - February 2014.
30. *Maxim Integrated Products v KeyCorp Bank*: Expert Witness for the Defendant for Non-Infringement (via Calfee, Halter & Griswold LLP) April 2013 - June 2013.
31. *Intellectual Ventures LLC vs. Check Point; et al.*: Expert Consultant for the Plaintiff for Infringement (via Susman Godfrey LLP), October 2012 - February 2015.

V. OTHER CONTRIBUTIONS

A. Seminar Presentations (Invited Papers and Talks at Meetings and Symposia)

1. Keynote: Well, It Worked on My Computer: Reproducibility in Computer Security Research. EPFL Summer Research Institute (SURI), July 2024. École Polytechnique Fédérale de Lausanne (EPFL, Switzerland).
2. Humans vs The Computer Interfaces: The Challenge of Separating Deepfakes/Bots from People. North Central Florida Institute of Internal Auditors (IIA), May 2024.
3. Humans vs The Computer Interfaces: The Challenge of Separating Deepfakes/Bots from People. UF Quest 2: Siri is my Superpower: Communicating with AI, March 2024. University of Florida.
4. Humans vs The Computer Interfaces: The Challenge of Separating Deepfakes/Bots from People. Federal Information Integrity Research and Development (FIIRD) Interworking Group (IWG), March 2024. via NITRD, OSTP.
5. AI driven voice cloning scams. Discussion at the White House with Anne Neuberger (Deputy National Security Advisor for Cyber and Emerging Technologies), Jessica Rosenworcel (Chair of the Federal Communications Commission) and Lina Khan (Chair of the Federal Trade Commission), January 2024. Lead Academic facilitator.
6. Keynote: Well, It Worked on My Computer: Reproducibility, Tech Transfer, and Computer Security Research. National Science Foundation Secure and Trustworthy Cyberspace (SaTC) Vision 2.0 Workshop, March 2023. University of Texas at Dallas.
7. Humans vs The Computer Interfaces: Separating Deepfakes/Bots from People Using Psychoacoustics. UCLA Electrical and Computer Engineering Distinguished Seminar, February 2023. University of California, Los Angeles.
8. Keynote: Exploiting the Gaps Between Human and Machine Understanding of Audio: Frameworks, Attacks, and Defenses. ISCA Symposium on Security and Privacy in Speech Communication (SPSC), November 2021. Virtual.
9. The State of Voice Cloning Technology. Federal Trade Commission (FTC) Workshop on Voice Cloning Technologies, January 2020. Washington, DC.
10. Because That's Where the Money Is: Designing Defenses for Entrenched Legacy Payment Systems. North Carolina State University Department of Computer Science Colloquium, January 2020. Raleigh, NC.
11. Moving from research to practice: How to maximize the impact of SaTC projects. National Science Foundation Secure and Trustworthy Cyberspace (SaTC) PI Meeting, October 2019. Alexandria, VA.
12. Because That's Where the Money Is: Designing Defenses for Entrenched Legacy Payment Systems. Purdue University Computer Science Excellence Lecture Series, October 2019. West Lafayette, IN.
13. Because That's Where the Money Is: Designing Defenses for Entrenched Legacy Payment Systems. Bank of America - Colloquium Series, March 2019. Charlotte, NC.
14. Because That's Where the Money Is: Designing Defenses for Entrenched Legacy Payment Systems. CISPA – Helmholtz Center for Information Security, Saarland University, February 2019. Saarbrücken, Germany.
15. Because That's Where the Money Is: Designing Defenses for Entrenched Legacy Payment Systems. University of Maryland - Distinguished Colloquium, February 2019. College Park, MD.

