

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

PALO ALTO NETWORKS, INC.
Petitioner

v.

CENTRIPETAL NETWORKS, INC.
Patent Owner

Case IPR2021-01155
Patent No. 10,567,343

DECLARATION OF DR. ROBERT AKL, D.Sc.

TABLE OF CONTENTS

I.	Introduction.....	7
II.	Background and Qualifications	8
III.	Level of Ordinary Skill in the Art.....	14
IV.	Materials Considered and Relied Upon	15
V.	Legal Standards.....	19
	A. Anticipation	19
	B. Obviousness.....	21
	C. Claim Construction.....	26
VI.	Background on the State of the Art	27
	A. Packet-Switched Networking and Communication Protocols.....	29
	B. Network Security Systems—Firewalls.....	31
	C. Network Security Systems—Intrusion Detection Systems and Intrusion Prevention Systems.....	35
	D. Inspecting and Applying Packet Filtering Rules to Encrypted and Unencrypted SSL/TLS Record Protocol Data.....	38
VII.	Overview of the '343 Patent.....	49
	A. Subject Matter Overview	49
	B. File History of the '343 Patent	58
	C. Interpretation of the '343 Patent Claims at Issue.....	61
	“operator”	61
VIII.	Overview of the Cited References	62
	A. Sourcefire (Ex. 1004).....	63

B.	Mantripragada (Ex. 1007)	71
IX.	The Prior Art Renders Claims 1-20 Obvious	71
A.	Independent Claims 1, 8, and 15	72
	[1.pre], [1.a]; [8.pre], [8.a], [8.b], [8.c]; [15.pre], [15.a]	73
	[1.b], [8.d], [15.b]: [determining/ determine], based on a packet header field value, whether [each packet of] the plurality of packets comprises data corresponding to first criterion specified by one or more packet-filtering rules;	78
	[1.c], [8.e], [15.c]: responsive to a determination [by the computing system] that a packet header field value of [the/a] first portion of packets comprises data corresponding to the first criterion specified by at least one matching packet-filtering rule, [applying/ apply], [by the computing system and] to each packet in the first portion of packets, one or more operators specified by the at least one matching packet-filtering rule;	83
	[1.d], [8.f], [15.d]: [determining/ determine], based on an application header field value, [the/a] second portion of packets based on whether the first portion of packets comprises data corresponding to second criterion specified by one or more operators specified by the at least one matching packet-filtering rule; and	86
	[1.e], [8.g], [15.e]: responsive to determining the second portion of packets that comprises data corresponding to the second criterion specified by one or more operators specified by the at least one matching packet-filtering rule, [applying/apply], [by the computing system and] to each packet in the second portion of packets [that match the second criterion], at least one packet transformation function configured to prevent an exfiltration operation, wherein the at least one packet transformation function indicates whether each packet in the second portion of packets is allowed to continue toward its destination.	92
B.	Dependent Claims 2, 9, and 16	95

	[2.pre], [9.pre], [16.pre]: The method of claim 1, wherein: / The apparatus of claim 8, wherein / The one or more non-transitory computer-readable media of claim 15, wherein:.....	95
	[2.a], [9.a], [16.a]: the first portion of packets comprises a packet header field value indicating a protocol type associated with a particular type of data transfer, and corresponding to the first criterion specified by the at least one matching packet - filtering rule.....	95
C.	Dependent Claims 3, 10, and 17.....	96
	[3.pre], [10.pre], [17.pre]: The method of claim 1, wherein: / The apparatus of claim 8, wherein: / The one or more non-transitory computer-readable media of claim 15, wherein:.....	96
	[3.a], [10.a], [17.a]: the first portion of packets comprises data indicating a destination port number associated with a particular type of data transfer, and corresponding to the first criterion specified by the at least one matching packet filtering rule.	96
D.	Dependent Claims 4, 11, and 18.....	99
	[4.pre], [11.pre], [18.pre]: The method / apparatus / one or more non-transitory computer-readable media of claim 1/8/15, wherein.....	99
	[4.a], [11.a], [18.a]: the application header field value identifies one or more hypertext transfer protocol (HTTP) methods, wherein the second portion of packets comprises a plurality of HTTP packets, wherein the second portion of packets comprises a third portion of packets and a fourth portion of packets, and wherein each packet of the third portion of packets comprises at least one GET method value, and each packet of the fourth portion of packets comprises at least one of a PUT method value, a POST method value, or a CONNECT method value, [the method further comprising:/and wherein the instructions, when executed by the at least one processor cause the apparatus to:/ and wherein the instructions, when	

	executed by the one or more computing devices, cause the one or more computing devices to:]	100
	[4.b], [11.b], [18.b]: [allowing/ allow] each packet of the third portion of packets to continue toward its respective destination; and.....	102
	[4.c], [11.c], [18.c]: [blocking/ block] each packet of the fourth portion of packets from continuing toward its respective destination.	103
E.	Dependents Claim 5 and 12	104
	[5.pre], [12.pre]: The method of claim 1, the method further comprising: / The apparatus of claim 8, wherein the instructions, when executed by the one or more computing devices, cause the apparatus to:.....	104
	[5.a], [12.a]: [allowing/ allow], based on a determination that the application header field value indicates a request for [specified resource data/data from a specified resource], each packet of the second portion of packets indicating a request for [specified resource data/data from a specified resource] to continue toward its respective destination; and.....	105
	[5.b], [12.b]: [blocking/ block], based on a determination that the application header field value indicates a request to submit data to be processed by a specified resource, each packet of the second portion of packets indicating a request to submit data to be processed by [the/a] specified [resource] from continuing toward its respective destination.....	106
F.	Dependent Claims 6, 13, and 19.....	107
	[6.pre], [13.pre], [19.pre]: The method of claim 1, further comprising / The apparatus of claim 8, wherein the instructions, when executed by the one or more computing devices, cause the apparatus to / The one or more non-transitory computer-readable media of claim 15, further comprising.....	107
	[6.a], [13.a], [19.a]: [dropping/drop] packets associated with a particular type of data transfer responsive to a	

	determination that the packets associated with the particular type of data transfer comprise data corresponding to a particular transport layer security (TLS) version value	107
G.	Dependent Claims 7, 14, and 20.....	109
	[7.pre], [14.pre], [20.pre]: The method of claim 1, further comprising / The apparatus of claim 8, wherein the instructions, when executed by the one or more computing devices, cause the apparatus to / The one or more non-transitory computer-readable media of claim 15, further comprising.....	109
	[7.a], [14.a], [20.a]: [dropping/drop] packets associated with a particular type of data transfer responsive to a determination that the packets associated with the particular type of data transfer comprise data corresponding to a particular real-time transport protocol value.	109
X.	Secondary Considerations	112
XI.	Conclusion	113

I, Robert Akl, D.Sc. of Dallas, Texas, declare that:

I. Introduction

1. My name is Robert Akl, and I have been retained by counsel for Petitioner Palo Alto Networks Inc. (“PAN” or “Petitioner”) as an expert witness to provide assistance regarding U.S. Patent No. 10,567,343 (“the ’343 Patent”). Specifically, I have been asked to consider the validity of claims 1-20 of the ’343 Patent (the “Challenged Claims”) in view of prior art, anticipation and obviousness considerations, and understanding of a person of ordinary skill in the art at the time of the invention as it relates to the ’343 Patent. I have personal knowledge of the facts and opinions set forth in this declaration and believe them to be true. If called upon to do so, I would testify competently thereto.

2. I am being compensated for my time at my standard consulting rate. I am also being reimbursed for expenses that I incur during the course of this work. My compensation is not contingent upon the results of my study, the substance of my expert opinions, or the outcome of any proceeding involving the challenged claims. I have no financial interest in the outcome of this matter or on the pending litigation between Petitioner and Patent Owner.

3. My analysis here is based on my years of education, research and experience, as well as my investigation and study of relevant materials, including those cited herein.

4. I may rely upon these materials, my knowledge and experience, and/or additional materials to rebut arguments raised by the Patent Owner. Further, I may also consider additional documents and information in forming any necessary opinions, including documents that may not yet have been provided to me.

5. My analysis of the materials produced in this proceeding is ongoing and I will continue to review any new material as it is provided. This declaration represents only those opinions I have formed to date. I reserve the right to revise, supplement, and/or amend my expert opinions stated herein based on new information and on my continuing analysis of the materials already provided.

II. Background and Qualifications

6. I am an expert in the field of telecommunication systems. I have studied, taught, practiced, and researched this field for over 25 years. I have summarized in this section my educational background, work experience, and other relevant qualifications. Attached hereto as Appendix A, is a true and correct copy of my curriculum vitae describing my background and experience.

7. I earned my Bachelor of Science degrees in Electrical Engineering and Computer Science *summa cum laude* with a grade point average of 4.0/4.0 and a ranking of first in my undergraduate class from Washington University in St. Louis in 1994. In 1996, I earned my Master of Science degree in Electrical Engineering from Washington University in St. Louis with a grade point average of 4.0/4.0. I

earned my Doctor of Science in Electrical Engineering from Washington University in St. Louis in 2000, again with a grade point average of 4.0/4.0, with my dissertation being on “Cell Design to Maximize Capacity in Cellular Code Division Multiple Access (CDMA) Networks.”

8. While a graduate student, from 1996 through 2000, I worked at MinMax Corporation in St. Louis, where I designed software packages that provided tools to flexibly allocate capacity in a CDMA communications network and maximize the number of subscribers. I also analyzed and simulated different audio compression schemes. I also validated the hardware architecture for an Asynchronous Transfer Mode (ATM) switch capable of channel group switching, as well as performed logical and timing simulations, and developed the hardware architecture for the ATM switch. I also worked with Teleware Corporation in Seoul, South Korea, where I designed and developed algorithms that were commercially deployed in a software package suite for analyzing the capacity in a CDMA network implementing the IS-95 standard to maximize the number of subscribers.

9. After obtaining my Doctor of Science degree, I worked as a Senior Systems Engineer at Comspace Corporation from October of 2000 to December of 2001. At Comspace, I designed and developed advanced data coding and modulation methods for improving the reliability and increasing the available data rates for cellular communications. I coded and simulated different encoding

schemes (including Turbo coding, Viterbi decoding, trellis coded modulation, and Reed-Muller codes) and modulation techniques using amplitude and phase characteristics and multi-level star constellations. This work further entailed the optimization of soft decision parameters and interleavers for additive white Gaussian and Rayleigh faded channels. In addition, I also extended the control and trunking of Logic Trunked Radio (LTR) to include one-to-one and one-to-many voice and data messaging.

10. In January of 2002, I joined the faculty of the University of New Orleans in Louisiana as an Assistant Professor in the Department of Electrical Engineering. While in this position, I designed and taught two new courses called “Computer Systems Design I and II.” I also developed a Computer Engineering Curriculum with a strong hardware-design emphasis, formed a wireless research group, and advised graduate and undergraduate students.

11. In September of 2002, I received an appointment as an Assistant Professor in the Department of Computer Science and Engineering at the University of North Texas (UNT), in Denton, Texas. In May of 2008, I became a tenured Associate Professor in the Department of Computer Science and Engineering. As a faculty member, I have taught courses and directed research in networking and telecommunications, including 2G, 3G, 4G, 5G, CDMA/WCDMA, GPS, GSM, UMTS, LTE, ad-hoc networks, Bluetooth, call admission control, channel coding,

communication interfaces and standards, compression, computer architecture, MIMO systems, multi-cell network optimization, network security, packet-networks, telephony, VoIP, Wi-Fi (IEEE 802.11), IEEE 802.15.4, Zigbee, wireless communication, and wireless sensors. I am also the director of the Wireless Sensor Lab (“WiSL”) at UNT. I am a member of the Center for Information and Cyber Security (CICS). It is the only program in the U.S. to be federally certified by the National Security Agency as a Center of Academic Excellence in Information Assurance Education and Research *and* Cyber Defense Research. I am also a member of the NSF Net-Centric & Cloud Software & Systems: Industry-University Cooperative Research Center (I/UCRC). Several of my research projects are funded by industry. In January of 2015, I was promoted to Associate Chair of Graduate Studies in the Department of Computer Science and Engineering.

12. In addition to advising and mentoring students at UNT, I was asked to join the faculty of the University of Arkansas in Little Rock as an Adjunct Assistant Professor from 2004 to 2008 in order to supervise the research of two Ph.D. graduate students who were doing research in wireless communications. At UNT, I have advised and supervised more than 250 undergraduate and graduate students, several of whom received a master’s or doctorate degree under my guidance.

13. Further, since 2005, I have received over a million dollars in funding from the State of Texas, Texas Higher Education Coordination Board, the National

Science Foundation, and industry to design and conduct robotics, video, and mobile gaming (*e.g.*, Xbox, PC, mobile device) programming summer camps for middle and high school students at UNT. By using video and mobile gaming as the backdrop, participants have learned coding and programming principles and developed an understanding of the role of physics and mathematics in video game design.

14. In addition to my academic work, I have remained active in the communication industry through my consulting work. In 2002, I consulted for Input/Output Inc. and designed and implemented algorithms for optimizing the frequency selection process used by sonar for scanning the bottom of the ocean. In 2004, I worked with Allegiant Integrated Solutions in Ft. Worth, Texas, to design and develop an integrated set of tools for fast deployment of wireless networks, using the IEEE 802.11 standard. Among other features, these tools optimize the placement of Access Points and determine their respective channel allocations to minimize interference and maximize capacity. I also assisted the Collin County Sheriff's Office (Texas) in a double homicide investigation, analyzing cellular record data to determine user location.

15. I have authored and co-authored over 90 journal publications, conference proceedings, technical papers, book chapters, and technical presentations in a broad array of communications-related technologies, including networking and

wireless communication. I have also developed and taught over 100 courses related to communications and computer systems, including several courses on signals and systems, 4G/LTE and 5G/NR, OFDM, VoIP, Wi-Fi (IEEE 802.11), IEEE 802.15.4, Zigbee, wireless communication, communications systems, communication interfaces and standards, sensor networks, source coding and compression, network security, computer systems design, game and app design, and computer architecture. These courses have included introductory courses on communication networks and signals and systems, as well as more advanced courses on wireless communications. A complete list of my publications and the courses I have developed and/or taught is also contained in my *curriculum vitae*.

16. My professional affiliations include services in various professional organizations and serving as a reviewer for a number of technical publications, journals, and conferences. I have also received a number of awards and recognitions, including the IEEE Professionalism Award (2008), UNT College of Engineering Outstanding Teacher Award (2008), and Tech Titan of the Future (2010) among others, which are listed in my *curriculum vitae*.

17. I have also served as an expert in certain legal proceedings. A list of cases in which I have testified at trial, hearing, or by deposition (including those during the past five years) is provided in my *curriculum vitae*. Over the years, I have been retained by both patent owners and petitioners.

III. Level of Ordinary Skill in the Art

18. In rendering the opinions set forth in this declaration, I was asked to consider the patent claims and the prior art through the eyes of one of ordinary skill at the time of the alleged invention, which I understand is asserted to be March 12, 2013—the filing date of the earliest application in the patent family. I understand that the factors considered in determining the ordinary level of skill in a field of art include the level of education and experience of persons working in the field; the types of problems encountered in the field; the teachings of the prior art, and the sophistication of the technology at the time of the alleged invention. I understand that one of ordinary skill is not a specific real individual, but rather is a hypothetical individual having the qualities reflected by the factors above. I understand that one of ordinary skill would also have knowledge from the teachings of the prior art, including the art cited below.

19. I have been asked to assume the definition of a person of ordinary skill in the art adopted by the Board in the related IPR proceeding for U.S. Patent No. 9,124,552. Specifically, I understand that the Board found that on or before March 12, 2013, one of ordinary skill relating to the technology of the '552 Patent (and thus the '343 Patent) would have had working knowledge of packet-switched networking, firewalls, security policies, communication protocols and layers, and the use of customized rules to address cyber-attacks. In addition, one of ordinary skill in the

art would have had a bachelor's degree in computer science, computer engineering, or an equivalent, and four years of industry experience. Lack of work experience can be remedied by additional education, and vice versa.

20. Before March 12, 2013, my level of skill in the art was at least that of one of ordinary skill. I am qualified to provide opinions concerning what one of ordinary skill would have known and understood at that time, and my analysis and conclusions herein are from the perspective of one of ordinary skill as of that date.

IV. Materials Considered and Relied Upon

21. In reaching the conclusions described in this declaration, I have relied on the documents and materials cited herein, including the following documents:

Exhibit ("Ex.")	Description
1001	U.S. Patent No. 10,567,343 (the "'343 Patent")
1002	File History for U.S. Patent No. 10,567,343
1004	Sourcefire 3D System User Guide Version 4.10 ("Sourcefire")
1005	Declaration of John Leone
1006	U.S. Pat. No. 7,032,031 ("Jungck")
1007	U.S. Pat. App. Publication 2009/0103524 ("Mantripragada")
1008	U.S. Pat. No. 8,370,936 ("Zuk")
1009	U.S. Pat. No. 7,900,042 ("Hall")
1010	SSL Inspection – Sourcefire Cybersecurity, archived at Internet Archive Wayback Machine on December 20, 2010 at https://web.archive.org/web/20101220151851/http://www.sourcefire.com/security-technologies/cyber-security-products/3d-system/sslencryption-decryption ("SSL Appliances Webpage").
1011	Sourcefire SSL Appliances data sheet, archived at Internet Archive Wayback Machine on March 26, 2012 at https://web.archive.org/web/20120316165725/http://www.ndm.net:80/sourcefirestore/downloads/Sourcefire%20SSL%20Appliance%20Fact

Exhibit ("Ex.")	Description
	%20Sheet%20%28English%29.pdf ("SSL Appliances Data Sheet").
1012	U.S. Patent No. 7,586,899 ("Mohaban")
1013	Bard, G.V., "Cryptology ePrint Archive: Report 2006/136: A Challenging but Feasible Blockwise-Adaptive Chosen-Plaintext Attack on SSL," archived at Internet Archive Wayback Machine on May 3, 2006 at https://web.archive.org/web/20060503043443/http://eprint.iacr.org:80/2006/136 .
1014	Bard, G.V., "A Challenging but Feasible Blockwise-Adaptive Chosen-Plaintext Attack on SSL," archived at Internet Archive Wayback Machine on May 3, 2006 at https://web.archive.org/web/20060503043443/http://eprint.iacr.org:80/2006/136
1015	T. Dierks & E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.1," <i>IETF RFC 4346</i> , April 2006, archived at Internet Archive Wayback Machine on June 26, 2006 at https://web.archive.org/web/20060626134135if_/http://www.ietf.org:80/rfc/rfc4346.txt .
1016	S. Turner & C. Polk, "Prohibiting Secure Sockets Layer (SSL) Version 2.0," <i>IETF RFC 6176</i> , March 2011, archived at Internet Archive Wayback Machine on March 18, 2011 at https://web.archive.org/web/20110318184421/https://tools.ietf.org/html/rfc6176 .
1017	NDM Company ("Sourcefire Authorized Distributor") Product Data Sheets Webpage, archived at Internet Archive Wayback Machine on October 28, 2011 at https://web.archive.org/web/20111028104123/http://www.ndm.net:80/sourcefirestore/Product-Data-Sheets/product-data-sheets ("NDM Webpage").
1018	A. Shostack, An Overview of SSL (version 2), May 1995, available from https://adam.shostack.org/ssl.html (accessed July 11, 2018), archived at Internet Archive Wayback Machine on May 25, 1997 at https://web.archive.org/web/19970525224610/http://www.homeport.org/~adam/ssl.html .
1019	U.S. Patent No. 9,055,093 ("Borders")
1020	Barry M. Leiner <i>et al.</i> , "A Brief History of the Internet". <i>Internet Society</i> , 1997, available at:

Exhibit ("Ex.")	Description
	https://www.internetsociety.org/internet/history-internet/brief-historyinternet/.
1021	"Internet Protocol," <i>IETF RFC 791</i> , J. Postel, ed., September 1981, archived at Internet Archive Wayback Machine on February 4, 2007 at https://web.archive.org/web/20070204151303/https://tools.ietf.org/html/rfc791 .
1022	"Transmission Control Protocol," <i>IETF RFC 793</i> . J. Postel, ed., September 1981, archived at Internet Archive Wayback Machine on February 2, 2007 at https://web.archive.org/web/20070202201546/https://tools.ietf.org/html/rfc793 .
1023	E. H. Spafford, "The Internet Worm Program: An Analysis," <i>Purdue Technical Report CSD-TR-823</i> , 1988.
1024	Eichin, M.W. and Rochlis, J.A., "With Microscope and Tweezers: An Analysis of the Internet Virus of November 1988," <i>Proceedings, 1989 IEEE Symposium on Security and Privacy</i> , (pp. 326-343), IEEE, 1989.
1025	T. Eisenberg <i>et al.</i> , "The Cornell Commission: On Morris and the Worm," <i>Communications of the ACM</i> , Volume 32 Number 6, June 1989.
1026	J. C. Mogul, "Simple and Flexible Datagram Access Controls for Unix-based Gateways," <i>Digital Western Research Laboratory Technical Report No. 89/4</i> , 1989.
1027	Heberlein, L. T.L, <i>et al.</i> , "A Network Security Monitor," <i>Proceedings, 1990 IEEE Computer Society Symposium on Security and Privacy</i> , 1990.
1028	"NetRanger Network Security Management System User Guide". v1.3.1 <i>Wheelgroup Corporation</i> , 1997.
1029	Roesch, M., "Snort: Lightweight Intrusion Detection for Networks," <i>Lisa</i> (Vol. 99, No. 1, pp. 229-238), 1999.
1030	Brandon De Hoyos, "IRC, ICQ, AIM and More: A History of Instant Messaging," <i>Lifewire</i> , Oct 2016, available from https://www.lifewire.com/im-a-brief-history-1949611 .
1031	M. Chatel, "Classical vs. Transparent IP Proxies," <i>IETF RFC 1919</i> , March 1996.
1032	K. Ingham & S. Forrest, "A History and Survey of Network Firewalls," <i>University of New Mexico Computer Science Department Technical Report 2002-37</i> , 2002.

Exhibit ("Ex.")	Description
1033	Sourcefire Support Website Webpage, archived at Internet Archive Wayback Machine on April 30, 2011 at https://web.archive.org/web/20110430194135/https://support.sourcefire.com/ ("Sourcefire Support Webpage").
1034	Business Wire, "Sourcefire Expands IPS Solutions Portfolio, Adding FirePOWER to the Industry's Leading Protection," April 18, 2011, available at https://www.businesswire.com/news/home/20110418005189/en/Sourcefire-Expands-IPS-Solutions-Portfolio-Adding-FirePOWER (accessed June 20, 2018)
1035	"File Transfer Protocol," IETF RFC 765, J. Postel, ed., June 1980, archived at Internet Archive Wayback Machine on February 6, 1997 at https://web.archive.org/web/20070206005843/https://tools.ietf.org/html/rfc765 .
1036	T. B. Lee, "The Original HTTP as defined in 1991," World Wide Web Consortium, 1991, archived at Internet Archive Wayback Machine on June 5, 1997 at https://web.archive.org/web/19970605071155/https://www.w3.org/Protocols/HTTP/AsImplemented.html .
1037	I. Ristic, "SSL/TLS and PKI History," April 2018, available from https://www.feistyduck.com/ssl-tls-and-pki-history/ .
1038	E. Rescorla, "HTTP over TLS", <i>IETF RFC 2818</i> , May 2000, archived at Internet Archive Wayback Machine on January 25, 2007 at https://web.archive.org/web/20070125003236/https://tools.ietf.org/html/rfc2818 .
1039	J. Moser, "The First Few Milliseconds of an HTTPS Connection", <i>Moserware</i> , June 10, 2009, archived at Internet Archive Wayback Machine on June 13, 2009 at https://web.archive.org/web/20090613041022/http://www.moserware.com/2009/06/first-few-milliseconds-of-https.html .
1040	U.S. Patent No. 9,124,552 (the "'552 Patent")
1042	IPR2018-01436, Pap. 40 (Jan. 23, 2020) (Final Written Decision for U.S. Patent No. 9,124,552)
1043	<i>Centripetal Networks, Inc. v. Cisco Sys., Inc.</i> , 847 F. App'x 869 (Fed. Cir. 2021)
1044	Chart Comparing the Challenged Claims to the Claims of U.S. Patent No. 9,124,552

22. These materials comprise patents, related documents, and printed publications. Each of these materials is a type of document that experts in my field would have reasonably relied upon when forming their opinions.

23. I have also relied on my education, training, research, knowledge, and personal and professional experience in the relevant technologies and systems that were already in use prior to, and within the timeframe of the earliest asserted priority date of the claimed subject matter in the '343 Patent, which is March 12, 2013.

V. Legal Standards

24. I am not a lawyer and do not provide any legal opinions, but I have been advised that certain legal standards are to be applied by technical experts in forming opinions regarding meaning and validity of patent claims. I have applied the legal standards described below, which were provided to me by counsel for Petitioner.

25. I have been informed that there are two ways in which prior art may render a patent claim unpatentable. First, the prior art can be shown to “anticipate” the claim. Second, the prior art can be shown to have made the claim “obvious” to a person of ordinary skill in the art.

A. Anticipation

26. I have been informed that if each and every element or step of a claim is disclosed within the “four corners” of a prior art reference, that claim is said to be “anticipated” by that single prior art reference and is unpatentable under 35 U.S.C.

§ 102 because the claimed invention is not, in fact, new or novel. I have been informed that the standard for anticipation in an *inter partes* review proceeding is by a preponderance of the evidence.

27. I have also been informed that a prior art reference can disclose a claim feature if that feature is expressly described by that reference, or inherent from its disclosure. I have been informed that something is inherent from a prior art reference, if the missing descriptive matter must necessarily be present, and it would be so recognized by a person of ordinary skill in the art. I have also been informed that inherency cannot be established by probabilities or possibilities, and that the mere fact that something may result from a given set of circumstances is not sufficient to show inherency.

28. I have further been informed that where a reference discloses multiple embodiments, the reference should not be limited to a preferred embodiment. Instead, each disclosed embodiment may anticipate.

29. Moreover, I have been informed that as part of an anticipation analysis, it is proper to take into account not only specific teachings of the reference, but also the inferences that one skilled in the art would reasonably be expected to draw therefrom. A reference can anticipate a claim even if it does not expressly spell out all the limitations arranged or combined as in the claim, if a person of ordinary skill

in the art, reading the reference, would at once envisage the claimed arrangement or combination.

30. I have been informed that a prior art document can disclose a claim feature, and anticipate a claimed invention, if that feature is described in another document that has been incorporated by reference. I have also been informed that, to incorporate by reference, the host document must identify with detailed particularity what specific material it incorporates, and clearly indicate where that material is found in the incorporated document. I have also been informed that, in making the determination of the extent to which material is incorporated into a host document, the standard of a person of ordinary skill in the art should be used to determine whether the host document describes the material to be incorporated by reference with sufficient particularity.

B. Obviousness

31. I have been informed that a claim may be unpatentable under 35 U.S.C. § 103(a) if the subject matter described by the claim as a whole would have been obvious to a hypothetical person of ordinary skill in the art in view of a prior art reference or in view of a combination of references at the time the claimed invention was made. I have been informed that obviousness is determined from the perspective of a hypothetical person of ordinary skill in the art and that the asserted claims of the patent should be read from the point of view of such a person at the

time the claimed invention was made. I have been informed that a hypothetical person of ordinary skill in the art is assumed to know and to have all relevant prior art in the field of endeavor covered by the patent in suit.

32. I have been informed that there are two criteria for determining whether prior art is analogous and thus can be considered prior art: (1) whether the art is from the same field of endeavor, regardless of the problem addressed, and (2) if the art is not within the field of the patentee's endeavor, whether the teachings of the art are still reasonably pertinent to the particular problem with which the patentee is involved. I have also been informed that the field of endeavor of a patent is not limited to the specific point of novelty, the narrowest possible conception of the field, or the particular focus within a given field. I have also been informed that a prior art reference is reasonably pertinent if, even though it may be in a different field from that of the patentee's endeavor, it is one which, because of the matter with which it deals, logically would have commended itself to a patentee's attention in considering his problem.

33. I have also been informed that an analysis of whether a claimed invention would have been obvious should be considered in light of the scope and content of the prior art, the differences (if any) between the prior art and the claimed invention, and the level of ordinary skill in the pertinent art involved. I have been informed as well that a prior art reference should be viewed as a whole.

34. I have also been informed that in considering whether an invention for a claimed combination would have been obvious, I may assess whether there are apparent reasons to combine known elements in the prior art in the manner claimed in view of interrelated teachings of multiple prior art references, the effects of demands known to the design community or present in the market place, and/or the background knowledge possessed by a person having ordinary skill in the art. I have been informed that other principles may be relied on in evaluating whether a claimed invention would have been obvious, and that these principles include the following:

- A combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results;
- When a device or technology is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or in a different one, so that if a person of ordinary skill can implement a predictable variation, the variation is likely obvious;
- If a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill;

- An explicit or implicit teaching, suggestion, or motivation to combine two prior art references to form the claimed combination may demonstrate obviousness, but proof of obviousness does not depend on or require showing a teaching, suggestion, or motivation to combine;
- Market demand, rather than scientific literature, can drive design trends and may show obviousness;
- In determining whether the subject matter of a patent claim would have been obvious, neither the particular motivation nor the avowed purpose of the named inventor controls;
- One of the ways in which a patent's subject can be proved obvious is by noting that there existed at the time of invention a known problem for which there was an obvious solution encompassed by the patent's claims;
- Any need or problem known in the field of endeavor at the time of invention and addressed by the patent can provide a reason for combining the elements in the manner claimed;
- "Common sense" teaches that familiar items may have obvious uses beyond their primary purposes, and in many cases a person of

ordinary skill will be able to fit the teachings of multiple patents together like pieces of a puzzle;

- A person of ordinary skill in the art is also a person of ordinary creativity, and is not an automaton;
- A patent claim can be proved obvious by showing that the claimed combination of elements was “obvious to try,” particularly when there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions such that a person of ordinary skill in the art would have had good reason to pursue the known options within his or her technical grasp; and
- One should be cautious of using hindsight in evaluating whether a claimed invention would have been obvious.

35. I have further been informed that, in making a determination as to whether or not the claimed invention would have been obvious to a person of ordinary skill, the Board may consider certain objective factors if they are present, such as: commercial success of products practicing the claimed invention; long-felt but unsolved need; teaching away; unexpected results; copying; and praise by others in the field. These factors are generally referred to as “secondary considerations” or “objective indicia” of nonobviousness. I have been informed, however, that for such objective evidence to be relevant to the obviousness of a claim, there must be a

causal relationship (called a “nexus”) between the claim and the evidence, and that this nexus must be based on what is claimed and novel in the claim rather than something in the prior art. I have also been informed that even when they are present, secondary considerations may be unable to overcome primary evidence of obviousness (*e.g.*, motivation to combine with predictable results) that is sufficiently strong.

C. Claim Construction

36. I have been informed that patent claims are construed from the viewpoint of a person of ordinary skill in the art of the patent at the time of the invention. I have been informed that patent claims generally should be interpreted consistent with their plain and ordinary meaning as understood by a person of ordinary skill in the art in the relevant time period (*i.e.*, at the time of the purported invention, or the so called “effective filing date” of the patent application), after reviewing the patent claim language, the specification and the prosecution history (*i.e.*, the intrinsic record).