16. Responsible Finance for the Digital Client. Foromic Conference, October 2018. Barranquilla, Colombia.
17. Panel: Authentication Challenges for New Interfaces, Devices, and Wireless Networks. ACM Conference on Security and Privacy in Wireless and Mobile Networks, June 2018. Stockholm, Sweden.
18. Sonar: Detecting SS7 Redirection Attacks Via Call Audio-Based Distance Bounding. CyberSecurity@KAIST Workshop - KAIST, June 2018. Daejeon, South Korea.
19. Why Caller-ID Spoofing Is So Easy (and Why End-To-End Solutions Are the Way Forward). IEEE Workshop on Technology and Consumer Protection (ConPro'18), May 2018. San Francisco, CA.
20. Panel: The Future of Cybersecurity. SEC Academic Conference - Auburn University, May 2018. Auburn, AL.
21. Sound Principles: Verifying Voice Commands in an IoT World. IoT Security Workshop - Aalto University, September 2017. Helsinki, Finland.
22. Who Do I Think You Are? Challenges and Opportunities in Telephony Authentication. Eurecom Institute, September 2017. Sophia Antipolis, France.
23. Panel: Infrastructure Stability. ITU-T Focus Group Digital Financial Services, December 2016. Geneva, Switzerland.
24. Who Do I Think You Are? Challenges and Opportunities in Telephony Authentication. ETH Zurich, December 2016. Zurich, Switzerland.
25. Who Do I Think You Are? Challenges and Opportunities in Telephony Authentication. University of Richmond, October 2016. Richmond, Virginia.
26. Who Do I Think You Are? Challenges and Opportunities in Telephony Authentication. Indiana University, September 2016. Bloomington, Indiana.
27. Who Do I Think You Are? Challenges and Opportunities in Telephony Authentication. Aalto University Computer Science Department Forum, August 2016. Helsinki, Finland.
28. Who Do I Think You Are? Challenges and Opportunities in Telephony Authentication. KAIST Information Security Seminar - Korean Advanced Institute of Science and Technology, June 2016. Daejeon, South Korea.
29. Updated Mobile Money Vulnerability Report. International Telecommunications Union Digital Financial Services Working Group Workshop, May 2016. Washington, DC.
30. Who Do I Think You Are? Challenges and Opportunities in Telephony Authentication. UF Eye Opener Discovery Breakfast - University of Florida, May 2016. Gainesville, FL.
31. Who Do I Think You Are? Challenges and Opportunities in Telephony Authentication. Illinois Science of Security (SoS) Lablet Speaker Series - University of Illinois, Urbana-Champaign, April 2016. Urbana-Champaign, Illinois.
32. New Trends in Cybersecurity: Vulnerabilities in Branchless Banking Systems. United States Departments of State and Justice - Cybersecurity and Cybercrime Workshop for Lusophone Africa, September 2015. Maputo, Mozambique.
33. New Trends in Cybersecurity: Vulnerabilities in Branchless Banking Systems. United States Departments of State and Justice - ECCAS Cybersecurity and Cybercrime Workshop, August 2015. Kinshasa, Democratic Republic of Congo.