37. I have further been informed that a person of ordinary skill in the art must read the claim terms in the context of the claim itself, as well as in the context of the entire patent specification. I understand that in the specification and prosecution history, the patentee may specifically define a claim term in a way that differs from the plain and ordinary meaning. I understand that the prosecution

history of the patent is a record of the proceedings before the U.S. Patent and Trademark Office, and may contain explicit representations or definitions made during prosecution that affect the scope of the patent claims. I understand that an applicant may, during the course of prosecuting the patent application, limit the scope of the claims to overcome prior art or to overcome an examiner's rejection, by clearly and unambiguously arguing to overcome or distinguish a prior art reference, or to clearly and unambiguously disavow claim coverage.

38. In interpreting the meaning of the claim language, I understand that a person of ordinary skill in the art may also consider “extrinsic” evidence, including expert testimony, inventor testimony, dictionaries, technical treatises, other patents, and scholarly publications. I understand this evidence is considered to ensure that a claim is construed in a way that is consistent with the understanding of those of ordinary skill in the art at the time of the claimed invention. This can be useful for technical terms whose meaning may differ from its ordinary English meaning. I understand that extrinsic evidence may not be relied on if it contradicts or varies the meaning of claim language provided by the intrinsic evidence, particularly if the applicant has explicitly defined a term in the intrinsic record.

VI. Background on the State of the Art

39. The following section describes the historical development of the art for network security systems, including firewalls and other packet filtering devices

for protecting computer networks from cyber-attacks, leading up to the state of the art as of March 2013. The prior art references, and the discussions of what was known to a person of ordinary skill in the art, provide the factual support for the general description of the state of the art at the time of the '343 Patent's earliest priority date. This section provides additional motivation to combine the teachings of the Sourcefire reference (Ex. 1004) with the knowledge, skill, and creativity of a person of ordinary skill in the art, provides support for why a person of ordinary skill in the art would have had a reasonable expectation of success in doing so, rebuts any assertion by the Patent Owner of unpredictability in the art, and rebuts any assertion by the Patent Owner of unexpected results.

40. This overview is based on the "State of the Art" summary provided by Dr. Stuart Staniford in his declaration in support of the petition for IPR of related U.S. Patent No. 9,124,552 (IPR2018-1436), filed by Cisco System, Inc. The '343 shares a common specification with the '552 Patent, and the claims themselves are directed to the same basic concepts. For that reason, the state of the art relevant to the '552 Patent is necessarily also relevant to the '343 Patent. Any similarity between Dr. Staniford's "State of the Art" overview and my own is intentional. I reviewed his opinions on the state of the art and the supporting documents cited therein, and agree with those opinions as set forth herein.

41. As of March 2013, a person of ordinary skill in the art working in the

field of network security systems, including firewalls and other packet filtering devices for protecting computer networks from cyber-attacks, would have been familiar with certain common concepts and related components including, but not limited to: packets-switched networking, firewalls, security policies, communication protocols and layers, and the use of customized rules to address cyber-attacks.

A. Packet-Switched Networking and Communication Protocols

42. The Internet has its roots in conceptual papers on packet switching in the 1960s. These papers led to the funding by DARPA of an experimental internetwork—the ARPANET—with the first two nodes being connected in late 1969, and then growing steadily through the 1970s as a means of doing research on computer networking. Ex. 1020. The early ARPANET used a protocol, NCP, that is no longer in use and proved not to be scalable.

43. NCP’s deficiencies led to the development of two new protocols—Internet Protocol (IP) to provide best effort delivery of data packets across heterogeneous networks (Ex. 1022), and Transmission Control Protocol (TCP) to provide reliable delivery of streams of data across an IP internetwork (Ex. 1021). These two protocols were described in the IETF Request For Comment (RFC) documents RFC 791 and 793 respectively in 1981. Ex. 1022; Ex. 1021. The ARPANET switched to using the new TCP/IP protocol suite on January 1, 1983. That date can be viewed as the birth date of the modern Internet. Ex. 1020.

44. Logically, the Internet is a web of intermediate network nodes and communication paths interconnecting those network nodes that provide multiple data transmission routes from any given point to any other given point on the network (*i.e.*, between any two computers). Ex. 1006, 3:59-4:11. The disparate networks are connected where they are joined by special devices—IP routers (often known in early literature as gateways)—that decide which way to send each packet. A router typically creates or maintains a table of the available routes and their conditions and uses this information to determine the best route for a given packet. Ex. 1006, 3:32-46, 38:29-39:8.

45. While TCP/IP provided the basic abstract facility to move data across a global network, higher-level application protocols were required to allow data communication to be managed conveniently. Application protocol data would be transmitted in the body of packets with IP and TCP headers. Particularly important applications included email (with protocols such as SMTP, POP and IMAP), with early email systems dating back to 1972. Ex. 1020. Other important protocols included FTP for making files available, first defined in 1980 (Ex. 1035), and the Hypertext Transfer Protocol (HTTP) for facilitating the exchange of hypertext markup language (HTML) and World Wide Web (WWW) content, first defined in 1991 (Ex. 1036). Similarly, Telnet and then SSH were developed for remote computer terminal access, and a variety of instant messaging protocols became

available for text-based communication (*e.g.*, AOL Instant Messenger (AIM) and ICQ, Windows Live Messenger, Yahoo! Messenger, Jabber using XMPP protocol, etc.). Ex. 1030.

B. Network Security Systems—Firewalls

46. Throughout the mid to late 1980s, the Internet grew very rapidly as more and more researchers and developers connected departmental mainframes, minicomputers, Unix workstations, and the like to it. The Internet reflected the openness and information sharing that characterizes the research community that developed it, with few security barriers beyond the requirement to login on remote systems (which often had weak or default passwords). Ex. 1020.

47. This era of complete openness spawned a variety of abuses by computer hackers. A particularly famous incident spawned the modern network security industry: the Internet worm of 1988. In this incident, a Cornell graduate student, Robert Morris Jr., released a worm—a self-replicating program capable of infecting other computers across the network via vulnerabilities in their operating system or applications. This worm infected about 6,000 computers across the Internet. Most of these computers were disabled, either by large numbers of copies of the worm clogging their memory, or at least by the need to disinfect and upgrade them. Many of the computers were important shared facilities with numerous users, so much of the work on the Internet was disrupted. The incident caused a great deal of analysis

and soul-searching amongst the Internet community. Within a year (by 1989), both the first firewall and the first network intrusion detection system (NIDS or IDS) were under development. Ex. 1023; Ex. 1024; Ex. 1025.

48. The development of firewalls had its roots in the late 1980s practice of using filtering rules on Internet routers. This development began primarily as a way to prevent misconfigurations on one network that generated a lot of noise packets from affecting an adjacent network. However, it then resulted in the development of security applications.

49. It was well-known to include firewall functionality in some routers to intercept and review packets for security purposes. Indeed, routers provided simple facilities to block packets based on their source and destination networks and TCP ports. For example, with Cisco routers in 1989: “one could deny access to all but one host on a particular network. The Cisco gateways also allow discrimination based on IP protocol type, and TCP or UDP port numbers.” Ex. 1026, 3. This example illustrates that the basic functionality of blocking or admitting packets based on a 5-tuple was available prior to 1989.

50. From early on, it was apparent that some computer applications required special handling. Thus, for example, the first commercial firewall, the DEC SEAL released in 1991, used a packet filter to control what packets could enter or leave the network but also required most application traffic to transit an explicit

application proxy server. Ex. 1032, 9. Here, an internal network client desiring to access exterior network resources would make a TCP connection to the application server (and the packet filter would be configured to allow this, but not direct outbound connections). The client would then authenticate to the application proxy, permissions would be checked and, if authorization was granted, the client connection was continued, and the proxy would make a second TCP connection to the external server. The proxy would then relay requests/responses in the application back and forth, while logging and possibly filtering the traffic. Ex. 1032, 8-11.

51. Early packet filters reasoned packet-by-packet without keeping any state information about prior packets. It quickly became apparent that this strategy was both expensive (in that possibly large firewall rulesets had to be separately processed for every packet, even in a long connection) and had security weaknesses (as hackers found ways to fool packet filters into allowing probe packets through because the filter did not know the state of the TCP connection). This led to the development of stateful inspection firewalls. Early stateful inspection firewalls applied the rules to the first packet in a TCP connection (the “syn packet”), and then kept track in an in-memory hash table of allowed connections and their current state, meaning that subsequent packets in the connection could be allowed or blocked based on a single hash lookup. Ex. 1032; Ex. 1008, 4:12-30.

52. By the mid-1990s, multiple systems had been developed which

incorporated application functionality “transparently” into stateful inspection firewalls. Ex. 1031 (published in March 1996). The way this worked is that the firewall would parse the application data inside packets it was passing, and investigate whether they might be significant to application traffic routing. For example, the Zuk patent, filed in 2002, stated that:

A packet filtering firewall with stateful inspection also has the ability to examine a packet in order to allow certain types of commands within an application while disallowing others. For example, a stateful inspection firewall can allow the FTP “get” command while disallowing the “put” command.

Ex. 1008, 4:31-41.

53. Based on understanding the application protocol, the firewall determined that certain TCP connections needed to be routed through a proxy. It would instantiate the relevant proxy code in its own memory and then connect the incoming client TCP connection to that proxy automatically, which would then perform authentication and provide proxy service on to the server. Thus, no later than 1996, a person of ordinary skill in the art familiar with then current firewall technology would have known that some firewalls were discriminating whether to pass or block packets based on examining both the 5-tuple (*e.g.*, TCP/IP headers), but also the application headers in the packet data bodies.

C. Network Security Systems—Intrusion Detection Systems and Intrusion Prevention Systems

54. In an independent line of development from firewalls, the Internet worm and other less serious incidents of the time inspired researchers to begin exploring systems specifically designed to look for intrusions on networks by examining the data packets crossing the network. Initially, network intrusion detection was done by placing network interfaces on broadcast networks into “promiscuous” mode to inspect all the traffic, which could then be copied into a user-mode application where it could be inspected and reasoned about. The earliest such system was the Network Security Monitor (NSM) developed at UC Davis in 1991. Ex. 1027. This evolved into the Air Force’s ASIM system, and a large part of the development team on that effort split off and formed a company, WheelGroup, which launched the first commercial network intrusion detection system (IDS) NetRanger in 1996.

55. NetRanger had a general facility to write rules that matched arbitrary strings in the application data inside packets, in addition to matching on IP and TCP packet header values. For example, the 1997 user guide for the NetRanger system illustrated how to log every HTTP GET request passing by in packets on the monitored network:

The following example shows how to log every HTTP GET on a network. The context info provided with the string will show what URL was used in the GET.

Web	SIGID	Port	Direction	Occurrences	String Match
RecordOfStringName	8001	80	1	1	"GET[]+"

Ex. 1028, 4-80.

56. Thus, from the beginning of the network intrusion detection industry, it was known to a person of ordinary skill in the art to look at application header data, and HTTP methods in particular, when examining packets transiting a network. Ex. 1028, 4-80.

57. The first widely adopted open-source network intrusion detection system (NIDS or just IDS), Snort, was first released in 1998, and was documented in a conference paper in 1999. Ex. 1029. Like other NIDS before it, Snort allowed the user to write rules for detecting particular network packets based on both the 5-tuple information in the packet IP and TCP headers, as well as strings found in packet application data. For example:

```
alert tcp any any -> 10.1.1.0/24 80 (content: "/cgi-bin/phf"; msg: "PHF probe!";)
```

58. This rule (see Ex. 1029, 232, Fig 5) shows that alerting was to be done based on TCP protocol packets from any source IP address and port, and destination IP addresses in the range 10.1.1.0/24, and with destination port 80 (characteristic of HTTP traffic). This particular content keyword (/cgi-bin/phf) matches a uniform

resource locator (URL) used in a popular attack of the time, but content keywords in general could be used to find and match any text in application data in packets transiting the network.

59. The earliest NIDS were strictly involved in detecting attacks; these systems worked off a copy of the packets and could only alert or log suspected intrusions, not directly block them. However, it was quickly realized that customers had a need to take automated action in some cases. Already by 1997, WheelGroup's NetRanger system, for example, could be used to reconfigure a packet filtering router to block all packets from the source IP address used in a detected attack. Ex. 1028, 4-26.

60. The need to block attacks, as well as the need to resolve various complex ambiguities in the meaning of packet traffic that computer intruders used to get around the first NIDS, meant that the industry quickly developed systems that were "inline" in the traffic flow, and which could choose to drop or pass any given packet. These systems were known as intrusion prevention systems (IPS), with multiple companies launching such products in 2002, including Intruvert (later acquired by McAfee), Tipping Point (3Com then HP), and OneSecure (later acquired by NetScreen then Juniper).

61. These inline intrusion prevention systems were capable of applying intrusion detection rules that looked both at 5-tuple header data and application data

(*e.g.*, content) in every packet and making pass/alert/block decisions on each such packet. For example, the Zuk patent, filed in 2002, described an IPS architecture in which packets were inspected, and the 5-tuple in the packet header was used to lookup the correct rules in a flow table (hash table of all TCP connections or UDP pseudo-connections currently being tracked by the system). The rules described in Zuk then allowed regular expression matching of any packet content in the application layer, and packet application data content matches would result in triggering the rule and dropping the packet. Ex. 1008, col. 18, Fig. 11.

D. Inspecting and Applying Packet Filtering Rules to Encrypted and Unencrypted SSL/TLS Record Protocol Data

62. The early versions of HTTP transmitted their data unencrypted (“in the clear”), leaving them vulnerable to eavesdropping and the like. To address this vulnerability, in 1995, Netscape Corporation released an implementation of the Secure Sockets Layer (SSL) protocol, which inserted an encrypted channel running on top of TCP, and underneath HTTP. The first public version of this SSL protocol, version 2, had security issues,¹ and was rapidly replaced with version 3 in 1996. The

¹ By March of 2011, the security issues with SSL v2 were so well-known that the IETF issued RFC 6176 which “require[d] that when Transport Layer Security (TLS) clients and servers establish connections, they never negotiate the connection of Secure Sockets Layer (SSL) version 2.0.” Ex. 1016.

IETF then became involved in standardizing these encryption protocols, and released TLS 1.0 which was a minor improvement on SSL v3 in 1999. To address well-known security vulnerabilities in SSL v2, SSL v3, and TLS 1.0, the IETF released TLS 1.1 in 2006, and then TLS 1.2 in 2008. *See, e.g.*, Ex. 1013 (“The attack in this paper is an application of the blockwise-adaptive chosen-plaintext attack paradigm, and is the only feasible attack to use this paradigm with a reasonable probability of success. The attack will work for all versions of SSL, and TLS version 1.0. This vulnerability and others are closed in TLS 1.1 (which is still in draft status) . . . It is hoped this paper will encourage the deprecation of SSL and speed the adoption of . . . TLS 1.1/1.2 when they are finially [sic] released.”); Ex. 1014. In RFC 4346 (TLS 1.1), the IETF stated:

This document is a revision of the TLS 1.0 [TLS1.0] protocol, and contains some small security improvements. The major changes are:

- The implicit Initialization Vector (IV) is replaced with an explicit IV to protect against CBC attacks [CBCATT].
- Handling of padding errors is changed to use the bad_record_mac alert rather than the decryption_failed alert to protect against CBC attacks. . . .

Ex. 1015, 4-5; *see also id.*, 20.

[CBCATT] describes a chosen plaintext attack on TLS that depends on knowing the IV for a record. Previous versions of TLS [TLS1.0] used the CBC residue of the previous record as the IV and therefore enabled this attack. This version uses an explicit IV in order to protect against this attack.

Ex. 1015, 74.

63. TLS 1.2 was still the most up to date version of the protocol by March 2013. Ex. 1037.

64. As discussed above, the well-known SSL and TLS protocols insert an encrypted channel running on top of TCP, and underneath HTTP. Ex. 1038, 1. As was well understood prior to March 2013, the SSL/TLS protocols include a higher layer subprotocol sitting on top of the SSL/TLS Record Protocol. There are four different subprotocols depending on the stage of communication between an HTTP/TLS client and an HTTP/TLS server (Ex. 1015, 13; Ex. 1038, 3):

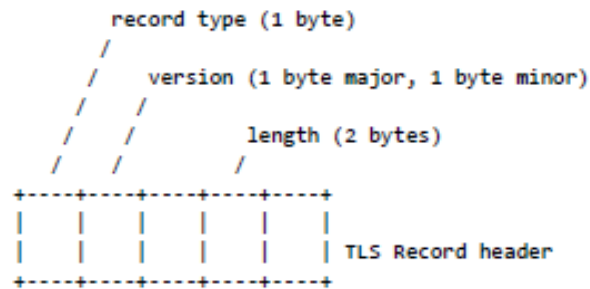
- Handshake subprotocol: allows the client and server to authenticate each other and to negotiate the cipher suite and other parameters of the connection (including SSL or TLS version). (Ex. 1015, 29-32)
- ChangeCipherSpec subprotocol: makes the previously negotiated parameters effective, so communication becomes encrypted. (Ex. 1015, 24)
- Alert subprotocol: used for communicating exceptions and to indicate potential problems that may compromise security. (Ex. 1015, 24-29.)
- Application Data subprotocol: takes arbitrary application-layer data (*e.g.*, HTTP data) generally, and feeds it through the secure channel. (Ex. 1015, 50)

65. The SSL/TLS Record Protocol (1) fragments the higher-layer subprotocol data into blocks of 214 bytes or less; then (2) optionally compresses each block of the subprotocol data, (3) adds a Message Authentication Code to each block, (4) if applicable (for Finished message Handshake subprotocol data, and ChangeCipherSpec, Alert, and Application Data subprotocol data), encrypts each

block according to the cipher spec, and (5) prepends an SSL/TLS Record header. Ex. 1015, 13, 16-19, 48.

66. As illustrated in the figure below, the SSL/TLS Record header added to every SSL/TLS “Record” (*i.e.*, the block formed using the above-described SSL/TLS Record protocol) includes the following three fields: (1) Byte 0: SSL/TLS Record type, (2) Bytes 1-2: SSL/TLS version (major/minor),² and (3) Bytes 3-4: Length of data in the Record (excluding the header itself):

² In SSL v2 Record Protocol, the SSL Record header has no SSL version (Ex. 1018) and a person of ordinary skill in the art familiar with the relevant art would thus know that the version could only be obtained from looking at the unencrypted body (*i.e.*, unencrypted SSL Record) of the SSL v2 handshake message packets. Ex. 1018.

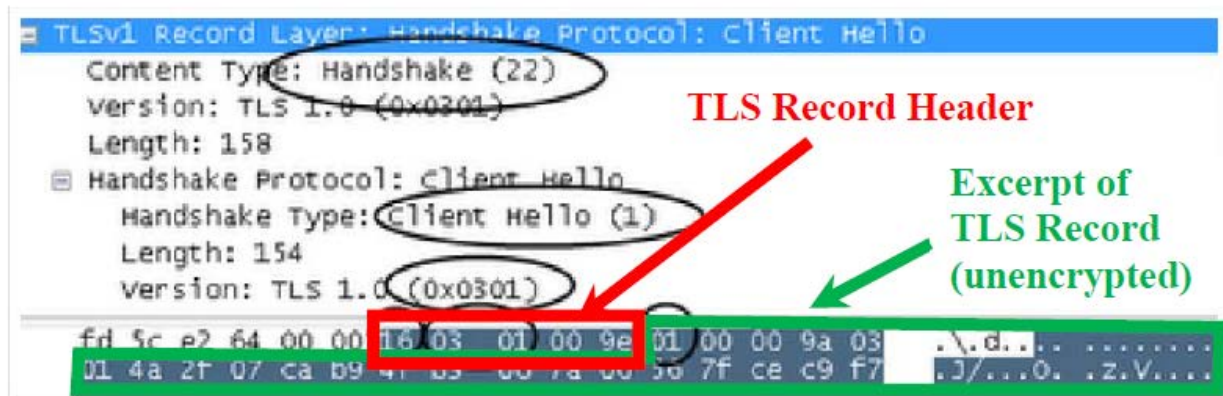


Record Type Values	dec	hex
CHANGE_CIPHER_SPEC	20	0x14
ALERT	21	0x15
HANDSHAKE	22	0x16
APPLICATION_DATA	23	0x17

Version Values	dec	hex
SSL 3.0	3,0	0x0300
TLS 1.0	3,1	0x0301
TLS 1.1	3,2	0x0302
TLS 1.2	3,3	0x0303

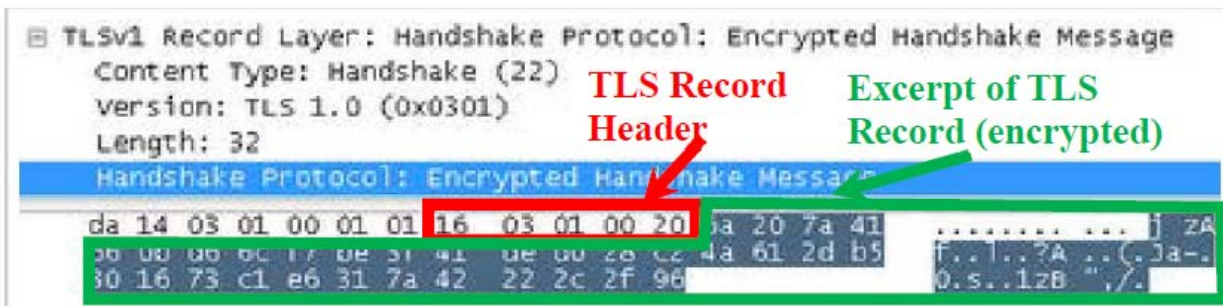
Ex. 1015, 16-17.

67. A person of ordinary skill in the art would have understood that if a user at the HTTP/TLS server (or at an intervening IDS or IPS) examined a TLS Record Protocol packet sent by a HTTP/TLS client requesting to establish a HTTPS session, the packet would include a TLS Record header and an unencrypted TLS Record for a “Client Hello” handshake subprotocol message as illustrated in the following example:



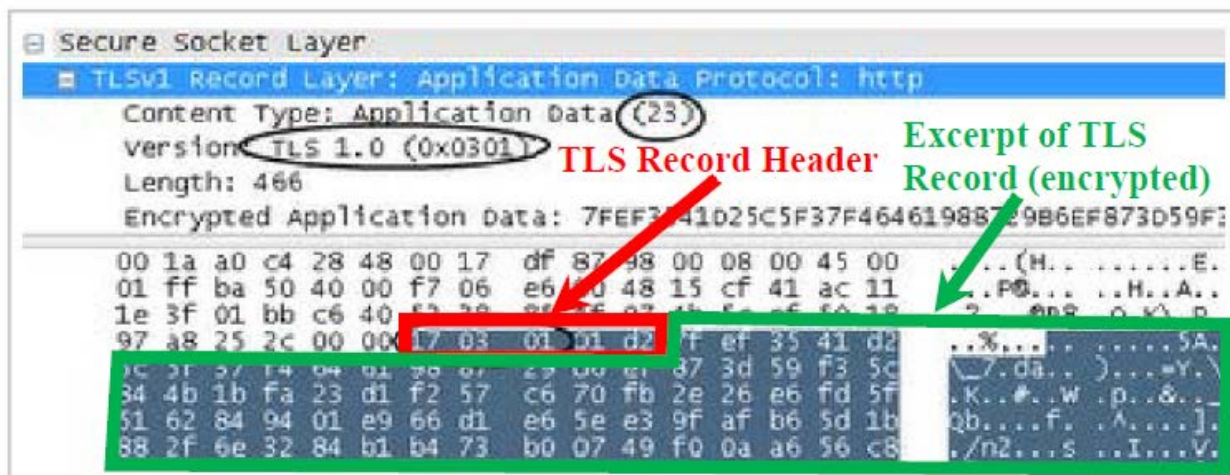
Ex. 1038, 3; Ex. 1039, 1-2. As shown in the above example, the TLS Record header specifies that the TLS Record type is Handshake (0x16 or 22), that the TLS version is TLS 1.0 (03x01 or 3,1), and that the length of unencrypted data in the TLS Record is 154 bytes (00x9e).

68. Similarly, a person of ordinary skill in the art would have understood that if a user at the HTTP/TLS server (or at an intervening IDS or IPS) examined another TLS Record Protocol packet subsequently sent by the HTTP/TLS client (and as the last message in the handshake subprotocol), this packet would include a TLS Record header and an encrypted TLS Record for a “Finished” handshake subprotocol message as illustrated in the following example:



Ex. 1038, 3; Ex. 1039, 11-12. As shown in the above example, the TLS Record header specifies that the TLS Record type is Handshake (0x16 or 22), that the TLS version is TLS 1.0 (03x01 or 3,1), and that the length of encrypted data in the TLS Record is 32 bytes (00x20).

69. Similarly, a person of ordinary skill in the art would have understood that if a user at the HTTP/TLS client (or at an intervening IDS or IPS) examined another TLS Record Protocol packet subsequently sent by the HTTP/TLS server (and as a message in the application data subprotocol), this packet would include a TLS Record header and an encrypted TLS Record for an application data subprotocol message as illustrated in the following example:



Ex. 1038, 3; Ex. 1039, 12-13. As shown in the above example, the TLS Record header specifies that the TLS Record type is Application Data (0x17 or 23), that the TLS version is TLS 1.0 (03x01 or 3,1), and that the length of encrypted data in the TLS Record is 466 bytes (01xd2).

70. As identified below, the '343 Patent describes “TLS Record Protocol packets” as “application packets contained in [] IP packets” and describes the “header fields of TLS Record Protocol packets” as being unencrypted and including “a value indicating the TLS version.” Ex. 1001, 8:14-19.

71. As encryption of HTTP traffic became more widespread, there was a need for ways to allow enterprise IDS/IPS devices to inspect the HTTPS traffic for decryption. The best known technique to inspect and apply rules to encrypted application layer data in HTTPS was to use an IDS or IPS (or device serving the IDS or IPS) to (1) serve as a proxy for a SSL/TLS handshake protocol between a client and a server, (2) apply packet filtering rules to the packet header data and application layer data in the handshake packets, (3) allow separate HTTPS sessions to be established between the client and proxy, and then between the proxy and server, (4) decrypt the application layer data in the HTTPS packets, and (5) inspect the application layer data and apply rules to either block or allow packets based on such rules, and then re-encrypt it for transmission to its eventual destination. *See e.g.*, Ex. 1009, Figs. 1, 2, 2:6-23, 3:33-37, 3:57-61, 4:25-30, 5:43-54, 5:64-6:3, 6:8-11.

72. For example, on its website³ and in its publicly accessible data sheet,⁴

³ SSL Appliances Webpage is a printout of a web page from the Sourcefire company's publicly available website <http://www.sourcefire.com/security-technologies/cyber-security-products/3d-system/ssl-encryption-decryption>. Ex. 1010. The webpage is available through the Internet Archive's Wayback Machine at <https://web.archive.org/web/20101220151851/http://www.sourcefire.com/securitytechnologies/cyber-security-products/3d-system/ssl-encryption-decryption> and was archived by the Internet Archive on December 20, 2010. Ex. 1010.

⁴ SSL Appliances Data Sheet (Ex. 1011) is a printout of the "Sourcefire SSL Appliance Fact Sheet" downloaded from a hyperlink of the same name on the NDM company's publicly available website <http://www.ndm.net:80/sourcefirestore/Product-Data-Sheets/product-data-sheets> ("NDM Webpage"). CS-1017; CS-1011. The webpage is available through the Internet Archive's Wayback Machine at <https://web.archive.org/web/20111028104123/http://www.ndm.net:80/sourcefirestore/Product-Data-Sheets/product-data-sheets> and was archived by the Internet Archive on October 28, 2011. Ex. 1017. I understand that the SSL Appliances Data Sheet (Ex. 1011) is available through the Internet Archive's Wayback Machine at

the Sourcefire company described a separate appliance (the “SSL Appliance”) that a network administrator could install and use to protect a network along with, for example, a 3D Sensor with IPS (See Section VIII.A below) so that the IPS could inspect and apply various rules to non-SSL and decrypted SSL packets to either block or allow such packets:

<https://web.archive.org/web/20120316165725/http://www.ndm.net:80/sourcefirestore/downloads/Sourcefire%20SSL%20Appliance%20Fact%20Sheet%20%28English%29.pdf> and by selecting the hyperlink “Sourcefire SSL Appliance Fact Sheet” on the NDM Webpage. Ex. 1017; Ex. 1011. I understand that the SSL Appliances Data Sheet has a copyright date of “© 2010 Sourcefire, Inc.” and was archived by the Internet Archive on March 16, 2012. Ex. 1011, 3. I also understand that, on the NDM Webpage, the NDM company identifies itself as a “Sourcefire Authorized Distributor.” Ex. 1017.

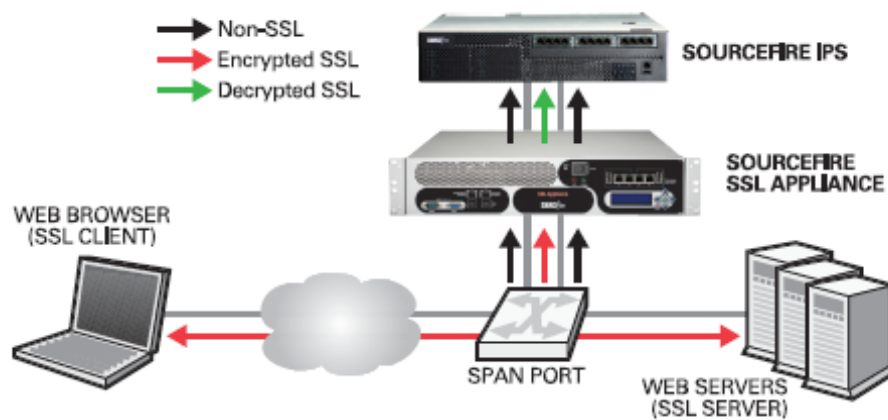


Figure 1. Passive IDS Configuration

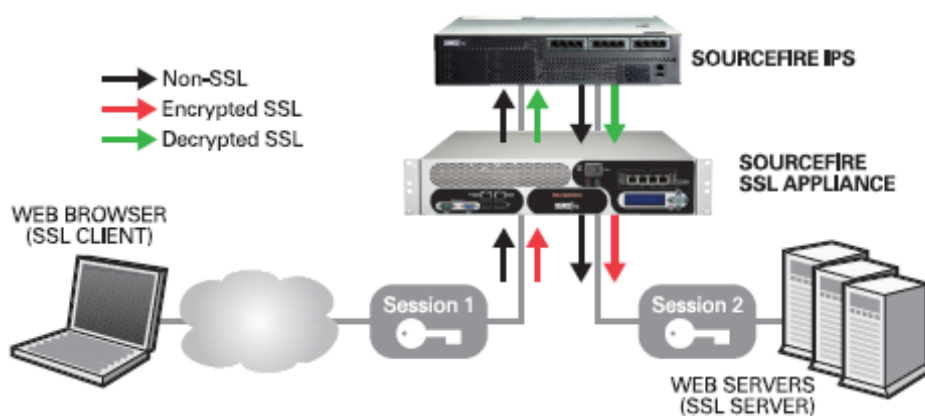


Figure 2. Inline IPS Configuration

Ex. 1010; Ex. 1011.