34. Chasing Telephony Security: Where the Wild Things... Are? University of Florida - Department Colloquium, January 2014. Gainesville, FL.
35. Chasing Telephony Security: Where the Wild Things... Are? Verizon Wireless RNC/Data Center, October 2013. Alpharetta, GA.
36. Chasing Telephony Security: Where the Wild Things... Are? University of Waterloo - CrySP Speaker Series on Privacy, October 2013. Waterloo, ON, Canada.
37. Analyzing Malicious Traffic in Cellular Networks. GSM Association's (GSMA) Mobile Malware Community Workshop, July 2013. Mountain View, CA.
38. Threats to Mobile Devices. US Federal Trade Commission (FTC) Public Forum - Invited Speaker, June 2013. Washington, D.C.
39. Chasing Telephony Security: Where the Wild Things... Are? University of Wisconsin - Madison, Security Seminar, March 2013. Madison, WI.
40. The Core of the Matter: Analyzing Malicious Traffic in Cellular Carriers. Invited Talk: Centre for Secure Information Technologies (CSIT) Queen's University, March 2013. Belfast, Northern Ireland.
41. Chasing Telephony Security: Where the Wild Things... Are? Stanford Security Seminar, March 2013. Stanford, CA.
42. Chasing Telephony Security: Where the Wild Things... Are? University of California, Berkeley, Security Group, March 2013. Berkeley, CA.
43. Chasing Telephony Security: Where the Wild Things... Are? Carnegie Mellon University CyLab Seminar, February 2013. Pittsburgh, PA.
44. Chasing Telephony Security: Where the Wild Things... Are? University of Oregon Department of Computer Science Colloquium, November 2012. Eugene, OR.
45. Chasing Telephony Security: Where the Wild Things... Are? University of Washington Department of Electrical Engineering, Network Security Lab (NSL): Invited Talk, November 2012. Seattle, WA.
46. Needles and Haystacks: Digging for Ground Truth on Mobile Malware. ZISC Workshop on Secure Mobile and Cloud Computing, ETH Zurich, June 2012. Zurich, Switzerland.
47. Panel: Advice for Early Career Faculty. CRA Career Mentoring Workshop, February 2012. Washington, D.C.
48. Research Challenges in Cellular and Mobile Network Security. US-China Software Workshop (Co-Sponsored by NSF and NSFC), September 2011. Beijing, China.
49. Mobile Security: Understanding Risks to Critical Infrastructure. Invited Talk: US Department of State East African Workshop on Cyberspace Security, July 2011. Nairobi, Kenya.
50. Tomorrow's Issues: Solving the Mobile Security Threat. Invited Talk: Centre for Secure Information Technologies (CSIT) Queen's University, March 2011. Belfast, Northern Ireland.
51. PinDr0p: Using Single-Ended Audio Features to Determine Call Provenance. Invited Talk: MITRE Corporation, March 2011. Burlington, MA.
52. Defeating Session Hijacking Attacks with Disposable Web Credentials. Invited Talk: Facebook, February 2011. Palo Alto, CA.

53. Understanding the Disruptive Potential of Malware in Cellular Networks. Invited Talk: RSA Conference, February 2011. San Francisco, CA.
54. Panel: Voice Security – Now Just a False Sense of Security and Privacy. Invited Panelist: Mobile Security Symposium, February 2011. San Francisco, CA.
55. Understanding the Disruptive Potential of Malware in Cellular Networks. Invited Talk: Concordia University, May 2010. Montreal, QC, Canada.
56. Characterizing the Impact of Rigidity on the Security of Cellular Networks. Qualcomm Research, March 2010. San Diego, CA.
57. Privacy and Security Concerns for Personal and Mobile Health Devices. Invited Talk: Workshop to Set A Research Agenda for Privacy and Security of Healthcare Technologies, October 2009. Indianapolis, IN.
58. Considerations for EAS Over Cellular Text Messaging Services. 3G Americas Webinar, July 2009.
59. University Telephony Research Panel. Conference on Principles, Systems and Applications of IP Telecommunications (IPTCOMM), July 2009.
60. The Evolving Mobile Landscape: Emerging Security Threats. Mobile Security eConference, SC Magazine, June 2008.
61. Characterizing the Impact of Rigidity on the Security of Cellular Networks. University of Washington, February 2009. Seattle, WA.
62. Characterizing the Impact of Rigidity on the Security of Cellular Networks. Microsoft Research, February 2009. Redmond, WA.
63. Next Year's Problems. Secure Computing (SC) Magazine Webinar, November 2008.
64. Panel: Embedded Systems and their Increasing Impact on Infrastructure Security. Workshop on Embedded Systems Security (WESS), October 2008.
65. Can you DoS me now? Security Issues in Cellular Networks. Georgia Institute of Technology, September 2008. Atlanta, GA.
66. Characterizing the Impact of Rigidity on the Security of Cellular Networks. Georgia Institute of Technology, April 2008. Atlanta, GA.
67. Characterizing the Impact of Rigidity on the Security of Cellular Networks. AT&T Research Labs, April 2008. Florham Park, NJ.
68. Characterizing the Impact of Rigidity on the Security of Cellular Networks. University of Arizona, March 2008. Tucson, AZ.
69. Cellular Networks Security Panel. USENIX Security Symposium, August 2007. Boston, MA.
70. malnets:Large-Scale Malicious Networks via Compromised Access Points. The Pennsylvania State University - ACM Club Invited Speaker, October 2006. State College, PA.
71. malnets:Large-Scale Malicious Networks via Compromised Access Points. The University of Michigan, October 2006. Ann Arbor, MI.
72. Exploiting Open Functionality in SMS-Capable Cellular Networks. The University of Michigan, October 2006. Ann Arbor, MI.
73. Exploiting Open Functionality in SMS-Capable Cellular Networks. High Technology Crime Investigation Association (HTCIA), September 2006. Pittsburgh, PA.