73. Based on the above-described historical development of the network security systems art, my expert opinion is that the claimed features of the '343 Patent were well-known in the art and were commonly used by many stateful firewall and IPS systems of the period between the late 1990s and late 2000s. These conventional stateful firewall and IPS systems employed a two-stage packet filtering process that used customizable rules to filter packets using the 5-tuple (Stage I), as well as

filtering based on application header information (Stage II), or indeed any aspect of application data.

74. In this declaration, I have analyzed the claims of the '343 Patent against a version 4.10 user guide for the Sourcefire 3D System (Ex. 1004), a commercial IPS derivative of the Snort open-source IDS referenced above. This user guide (Ex. 1004) describes certain functionality of the Sourcefire 3D System which was well-known to a person of ordinary skill in the art prior to March 2013. The Sourcefire 3D System was widely available, was based on the leading open-source network IDS (Snort), and had a large user community developing substantial collections of rules for it. Moreover, as outlined in detail below, the user manual (Ex. 1004) discloses a two-stage packet filtering process that uses customizable rules to filter packets using the 5-tuple (Stage I), as well as filtering based on application header information including PUT and GET commands, and TLS version to block or allow the packets (Stage II).

VII. Overview of the '343 Patent

A. Subject Matter Overview

75. The '343 Patent is titled "Filtering Network Data Transfers." '343 Patent (Ex. 1001) at Cover. The '343 Patent relates to systems and methods for filtering data packets leaving a network to ensure that the packets do not improperly transfer sensitive data from the network—a breach associated with a type of cyber-

attack known as exfiltration. Exfiltrations are often facilitated using popular network data transfer protocols, such as the Hypertext Transfer Protocol (HTTP) used by the Web. Such attacks may appear to an insufficiently careful observer as normal network behavior. For example, older network trust models (*e.g.*, those used by firewalls that do not interpret application-layer data like HTTP) may interpret exfiltrations as trusted operations. '343 Patent, 1:28-51.

76. The '343 Patent purports to solve this problem and detect exfiltration attempts via a Packet Security Gateway (PSG) that receives and examines packets that are exiting a trusted network towards an external network. A PSG is configured to receive packets and apply packet filtering rules, as dictated by a Security Policy Management Server. The packet filtering rules implement “operators” that allow or block packets based on the rules. '343 Patent, 3:62-4:14, 8:39-52.

77. Figure 1 of the '343 Patent (below) shows an exemplary Network Environment 100 protected by PSGs 110 and 112 and a Security Policy Management Server 114:

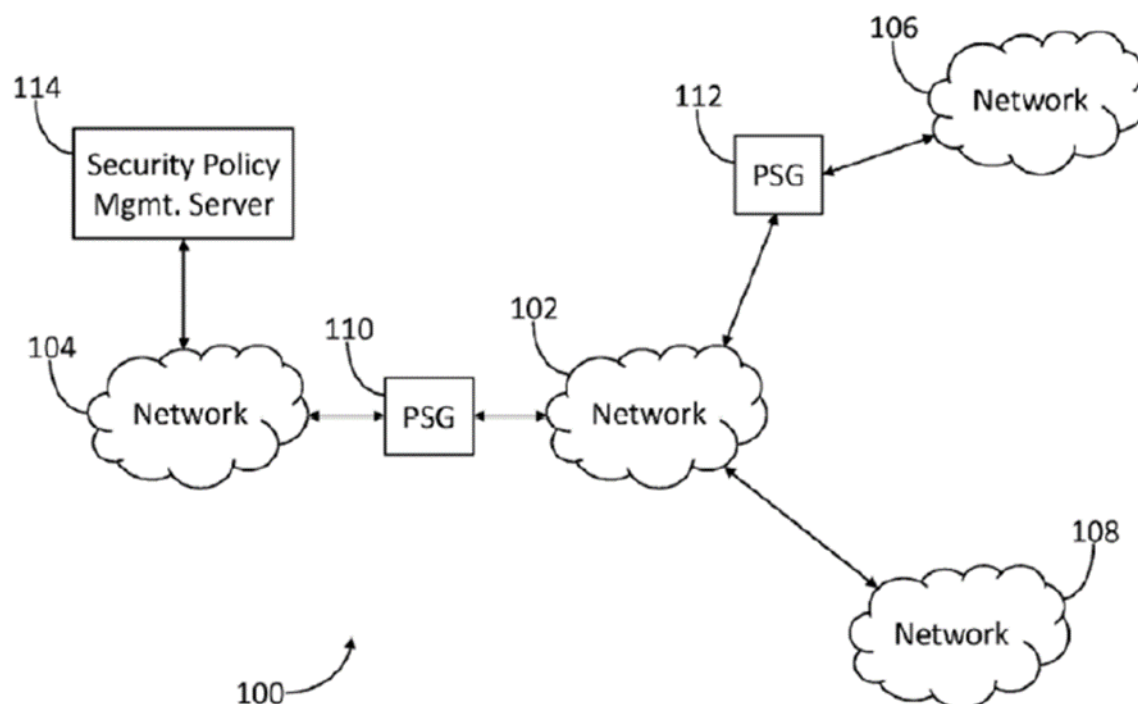
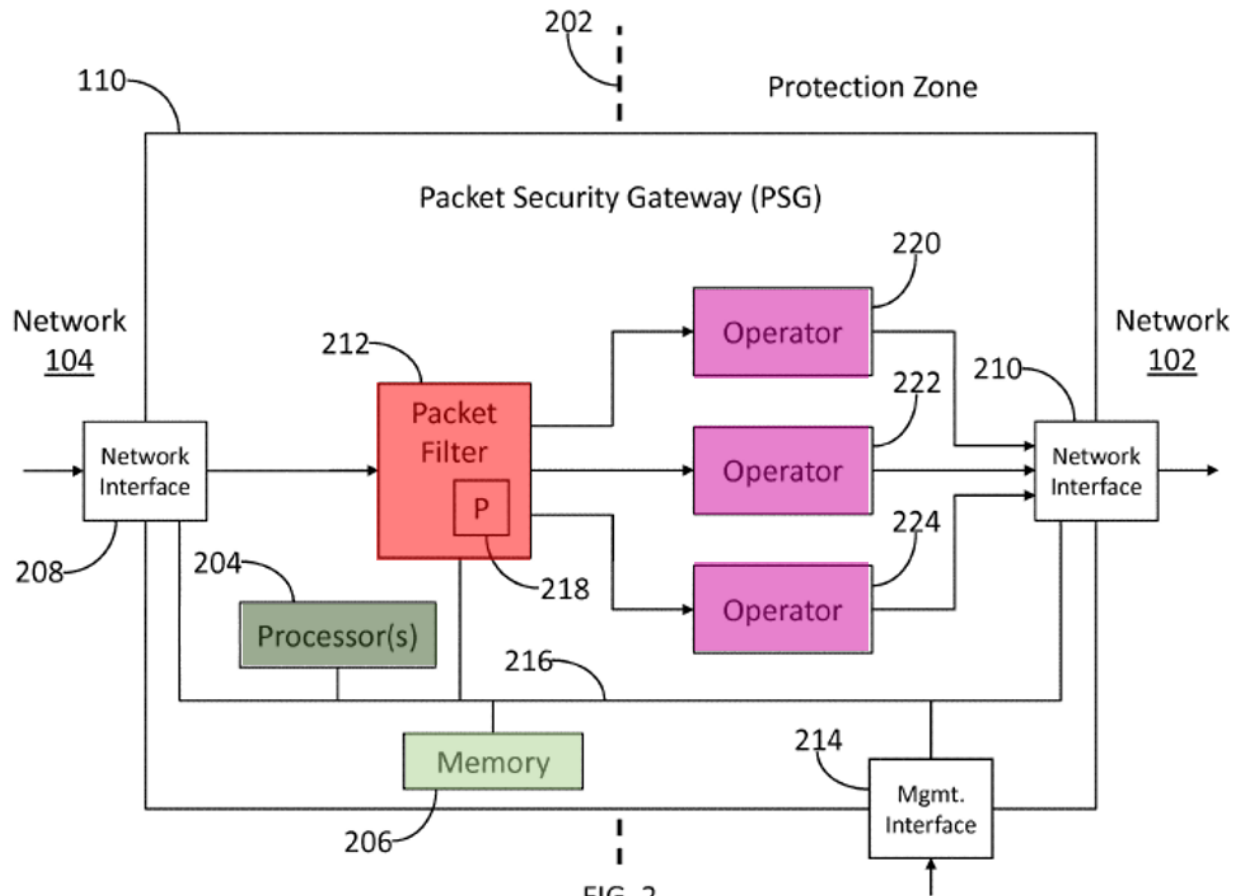


FIG. 1

Network 102, which may be the public Internet or some other large TCP/IP network, connects Network 104, Network 106, and Network 108, which may be Local Area Networks (LANs) or Wide Area Networks (WANs), such as those operated by or associated with various organizations. Network 104 and Network 106 can be distinct LANs associated with a common enterprise X and may each form part of a protected network associated with Security Policy Management Server 114. Network 104 and Network 106 are protected by PSGs 112 and 110 respectively. Network 108 may be owned and operated by a cyber criminal organization Z, which may attempt to attack or collect data from one or more of the networks associated with enterprise X (e.g., Network 104 or Network 106). In this case, PSGs 112 and 110 may protect Network

104 and Network 106 from one or more cyber attacks initiated from Network 108. '343 Patent, 3:31-4:41.

78. Figure 2 of the '343 Patent, which has been color annotated below, shows an embodiment of PSG 110:



PSG 110 may include Network Interface 208 and Network Interface 210, which connect PSG 110 to the respective networks, and a Management Interface 214, which connects to Security Policy Management Server 114. PSG 110 may also include one or more **Processors 204** and **Memory 206**. The **Memory 206** may include program modules which, when executed by the **Processor(s) 204**, may

configure the PSG 110 to enforce network policies that perform the '343 claim limitations. These policies are implemented by **Packet Filter 212** on packets that arrive at the PSG 110 via Network Interface 208. This filter will examine the packets and then forward the packets to one or more **Operators 220, 222, and 224** based on the implemented policies and corresponding rules. An **Operator** may allow traffic, block traffic, or perform an additional function. See Section VII.C (discussing claim construction for the term “operator”). If a packet is allowed to continue towards its destination, a Network Interface 210 connects the packet to its destination via Network 102.

79. The PSG may perform a two-stage filtering process on the received packets. Figure 4 (color annotated) illustrates this process:

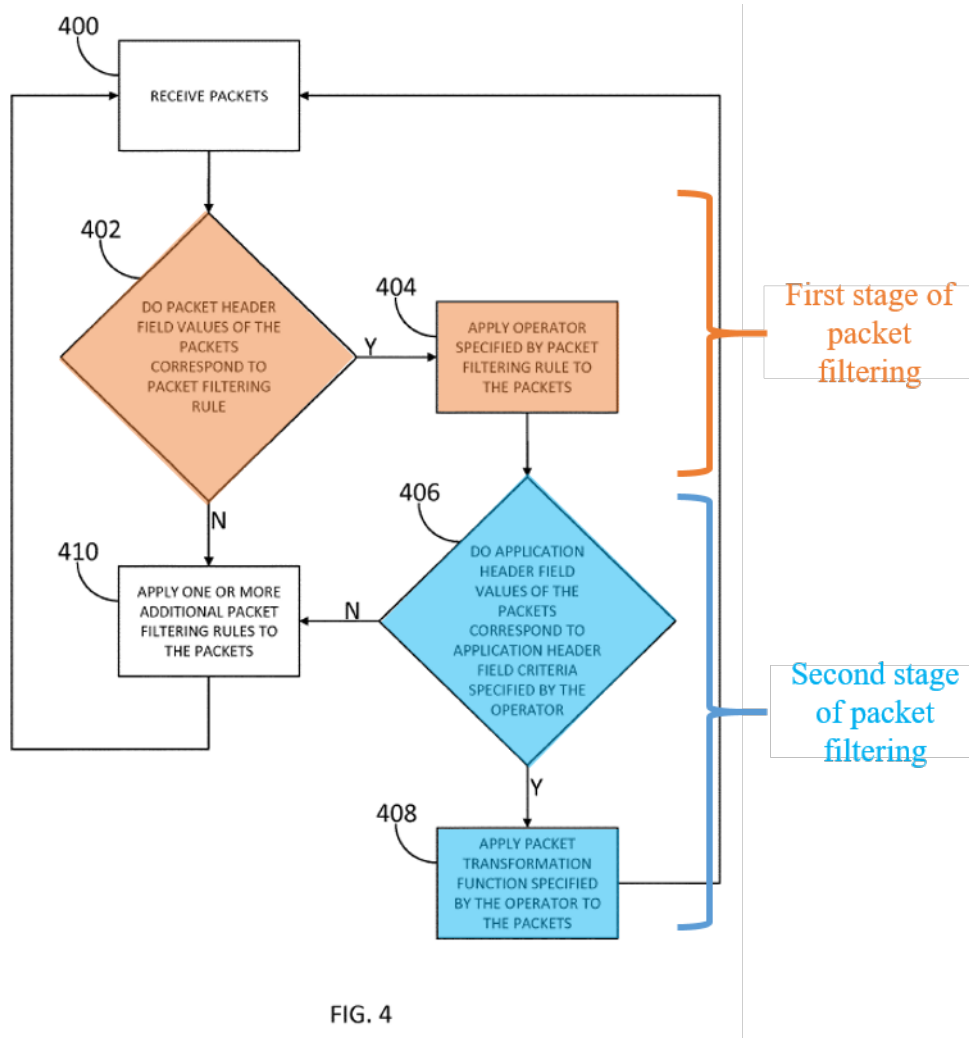
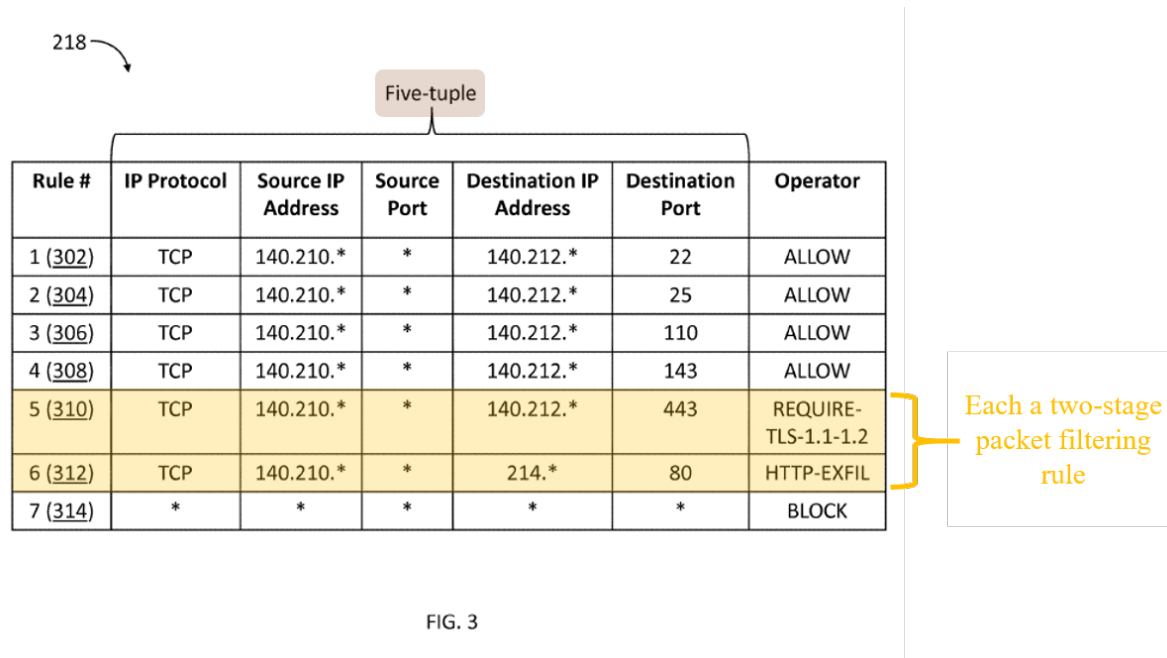


FIG. 4

Packets are received at step 400, and enter **the first filtering stage at Step 402**, where **a determination is made as to whether a portion of the received packets have packet header field values (e.g., one or more data section protocols, source IP addresses, source ports, destination IP addresses, or destination ports) that correspond to a packet filtering rule** provided by the security policy management server. If it is determined that one or more of the portion of received packets have a packet header field value corresponding to a packet filtering rule, **an operator specified by that first stage packet filtering rule is applied to the portion of**

received packets in Step 404. In a one-stage filtering process, the operator would simply block or allow the packets based on the first determination described above. In the two-stage filtering process of Figure 4, however, the operator may specify one or more application-header-field criteria and a packet transformation function. Thus, **in Step 406, a determination is made as to whether one or more application header field values of one or more of the packets—i.e., the packets that entered the second stage—correspond to one or more application header field criteria** specified by the operator from Step 404. If it is determined that one or more of the portion of received packets have application header field values corresponding to one or more application header field criteria, **a packet transformation function is applied to one or more of the packets at Step 408.** A packet transformation function may allow a packet to continue to its destination or block a packet from continuing towards its destination. '343 Patent, 2:24-39, 8:39-51, 9:11-10:47.