74. Trends in Security: Critical Engineering in the Large. Schlumberger Innovate IT! Workshop, May 2006. Cambridge, MA.
75. Exploiting Open Functionality in SMS-Capable Cellular Networks. InfraGard Pittsburgh Chapter General Meeting, September 2006. Pittsburgh, PA.
76. Exploiting Open Functionality in SMS-Capable Cellular Networks. InfraGard Pittsburgh Chapter General Meeting, September 2006. Pittsburgh, PA.
77. How technology can fight digital fakery. The Babbage Podcast/The Economist <https://shows.acast.com/theeconomistbabbage/episodes/babbage-how-to-detect-a-deepfake/>, January 2023.
78. Deepfake audio has a tell and researchers can spot it. Ars Technica <https://arstechnica.com/information-technology/2022/09/researchers-use-fluid-dynamics-to-spot-deepfake-voices/>, September 2022.
79. This security tool could help stop the problem of ransomware in its tracks. TheJournal.ie <https://www.thejournal.ie/ransomware-researchers-stop-2875032-Jul2016/>, July 2016.

B. Special Activities

Presentations to Lay Media

1. Researchers Unleash Ransomware Annihilation. BankInfoSecurity - <http://www.bankinfosecurity.com/researchers-unleash-ransomware-annihilation-a-9255/>, July 2016.
2. CryptoDrop Stops Ransomware by Stopping its Encryption. Security Intelligence - https://securityintelligence.com/news/cryptodrop-stops-ransomware-by-stopping-its-encryption/?utm_source=tfeed&utm_medium=twitter, July 2016.
3. Ransomware 'stopped' by new software. BBC - <http://www.bbc.com/news/technology-36772461>, July 2016.
4. Researchers create effective anti-ransomware solution. Help Net Security - <https://www.helpnetsecurity.com/2016/07/12/anti-ransomware-solution/>, July 2016.
5. Florida U boffins think they've defeated all ransomware. http://www.theregister.co.uk/2016/07/12/ransomware_defeated/, July 2016.
6. This Anti-Ransomware Tool Could Save You Hundreds of Pounds. Huffington Post - http://www.huffingtonpost.co.uk/entry/anti-ransomware-tool-save-hundreds-pounds_uk_57838beee4b0935d4b4b30ba, July 2016.
7. Researchers develop method to stop 100% of ransomware before it encrypts all files. Myce - <http://www.myce.com/news/researchers-develop-method-stop-100-ransomware-encrypts-files-79873/>, July 2016.
8. Desarrollan una solución para detener el ransomware. ComputerHoy - <http://computerhoy.com/noticias/software/desarrollan-solucion-detener-ransomware-47972/>, July 2016.