80. Figure 3 (color annotated) shows a sample set of packet filtering rules that may comprise a dynamic security policy that is dictated by a Security Policy Management Server and enforced by one or more PSGs:



The **five-tuple** provides examples of the types of values available in the packet header and used as criteria for packet-filtering rules in the **first stage of filtering** for an IP packet, including IP protocol, source IP addresses, source ports, destination IP addresses, and destination ports. For example, rule 1 (302) specifies that IP packets containing one or more TCP packets, originating from an IP address that begins with 140.210, has any source port, destined for an IP address that begins with 140.212, and destined for port 22 (*e.g.*, packets associated with the Secure Shell (SSH) protocol) should have an ALLOW operator applied to them. This is an example of a one-stage filtering process. '343 Patent, 5:50-6:24.

81. Rules 5 (310) and 6 (312) are examples of two-stage packet filtering rules. The **first stage** involves an evaluation of the received packets against the criteria specified in the five-tuple, and if they correspond, the **second stage** involves

an evaluation of the corresponding packets against the additional criteria specified by the operator. For example, Rule 5 (310) specifies that in **the first stage of filtering**, IP packets containing one or more TCP packets, originating from an IP address that begins with 140.210, has any source port, destined for an IP address that begins with 140.212, and destined for port 443 (*e.g.*, packets associated with the port for Hypertext Transfer Protocol Secure (HTTPS) protocol) should have a REQUIRE-TLS 1.1-1.2 Operator applied to them. Thus, in the **second stage of filtering**, the REQUIRE-TLS 1.1-1.2 Operator checks the application header values of the corresponding packets for a specified Transport Layer Security (TLS) protocol (*e.g.*, only packets matching all the packet header criteria and having application header values that indicate HTTPS sessions using TLS versions 1.1 or 1.2 will be allowed to continue). The rule allows only versions 1.1 or 1.2 of the TLS protocol “because the popular TLS version 1.0 protocol has a known security vulnerability that attackers may exploit to decrypt HTTPS sessions.” ’343 Patent, 6:25-33, 6:40-7:23.

82. As another example, Rule 6 (312) specifies in the **first stage of filtering** that IP packets containing one or more TCP packets, originating from an IP address that begins with 140.210, has any source port, destined for an IP address that begins with 214, and destined for port 80 (*e.g.*, associated with the port for Hypertext Transfer Protocol (HTTP) protocol) should have a HTTP-EXFIL Operator applied

to them. Thus, in the **second stage of filtering** applied to HTTP packets determined by the first stage of filtering, the HTTP-EXFIL Operator allows HTTP packets containing a GET method (*e.g.*, a read-only method), but blocks HTTP packets containing other HTTP methods (*e.g.*, read-write methods like PUT, POST, CONNECT, etc.). The rule blocks HTTP PUT or POST methods because they are often used to exfiltrate sensitive data. '343 Patent, 6:25-33, 6:40-7:23.

83. Thus, the '343 Patent describes a filtering process that includes a second stage of filtering, for example to decide whether to allow or block packets using application header criteria such as (1) encryption protocol (*e.g.*, TLS versions) or (2) HTTP methods (*e.g.*, GET versus PUT, POST, CONNECT).

84. Each of the '343 Patent claims is directed to an implementation of a two-stage packet filtering process as described above. Independent Claim 1 is directed to a method; independent Claim 8 is directed to a corresponding apparatus; and independent Claim 15 is directed to a corresponding computer program. Therefore, Claim 1 is representative of the independent claims.

B. File History of the '343 Patent

85. I reviewed the file history of the '343 Patent (Ex. 1002). I understand that the '343 Patent issued on February 18, 2020 from U.S. Application No. 15/614,956 (the '343 application"), which was filed on June 6, 2017. The '343 Patent is a continuation of U.S. Application No. 14/625,486 (now U.S. Patent No.

9,686,193), filed February 18, 2015, which is a continuation of U.S. Application No. 13/795,822 (now U.S. 9,124,552), filed March 12, 2013.

86. On September 27, 2018, the Examiner issued an Office Action rejecting all twenty of the pending claims. Claims 1-3, 6, 8, 13, and 15 were rejected as unpatentable on the ground of nonstatutory double-patenting over claims 1-3, 16, 18, and 19 of the '193 Patent. Ex. 1002, 1326. Claims 1-3, 5, 8-10, 12, and 15-17 were rejected under pre-AIA 35 U.S.C. § 103(a) as unpatentable over U.S. Pub. No. 2011/00559916 ("Ahn") in view of U.S. Pub. No. 2003/0005122 ("Freimuth"). *Id.*, 1334. The Examiner added other references for rejections of the dependent claims. *Id.*, 1343, 1346.

87. On March 27, 2019, Applicant responded by amending the claims, including amendments to independent claims 1, 8, and 15 to differentiate between "a first criterion" specified to examine a packet header and a "second criterion" specified to examine an "application header field value," and to recite a "matching" packet-filtering rule, among other changes. *Id.*, 1424, 1427, 1429. Applicant argued that Ahn fails to teach or suggest a determination based on an application header field value. Applicant further argued that Freimuth fails to "teach or suggest processing packets comprising data corresponding to a first criterion specified by at least one matching packet-filtering rule" because after the TCP handshake completes and data is received to determine the HTTP header, the kernel of Freimuth classifies

the request based on the header values. *Id.*, 1421-22.

88. On June 3, 2019, the Examiner again rejected all twenty pending claims. The Examiner rejected claims 1, 8, and 15 for failing to comply with the written description requirement of 35 U.S.C. § 112 (pre-AIA), first paragraph because the specification did not convey that the “second portion of packets is a portion of the first portion of packets.” *Id.*, 1552. The Examiner also rejected the claims under pre-AIA 35 U.S.C. § 103(a). Claims 1-3, 5, 8-10, 12, and 15-17 were rejected as unpatentable over Ahn in view of U.S. Pub. 2012/0084866 (“Stolfo”). *Id.*, 1554. The Examiner added other references for rejections of the dependent claims. *Id.*, 1562, 1565, 1567, 1569.

89. On November 4, 2019, Applicant filed a Request for Continued Examination and amended independent claims 1, 8, and 15 to recite that “at least one packet transformation function indicates whether each packet in the second portion of packets is allowed to continue toward its destination” and to remove the limitation that the second portion of packets is a portion of the first portion of packets. *Id.*, 1603, 1605-08. In response to the Examiner’s rejections, Applicant argued that the references do not disclose “apply[ing], to each packet in the second portion of packets that match the second criterion, at least one packet transformation function configured to prevent an exfiltration operation, wherein the at least one packet transformation function indicates whether each packet in the second portion of

packets is allowed to continue toward its destination” as recited in independent claims 1, 8, and 15. *Id.*, 1612-13. In particular, Applicant argued that Stolfo instead disclosed generating decoy traffic rather than applying a transformation to existing network traffic, and does not prevent an exfiltration operation. *Id.*, 1613.

90. A Notice of Allowance issued on December 12, 2019, in which the examiner pointed to Applicant’s November 4, 2019 arguments with respect to independent claims 1, 8, and 15 as a reason for allowance. *Id.*, 1639-40. Thus, the ’343 Patent claims were apparently allowed because the Examiner concluded that the prior art of record failed to disclose a two-stage packet filtering process.

C. Interpretation of the ’343 Patent Claims at Issue

91. I have been informed that only terms necessary to resolve the controversy need to be construed. For the purposes of the present declaration, I have applied the legal principles outlined above in Section V.C and propose a construction for a single claim term, as indicated below.

“operator”

92. Claim 1 recites “one or more operators specified by the at least one matching packet-filtering rule.” ’343 Patent, 11:42-44. In computer science, the term “operator” is used to refer to characters or symbols that determine an action or process to be performed. In the context of the ’343 Patent, an operator represents the action or process applied to packets. The action or process may involve, for example,

a “packet transformation function” such as “an ALLOW packet transformation function.” ’343 Patent, 2:24-58, 5:38-40, 7:41-48, 9:53-61. Some operators may specify “application header field criteria” that condition the packet transformation function; examples of these operators are characterized as “additional operators” in the specification. *Id.*, 2:28-34. The ’343 Patent does not require operators to follow any particular format.

93. As discussed further below, the ’343 claims refer to operators that specify application header field criteria. For example, claim 1 recites, in limitation [1.c], “applying ... one or more operators specified by the at least one matching packet-filtering rule,” and recites in limitation [1.d], “determining, based on an application header field value, the second portion of packets based on whether the first portion of packets comprises data corresponding to second criterion specified by one or more operators specified by the at least one matching packet-filtering rule.” ’343 Patent, 11:41-57. Thus, the term “operator” has its plain and ordinary meaning. As recited in the claim, and like the additional operator described in the specification, the claimed “operator” specifies one or more application-header-field-value criteria, but otherwise is not limited to any particular exemplary embodiment of the additional operator described the specification.

VIII. Overview of the Cited References

94. General descriptions provided for the following references and

combinations thereof are incorporated into each subsection and mapping of the claims that includes citations to these references.

A. Sourcefire (Ex. 1004)

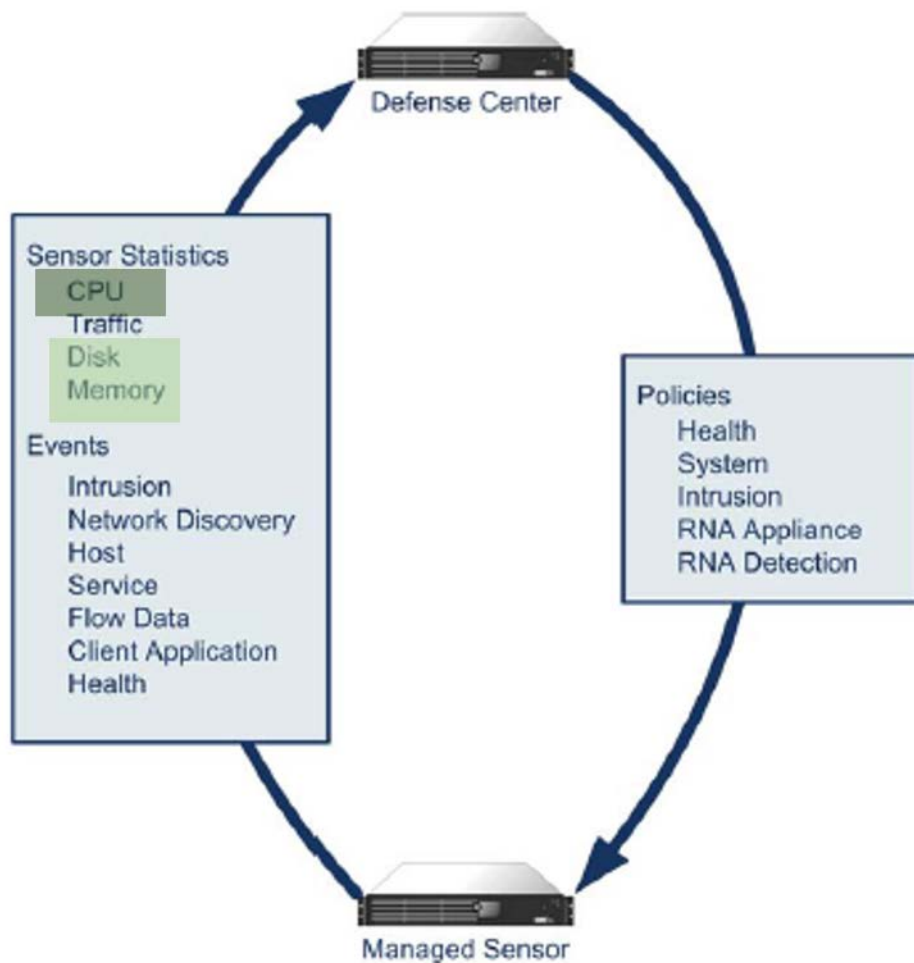
95. The Sourcefire 3D System User Guide (Version 4.10) (“Sourcefire”) is a user manual for the Sourcefire “3D System.” The 3D System provides “real-time network intelligence for real-time network defense.” Sourcefire, 32.

96. 3D Sensors are packet-filtering devices that “aggregate network and user intelligence while defending the network against internal and external threats.” 3D Sensors run any combination of three major software components: Intrusion Prevention System (“IPS”), Real-time Network Awareness (“RNA”), and Real-time User Awareness (“RUA”). Sourcefire, 33-34.

97. IPS utilizes a rules-based detection engine that allows monitoring of the network for attacks that might affect the availability, integrity, or confidentiality of hosts on the network. 3D Sensors that use IPS include a set of pre-developed intrusion rules that detect attacks that are most likely to occur on the network. The user can also create custom intrusion rules. If 3D Sensors are deployed inline on the network to create an inline detection engine, the 3D Sensors can be configured to drop or replace packets based on the intrusion rules. Sourcefire, 33-35.

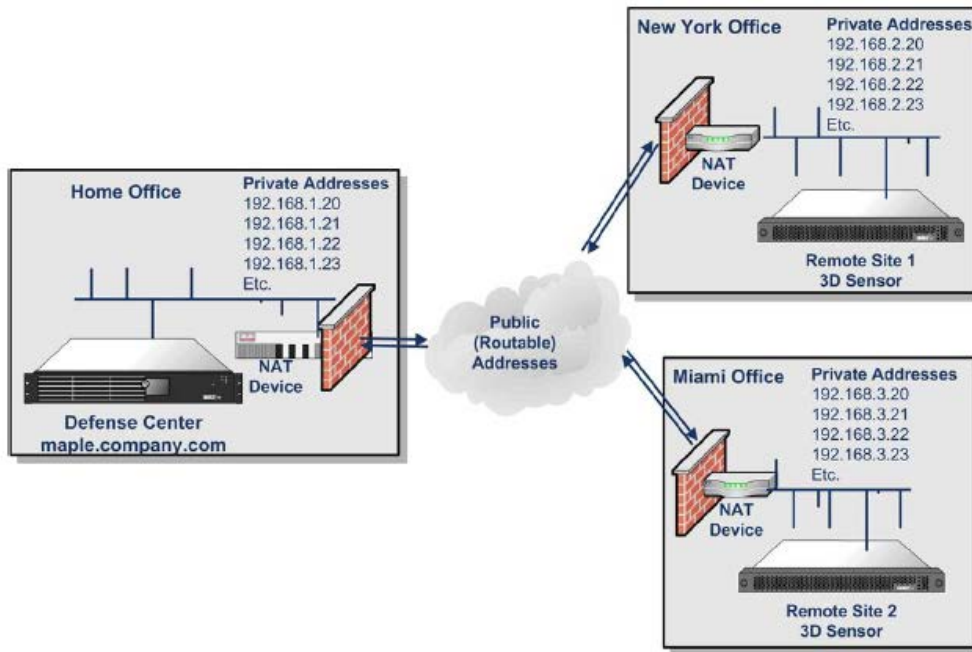
98. Each 3D Sensor includes a **processor** (CPU), **memory**, and **disk storage**. 3D Sensors can be managed by a Defense Center, in which case these

sensors will send updates to the Defense Center, and the Defense Center in turn pushes health, system, and detection policies to the 3D Sensors. The below figure illustrates an exemplary system containing a 3D Sensor and a Defense Center.



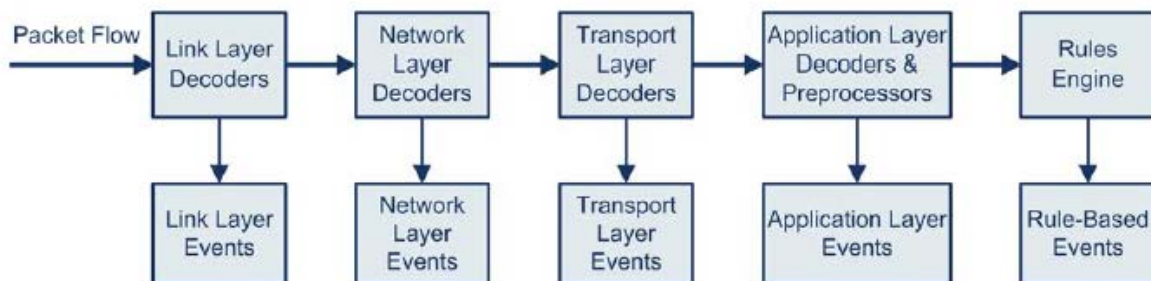
Sourcefire, 107.

99. 3D Sensors are placed at a boundary of protected networks, as shown in the below figure.



Sourcefire, 118.

100. 3D Sensors have three different configurations: passive, inline, or inline with fail-open. In inline deployments, a sensor's inline rules can be configured "to drop suspicious packets." Sourcefire, 70, 222, 253, 254, 262, 263, 264. The following figure illustrates how traffic is analyzed by the 3D Sensor:



Sourcefire, 256.

101. The packets are decoded by packet decoders which convert the packet

headers and payloads into a format that can be easily used by the preprocessors and rules engine. These decoders include “a link layer decoder such as the Ethernet II decoder, a network layer decoder such as the IP decoder, a transport layer decoder such as the TCP decoder, [and] an application layer decoder.” After these packets are decoded, they are sent to “Sourcefire preprocessors [which] normalize traffic and help identify network layer and transport layer protocol anomalies.” An example preprocessor is an application layer preprocessor such as the HTTP Inspect preprocessor.

102. After preprocessing, packets in the 3D Sensor are sent to a Rules Engine. “As packets arrive at the rules engine, it selects the appropriate rule subsets to apply to each packet.” These rules can “restrict packet inspection to the packets originating from specific IP addresses or those destined to a specific IP address.” Preprocessors and the rules engine can also be programmed to follow drop rules. Sourcefire, 256-257, 259-260, 484, 766.

103. The 3D Sensors are sold with pre-programmed rules that were developed by Sourcefire Vulnerability Research Team (VRT). IPS allows a user to select which of these rules they would like to enforce. A user can also write custom rules, either by modifying pre-programmed rules or by writing new rules altogether. Sourcefire encourages this, explaining that “the most successful rules target traffic that may attempt to exploit known vulnerabilities rather than specific known

exploits.” Sourcefire, 762. Rules can be applied on a packet-by-packet basis. Sourcefire, 259.

104. Sourcefire discloses that intrusion rules have two logical sections: the rule header and the rule options. The rule header, which has 5-tuple values associated with it, contains:

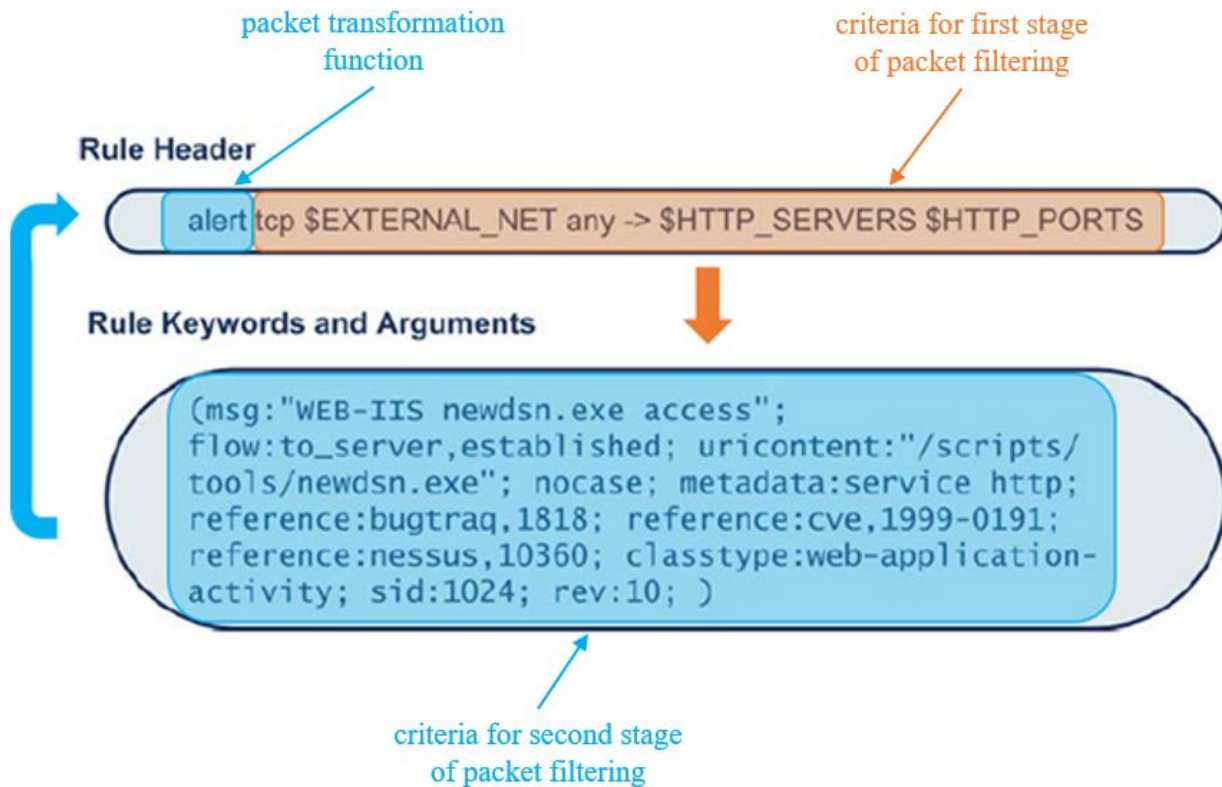
- The 5-tuple:
 - o Protocol
 - o Source and destination IP addresses
 - o Source and destination ports
- The rule’s action or type (*e.g.*, alert and allow, drop, ignore and allow).

The rule options contain:

- Event messages
- Keywords and their parameters and arguments
- Patterns that a packet’s payload must match to trigger the rule
- Specification of which parts of the packet the rules engine should inspect

Sourcefire, 256.

105. Sourcefire provides an example intrusion rule that shows these different parts of an intrusion rule (*i.e.*, a “rule header” which includes parameter and arguments and “rule options” which include keywords and arguments):



Sourcefire, 763 (annotated with labels and arrows to illustrate the rule flow). Based on this rule's structure, this rule is used for a two-stage filtering process, as indicated by the annotations in the above figure.

106. For example, Sourcefire discloses **filtering packets based on packet header information (i.e., a first stage of filtering using first criteria)**. This stage of processing occurs at the network layer decoder (which looks at IP address and protocol) and the transport layer decoder (which looks at ports). This packet header information is compared against the 5-tuple values in the rule header (i.e., the protocol, the source and destination IP addresses, and the source and destination ports), just like Stage 1 of filtering in the '343 Patent. Sourcefire, 256-259, 764-769.

107. Sourcefire further discloses **applying, to packets that match the first criteria, “rule keywords and arguments” from the “rule options” (i.e., second criteria) and applying a rule action (i.e., a packet transformation function) to packets that match the rule keywords and arguments (i.e., a second stage of filtering using second criteria).** A user can specify the behavior of a rule by combining keywords. Keywords and their associated values (called arguments) dictate how the system evaluates packets and packet-related values that the rules engine tests. The Sourcefire 3D System supports keywords that allow the user to perform inspection functions, such as content matching, protocol-specific pattern matching, and state-specific matching. Any number of compatible keywords can be combined to create highly specific rules. The rule also specifies a “rule action” that will pass, alert (and pass), or drop packets that meet the criteria indicated by the rule keywords and arguments specified by the rule. Sourcefire, 297, 359, 761, 765, 769. The rule keywords/arguments are an “operator” as that term is used in the Challenged Claims.

108. As one example of rule keywords for second-stage filtering, Sourcefire discloses SSL rule keywords that invoke the Secure Sockets Layer (SSL) preprocessor to **parse SSL/TLS traffic for encrypted sessions and match against SSL/TLS version information.** Thus, a user could protect a network against a vulnerability associated with a particular encryption protocol version (e.g., TLS

Version 1.0) by writing a rule using the ssl_version keyword that invokes the SSL preprocessor to identify packets that use that protocol version, which had known vulnerabilities.

109. As another example, Sourcefire discloses that HTTP content keywords can be used to write preprocessor and intrusion rules for another application-layer preprocessor, called the HTTPS inspect preprocessor, and for the rules engine, to **search for matched application-layer header content in packets using known HTTP methods “such as GET, PUT, [and] CONNECT.”** Sourcefire, 491,761, 785-786, 807.

110. Thus, Sourcefire describes **filtering packets based on packet header information including the 5-tuple (first criteria)**, just like the Stage I evaluation described in the '343 Patent. Every intrusion rule is qualified by the 5-tuple and, for a given 3D Sensor with IPS, this Stage I evaluation is performed as a packet arrives at the network layer decoder looking at IP address and protocol, and the transport layer decoder looking at ports. Sourcefire, 256-259, 764-769. Sourcefire further describes how a user could design various rules to **pass or drop packets (apply a packet transformation function) based on application-layer header information such as SSL/TLS version or HTTP methods (second criteria)**, just like the Stage II evaluation described in the '343 Patent. Sourcefire, 256-260, 297, 359, 761, 765.

B. Mantripragada (Ex. 1007)

111. U.S. Pat. App. Publication 2009/0103524 (“Mantripragada”) is directed to enforcing security rules for Voice-over-IP (VoIP) and Unified Communications (UC) technology in IP communication networks. Mantripragada, Abstract, ¶¶2-3. For example, Mantripragada teaches monitoring RTP streams to ensure that various RTP values are not tampered with while the RTP session is active. *Id.* ¶¶69-70. RTP values that are checked include TP protocol header sequence number (16 bits), RTP protocol header timestamp (32 bits), RTP protocol header synchronization source identification (32 bits), version (2 bits), padding (1 bit), extension (1 bit), CSRC (Contributing Source) count (4 bits), marker (1 bit), and payload type (7 bits). *Id.* ¶70. A POSITA would have understood that each of these variables is located within the header of an RTP packet. Ex. 1012, 24:55-67 (showing the structure of an RTP header). Mantripragada further teaches rejecting (*i.e.*, dropping) packets that have RTP values that are out of compliance. *Id.* ¶70 (“Media streams that are not compliant with existing session flows ... are rejected....”).

IX. The Prior Art Renders Claims 1-20 Obvious

112. In my expert opinion, for the reasons detailed in this declaration, a person of ordinary skill in the art would have recognized that each and every limitation of claims 1-20 of the ’343 Patent, combined as claimed, is disclosed and at least rendered obvious by the prior art.

113. Specifically, it is my expert opinion that claims 1-20 of the '343 Patent are rendered obvious by the below grounds, as further discussed herein.

Ground	Statute	Claim(s)	Prior Art
1	§103	1-6, 8-13, 15-19	Sourcefire
2	§103	7, 14, 20	Sourcefire in view of Mantripragada

114. Much of my analysis is presented within claim charts showing the portion each reference that discloses or renders obvious each of the claim limitations. The charts present, limitation by limitation, the relevant disclosures from the prior art references, and map the claim elements of the Challenged Claims to elements from those references. I understand that the Petition in this proceeding contains similar claim charts. I had input into and reviewed the claim charts in the Petition.

115. The citations in the claim charts are sufficient to demonstrate the unpatentability of each claim, but are not intended to be exhaustive. I reserve the right to amend the charts, along with any other portion of my declaration, as needed.

A. Independent Claims 1, 8, and 15

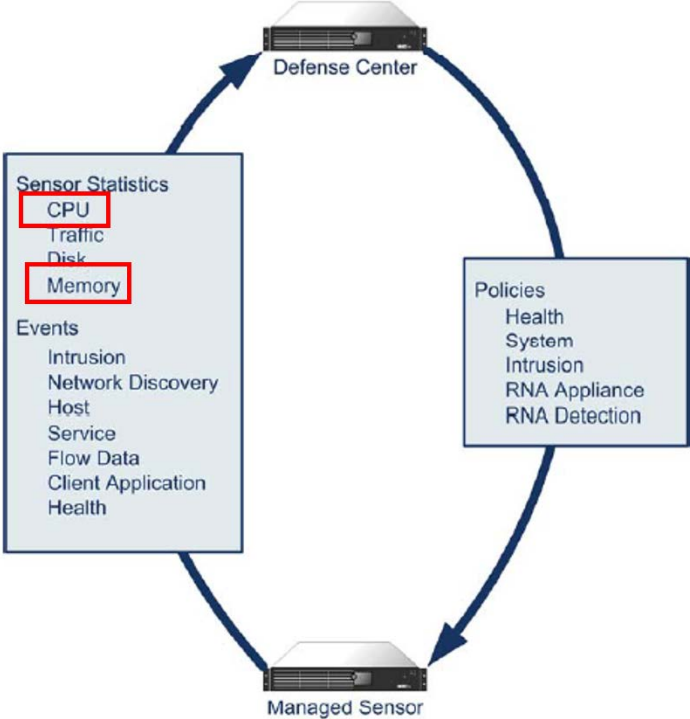
116. Claim 1, 8, and 15 are parallel claims directed to the same implementation of a two-stage packet filtering process. Independent Claim 1 is directed to a method; independent Claim 8 is directed to a corresponding apparatus that performs the Claim 1 method; and independent Claim 15 is directed to a computer program that performs the Claim 1 method. In my analysis below, I have

analyzed the three claims together.