9. Why your antivirus software can't stop ransomware. Futurity - <http://www.futurity.org/ransomware-computer-files-1198242-2/>, July 2016.
10. CryptoDrop Gives Users Hope to Prevent Ransomware Infections in the Future. Softpedia - <http://news.softpedia.com/news/cryptodrop-gives-users-hope-to-prevent-ransomware-infections-in-the-future-506187.shtml>, July 2016.
11. Could this be the answer to the ransomware threat?, Consumer Affairs. Consumer Affairs - <https://www.consumeraffairs.com/news/could-this-be-the-answer-to-the-ransomware-threat-071116.html>, July 2016.
12. Extortion extinction: Researchers develop a way to stop ransomware. Phys.org - <http://phys.org/news/2016-07-extortion-extinction-ransomware.html>, July 2016.
13. Researchers Develop A Way To Stop Ransomware By Watching The Filesystem. Slashdot - <https://yro.slashdot.org/story/16/07/08/2242244/researchers-develop-a-way-to-stop-ransomware-by-watching-the-filesystem>, July 2016.
14. Mohul Ghosh. Trak.in - Digital Money Apps In India Are Unsafe and Unsecured - Researchers. <http://trak.in/tags/business/2015/08/17/digital-money-apps-india-unsafe-unsecured/>, August 2015.
15. Richard Handford. Mobile World Live - Survey finds security holes in mobile money apps. <http://www.mobileworldlive.com/money/news-money/survey-finds-security-holes-in-mobile-money-apps/#.Vc27Y-QTmsQ.twitter>, August 2015.
16. JENNIFER VALENTINO-DEVRIES. Wall Street Journal - Researchers Find Security Flaws in Developing-World Money Apps. <http://blogs.wsj.com/digits/2015/08/11/researchers-find-security-flaws-in-developing-world-money-apps/>, August 2015.
17. Jonathon Cheng. Wall Street Journal - Samsung Phone Studied for Possible Security Gap. <http://online.wsj.com/news/articles/SB10001424052702304244904579276191788427198>, December 2013.
18. N. V. The Economist - The Threat in the Pocket. <http://www.economist.com/blogs/babbage/2013/10/difference-engine-0?fsrc=scn/fb/wl/bl/thethreatinthepocket>, October 2013.
19. Antone Gonsalves. ComputerWorld - Let's Dump Anti-Virus and Move On:. <http://blogs.computerworld.com/mobile-security/22969/lets-dump-av-and-move>, October 2013.
20. Mathew J. Schwartz. InformationWeek - Google: Don't Fear Android Malware. <http://www.informationweek.com/security/mobile/google-dont-fear-android-malware/240162399>, October 2013.
21. Kirsten Doyle. ITWeb - Android Threat Exaggerated, or is it? http://www.itweb.co.za/index.php?option=com_content&view=article&id=68055, October 2013.
22. Danielle Walker. SC Magazine - Mobile malware prevalence expands, but privacy-abusing apps should be top of mind. <http://www.scmagazine.com/mobile-malware-prevalence-expands-but-privacy-abusing-apps-should-be-top-of-mind/article/300597/>, June 2013.

23. Jim Burress. WABE NPR - Mobile Web Browsers Full of Security Risks, Tech Professor Finds. <http://wabe.org/post/mobile-web-browsers-full-security-risks-tech-professor-finds>, December 2012.
24. Mark Huffman. Consumer Affairs - Georgia Tech: mobile browsers fail safety test. <http://www.consumeraffairs.com/news/georgia-tech-mobile-browsers-fail-safety-test-120612.html>, December 2012.
25. Matthew J. Schwartz. Information Week - Blame Screen Size: Mobile Browsers Flunk Security Tests. <http://www.informationweek.com/security/mobile/blame-screen-size-mobile-browsers-flunk/240143999>, December 2012.
26. Jon Gold. Network World - Ga. Tech researchers: Mobile Browsers need better HTTPS indicators. <http://www.networkworld.com/news/2012/120512-mobile-browsers-264846.html>, December 2012.
27. United Press International. Study: Most mobile Web browsers unsafe. http://www.upi.com/Science_News/Technology/2012/12/05/Study-Most-mobile-Web-browsers-unsafe/UPI-73431354743353/#ixzz2EGtQsuLd, December 2012.
28. Suzanne Choney. Mobile browser woes can fool even experts: report. <http://www.nbcnews.com/technology/mobile-browser-woes-can-fool-even-experts-report-1C7451203>, December 2012.
29. Meghan Kelly. VentureBeat - 3 hot security startups to watch. <http://venturebeat.com/2012/02/27/3-security-startups-to-watch-at-the-2012-rsa-conference/>, February 2012.
30. Jacob Goodwin. Government Security News - RSA 2012 – Pindrop Security can distinguish a fraudulent phone call from a real one. <http://www.gsnmagazine.com/node/25721?c=communications>, February 2012.
31. Matt Liebowitz. Phone hack logs keystrokes from nearby computers. MSNBC.com - http://www.msnbc.msn.com/id/44993238/ns/technology_and_science-security/#.TqU5MNSjPh4, October 2011.
32. Jacob Aron. iPhone keylogger can snoop on desktop typing. New Scientist - <http://www.newscientist.com/article/dn21059-iphone-keylogger-can-snoop-on-desktop-typing.html>, October 2011.
33. iPhone Keylogger Can Snoop on Desktop Typing. Slashdot - <http://mobile.slashdot.org/story/11/10/18/2346222/iphone-keylogger-can-snoop-on-desktop-typing>, October 2011.
34. Robert Lemos. Smart Phones Could Hear Your Password. Technology Review - <http://www.technologyreview.com/computing/38913/?p1=A2>, October 2011.
35. Kevin McCaney. Bad vibrations: How smart phones could steal PC passwords. Government Computer News - <http://gcn.com/articles/2011/10/18/smart-phone-sensors-steal-keystrokes.aspx>, October 2011.
36. PhysOrg. Turning iPhone into spiPhone: Smartphones' accelerometer can track strokes on nearby keyboards. PhysOrg.com - <http://www.physorg.com/news/2011-10-iphone-spihone-smartphones-accelerometer-track.html>, October 2011.

37. Brid-Aine Parnell. Securo-boffins call for 'self-aware' defensive technologies. The Register - http://www.theregister.co.uk/2011/09/14/self_aware_cyber_security_technologies_should_be_a_top_priority/, September 2011.
38. Clay Dillow. 'PinDr0p' Tech Uses Unique Noise Fingerprints to Trace Calls. Popular Science - <http://www.popsci.com/technology/article/2010-10/pindr0p-tech-tags-phone-calls-unique-fingerprints-trace-call-paths-across-networks>, October 2010.
39. Lewis Page. Voice-routing call fingerprint system fights vishing. The Register - http://www.theregister.co.uk/2010/10/06/voice_fingerprints, October 2010.
40. Science Daily. Voice Phishing: System to Trace Telephone Call Paths Across Multiple Networks Developed. <http://www.sciencedaily.com/releases/2010/10/101005121820.htm>, October 2010.
41. Brian Kalish. To Text or Not to Text During Emergencies. NextGov.com - http://www.nextgov.com/nextgov/ng_20100914_5986.php?oref=topnews, September 2010.
42. Ki Mae Heussner. 'Operation Chokehold': Fake Steve Jobs Rallies iPhone Users to Cripple AT&T Network. ABC News - <http://abcnews.go.com/Technology/GadgetGuide/fake-steve-jobs-rallies-iphone-users-cripple-att/story?id=9355447>, December 2009.
43. Bob Brown. Researchers Set Their Sights on iPhones, Mobile Malware. PC World Magazine - http://www.pcworld.com/article/182005/iphone_worms_mobile_malware.html?tk=rss, November 2009.
44. MacGregor Campbell. Botnets show their disruptive potential. New Scientist Magazine - <http://www.newscientist.com/article/mg20427347.000-mobile-botnets-show-their-disruptive-potential.html>, November 2009.
45. Angela Moscaritolo. Remote repair for infected phones in development. SC Magazine - <http://www.scmagazineus.com/remote-repair-for-infected-phones-in-development/article/157504/>, November 2009.
46. Bob Brown. iPhone worms, other smartphone malware in researchers' sights. Network World - <http://www.networkworld.com/news/2009/111109-smartphone-security-georgia-tech.html?hpg1=bn>, November 2009.
47. Urvaksh Karkaria. GT researchers work to secure cellphones. Atlanta Business Chronicle - <http://atlanta.bizjournals.com/atlanta/blog/atlantech/2009/11/cellphone.html>, November 2009.
48. Making Carriers Shoulder Smartphone Security. http://mobile.slashdot.org/story/09/11/11_/2318247/Making-Carriers-Shoulder-Smartphone-Security?art_pos=31, November 2009.
49. Ben Meyer. Georgia Tech to Lead Fight Against Cell Phone Hackers. NBC 11 Atlanta - <http://www.11alive.com/news/local/story.aspx?storyid=132505&catid=3>, July 2009.
50. Illena Armstrong. Safeguarding your mobile networks. SC Magazine - <http://www.scmagazineus.com/Safeguarding-your-mobile-networks/article/138289/>, June 2009.
51. Kelli B. Grant. Four Free Cellphone Apps to Help Manage Your Money. SmartMoney Magazine - <http://www.smartmoney.com/Spending/Deals/4-Great-Free-Finance-Apps-for-Cellphones/>, June 2009.

52. Amanda Hoffstrom. Technology's limitations in alerting campus danger. UWire Magazine - <http://www.uwire.com/Article.aspx?id=3738798>, February 2009.
53. Laura Sydell. Compromise Allows Obama To Keep BlackBerry. National Public Radio - <http://www.npr.org/templates/story/story.php?storyId=99790788>, January 2009.
54. Dennis Carter. Questions abound as emergency alert flops Virginia Tech's text-message alert system failed when the sound of gunfire was heard on campus; officials scramble to understand why. eSchool News - http://www.eschoolnews.com/iphone/top-story/index.cfm?i=56122#_56122, November 2008.
55. Jessica Bauer. Study: Text alerts may fail in real emergency. Diamondback Online - <http://media.www.diamondbackonline.com/media/storage/paper873/news/2008/10/14/News/Study.Text.Alerts.May.Fail.In.Real.Emergency-3485509.shtml>, October 2008.
56. Associated Press. Hackers Expected To Start Targeting Cell Phones. <http://cbs5.com/watercooler/Cell.Phones.Hackers.2.840909.html>, 2008.
57. Associated Press. College alert systems unreliable, study says. http://www.ajc.com/search/content/metro/stories/2008/09/25/college_campus_alerts.html, 2008.
58. Lee Shearer. Study: Campus alerts unreliable. Athens Banner Herald http://www.onlineathens.com/stories/092508/uga_336494829.shtml, 2008.
59. Bill Ray. 3G Americas warns against text warning systems. The Register - http://www.theregister.co.uk/2008/09/18/emergency_text/, 2008.
60. 3G Americas. 3G Americas Highlights New Research Report on Use of Cellular Text Messaging for Emergency Alert Services. 3G Americas http://www.3gamericas.org/English/news_room/DisplayPressRelease.cfm?id=3400&s=ENG, 2008.
61. Evan Koblentz. Web Exclusive: From Messaging to Management Duty. Wireless Week - <http://www.wirelessweek.com/Messaging-to-Management-Duty.aspx>, 2008.
62. Christopher Beam. How Do You Intercept a Text Message? Turn your cell phone into a spy gadget. Slate Magazine <http://www.slate.com/id/2161402/>, 2007.
63. Jamming Cellphones with Text Messages. Slashdot <http://it.slashdot.org/it/05/10/05/1839217.shtml?tid=215&tid=172>, 2005.
64. Cell phone networks at risk? CNN http://money.cnn.com/2005/10/05/technology/hacker_cellphones/, 2005.
65. John Schwartz. Text Hackers Could Jam Cellphones, a Paper Says. The New York Times <http://www.nytimes.com/2005/10/05/technology/05phone.html?ex=1286164800&en=d917b9cd43dfaa31&ei=5090&partner=rssuserland&emc=rss>, 2005.
- 66.
- 67.
- 68.