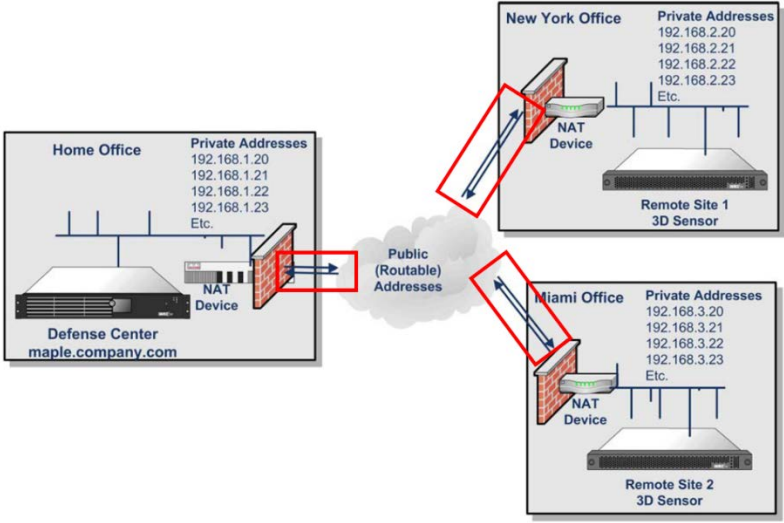
[1.pre], [1.a]; [8.pre], [8.a], [8.b], [8.c]; [15.pre], [15.a]

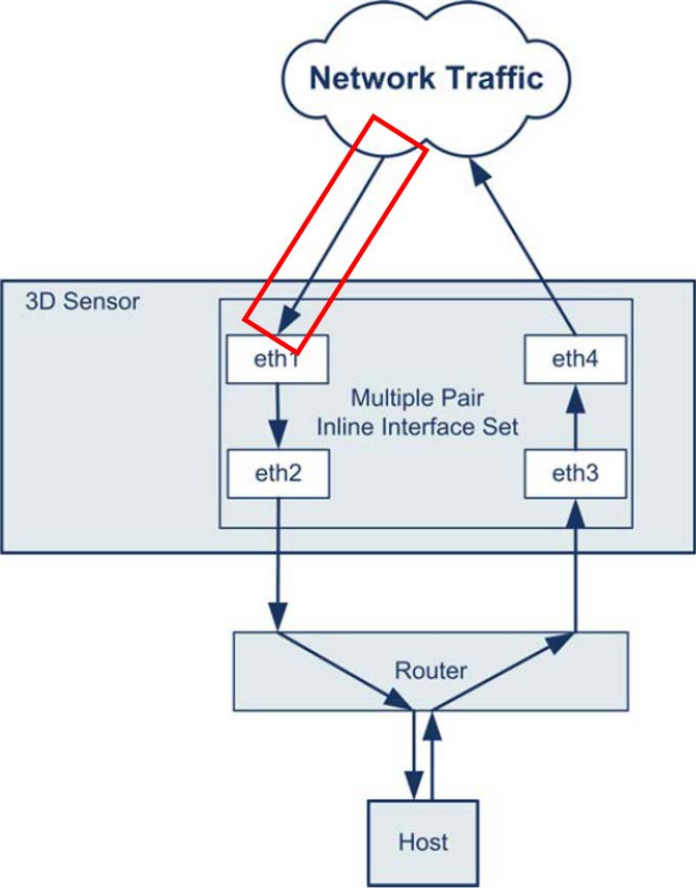
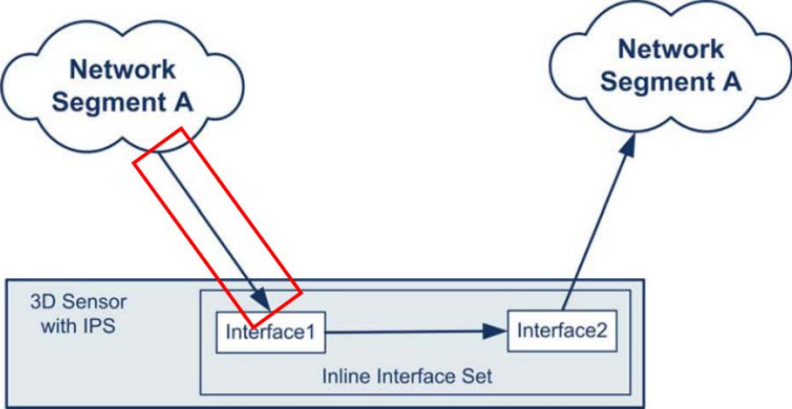
117. To the extent the preambles of claims 1, 8, and 15 are limiting, Sourcefire discloses the preambles. In addition, Sourcefire discloses limitations [1.a], [8.a], [8.b], [8.c], and [15.a], which are discussed together with the preambles.

Claim Language	Sourcefire
[1.pre] A method comprising: [1.a] receiving, by a computing system comprising memory and at least one processor, a plurality of packets, w[herein the plurality of packets comprises a first portion of packets and a second portion of packets;	Sourcefire discloses [1.pre] a method comprising: [1.a] receiving, by a computing system (e.g., “3D Sensor”) comprising memory (e.g., comprising “memory,” “disk”) and at least one processor (e.g., comprising a “CPU”), a plurality of packets, wherein the plurality of packets comprises a first portion of packets and a second portion of packets (e.g., receiving “packets” comprising first and second portions). Sourcefire discloses [8.pre] an apparatus (e.g., “3D Sensor”) comprising: [8.a] at least one processor (e.g., “CPU”); and [8.b] memory (e.g., “memory,” “disk”) storing instructions that, when executed by the at least one processor (e.g., executed by the “processor,” “CPU”), cause the apparatus to: [8.c] receive a plurality of packets (e.g., receiving “packets”); Sourcefire discloses [15.pre] one or more non-transitory computer-readable media (e.g., “memory,” “disk”) comprising instructions that, when executed by one or more computing devices (e.g., executed by the “CPU” of the “3D Sensor”), cause the one or more computing devices to: [15.a] receive a plurality of packets (e.g., receiving “packets”);
[8.pre] An apparatus comprising: [8.a] at least one processor; and	Sourcefire discloses a computing system referred to as a “3D Sensor” which comprises memory, such as RAM and disk

Claim Language	Sourcefire
[8.b] memory storing instructions that, when executed by the at least one processor, cause the apparatus to: [8.c] receive a plurality of packets; [15.pre] One or more non-transitory computer-readable media comprising instructions that, when executed by one or more computing devices, cause the one or more computing devices to: [15.a] receive a plurality of packets;	storage, and a processor. A person of ordinary skill in the art would have understood that the software that runs on the 3D sensor to implement custom intrusion rules is stored in memory (<i>e.g.</i>, RAM or disk) and executes on the CPU. 90: “The System Load widget shows the <i>CPU usage (for each CPU), memory (RAM) usage...</i>”⁵ 204: “The maximum number of detection engines per sensor is two times the number of CPUs (for example, if a <i>sensor has 24 CPUs</i>, you can create up to 48 detection engines).” 107:  The diagram illustrates the Sourcefire architecture. At the top is the 'Defense Center' and at the bottom is the 'Managed Sensor'. A curved arrow points from the Managed Sensor up to the Defense Center, and another curved arrow points from the Defense Center down to the Managed Sensor. In the center is a large box containing three sections: 'Sensor Statistics' (with 'CPU' and 'Memory' highlighted by red boxes), 'Policies' (listing Health, System, Intrusion, RNA Appliance, and RNA Detection), and 'Events' (listing Intrusion, Network Discovery, Host, Service, Flow Data, Client Application, and Health). Sourcefire discloses receiving by the computing system a plurality of packets. For example, the 3D Sensor receives

⁵ All emphasis is added unless otherwise noted.

Claim Language	Sourcefire
	internet traffic and captures packets within that traffic for examination. 118:  The diagram illustrates a network architecture with three main components: Home Office, New York Office, and Miami Office. The Home Office is labeled 'Defense Center maple.company.com' and lists private addresses: 192.168.1.20, 192.168.1.21, 192.168.1.22, 192.168.1.23, and 'Etc.'. It contains a NAT Device. The New York Office is labeled 'New York Office' and lists private addresses: 192.168.2.20, 192.168.2.21, 192.168.2.22, 192.168.2.23, and 'Etc.'. It contains a NAT Device. The Miami Office is labeled 'Miami Office' and lists private addresses: 192.168.3.20, 192.168.3.21, 192.168.3.22, 192.168.3.23, and 'Etc.'. It contains a NAT Device. A cloud labeled 'Public (Routeable) Addresses' is connected to the NAT devices of all three offices. Red arrows indicate traffic flow from the Home Office to the New York Office and from the Home Office to the Miami Office. A red box highlights the NAT Device in the Home Office. 235:

Claim Language	Sourcefire
	 Network Traffic 3D Sensor eth1 eth2 eth3 eth4 Multiple Pair Inline Interface Set Router Host 264:  Network Segment A Network Segment A 3D Sensor with IPS Interface1 Interface2 Inline Interface Set 34: “By placing 3D Sensors with IPS on key network segments, you can <i>examine the packets</i> that traverse your network for malicious activity.”

Claim Language	Sourcefire
	257: “Before packets can be inspected by the IPS detection engine, the <i>packets must be captured from the network</i>. The following illustration shows how the detection engine sniffs packets, then decodes them prior to any further analysis.” 257: <pre> graph TD PS[Packet Stream] --> NIC[NIC] subgraph Sensor [3D Sensor with IPS] NIC -- "Ethernet, SLIP, 802.11, etc." --> PD[Packet Decoder] end </pre> The diagram shows a horizontal double-headed arrow at the top labeled "Packet Stream". A vertical arrow points down from this stream into a box labeled "NIC". This "NIC" box is inside a larger box labeled "3D Sensor with IPS". From the "NIC" box, an arrow points right to another box labeled "Packet Decoder". The arrow between the NIC and the Packet Decoder is labeled "Ethernet, SLIP, 802.11, etc.". 264: “[T]he detection engine using an inline interface set can <i>see all of the traffic that passes through the interfaces</i> in its interface set, regardless of the traffic’s destination.” - See also 33, 118, 222-230, 234-235, 253-254, 257, 262-263, 265, 435-439. A person of ordinary skill in the art would have understood that the plurality of packets have a first portion of packets and a second portion of packets. For example, in the two-stage packet filtering process described below with respect to Elements [1.b] through [1.e], received packets are compared against criteria in the rule header, and then the packets that correspond to those criteria (<i>i.e.</i>, a first portion of packets) are compared against criteria in the rule keywords and arguments to determine which packets are allowed (<i>i.e.</i>, a second portion of packets).

[1.b], [8.d], [15.b]: [determining/ determine], based on a packet header field value, whether [each packet of] the plurality of packets comprises data corresponding to first criterion specified by one or more packet-filtering rules;

118. Sourcefire discloses limitations [1.b], [8.d], and [15.b].

Claim Language	Sourcefire
[1.b]/[8.d]/[15.b] [determining/ determine], based on a packet header field value, whether [each packet of] the plurality of packets comprises data corresponding to first criterion specified by one or more packet-filtering rules;	Sourcefire discloses determining, based on a packet header field value (e.g., based on values such as “protocol” or “source and destination IP address” in the packet header that are obtained by “decod[ing] through the first three TCP/IP layers”), whether each packet of the plurality of packets comprises data corresponding to first criterion (e.g., “compares packets against the conditions specified in each rule” which begins with a first stage of applying criteria such as “protocol” or “source and destination IP addresses” in a 5-tuple in the “rule header”) specified by one or more packet-filtering rules (e.g., “intrusion rules”). Sourcefire discloses first criterion specified by packet-filtering rules. For example, Sourcefire discloses intrusion rules. Sourcefire, 428-430. Each rule has two parts: the rule header and the rule options. Each rule header includes a 5-tuple (shown in orange, below) specifying the protocol, the source and destination IP address, and the source and destination ports that the rule applies to (i.e., first criteria). This is consistent with the ’343 Patent’s disclosure that criteria can include data in the “form of a five-tuple” which may comprise values such as protocol type, source IP addresses, or destination IP address. ’343 Patent, 5:53-64. • 763 (annotated):

Claim Language	Sourcefire																								
	packet transformation function criteria for first stage of packet filtering Rule Header Rule Keywords and Arguments criteria for second stage of packet filtering 764: Type Source IP Operator Destination Port Protocol Source Port Destination IP Rule Header Values <table><thead><tr><th>Rule Header Component</th><th>Example Value</th><th>This Value...</th></tr></thead><tbody><tr><td>Action</td><td>alert</td><td>Generates an intrusion event when triggered.</td></tr><tr><td>Protocol</td><td>tcp</td><td>Tests TCP traffic only.</td></tr><tr><td>Source IP Address</td><td>\$EXTERNAL_NET</td><td>Tests traffic coming from any host that is not on your internal network.</td></tr><tr><td>Source Ports</td><td>any</td><td>Tests traffic coming from any port on the originating host.</td></tr><tr><td>Operator</td><td>-></td><td>Tests external traffic (destined for the web servers on your network).</td></tr><tr><td>Destination IP Address</td><td>\$HTTP_SERVERS</td><td>Tests traffic to be delivered to any host specified as a web server on your internal network.</td></tr><tr><td>Destination Ports</td><td>\$HTTP_PORTS</td><td>Tests traffic delivered to an HTTP port on your internal network.</td></tr></tbody></table> 761: “An intrusion rule is a specified set of keywords and arguments on a 3D Sensor with the IPS component that detects attempts to exploit vulnerabilities on your network by analyzing network traffic to check if it	Rule Header Component	Example Value	This Value...	Action	alert	Generates an intrusion event when triggered.	Protocol	tcp	Tests TCP traffic only.	Source IP Address	\$EXTERNAL_NET	Tests traffic coming from any host that is not on your internal network.	Source Ports	any	Tests traffic coming from any port on the originating host.	Operator	->	Tests external traffic (destined for the web servers on your network).	Destination IP Address	\$HTTP_SERVERS	Tests traffic to be delivered to any host specified as a web server on your internal network.	Destination Ports	\$HTTP_PORTS	Tests traffic delivered to an HTTP port on your internal network.
Rule Header Component	Example Value	This Value...																							
Action	alert	Generates an intrusion event when triggered.																							
Protocol	tcp	Tests TCP traffic only.																							
Source IP Address	\$EXTERNAL_NET	Tests traffic coming from any host that is not on your internal network.																							
Source Ports	any	Tests traffic coming from any port on the originating host.																							
Operator	->	Tests external traffic (destined for the web servers on your network).																							
Destination IP Address	\$HTTP_SERVERS	Tests traffic to be delivered to any host specified as a web server on your internal network.																							
Destination Ports	\$HTTP_PORTS	Tests traffic delivered to an HTTP port on your internal network.																							

Claim Language	Sourcefire
	matches the criteria in the rule. IPS <i>compares packets against the conditions specified in each rule</i> and, if the packet data matches all the conditions specified in a rule, the rule triggers.” • 762: “All standard text rules contain two logical sections: the rule header and the rule options. <i>The rule header contains:</i> ○ the rule’s action or type ○ <i>the protocol</i> ○ <i>the source and destination IP addresses and netmasks</i> ○ direction indicators showing the flow of traffic from source to destination ○ <i>the source and destination ports</i> • 766: “you must <i>specify the protocol of the traffic the rule inspects</i>” • 766: “You can <i>restrict packet inspection to the packets originating from specific IP addresses or those destined to a specific IP address.</i>” • 768: “You can <i>restrict packet inspection to packets originating from or destined to specific ports.</i>” • 861-62: “[Y]ou can <i>use the rule header settings to focus the rule to only match traffic using a specific protocol and traveling to or from specific IP addresses or ports.</i>” • See also 254, 257, 260, 358-361, 428-430, 485, 624-626, 645, 765-771, 869-872. Sourcefire discloses determining whether a packet header field value (<i>e.g.</i>, protocol, source and destination IP address, source and destination ports) of each packet has data corresponding to the first criterion of the packet-filtering rule (<i>i.e.</i>, corresponding to the 5-tuple values specified by the rule

Claim Language	Sourcefire
	header). As explained above, the rule header in an intrusion rule specifies the protocol, source and destination addresses, and source and destination ports that identify the packets to be inspected. As each packet is received by the 3D Sensor, it is decoded by link layer, network layer, transport layer, and application layer decoders to extract header information before being sent to a set of preprocessors and the rules engine. Thus, a person of ordinary skill in the art would have understood that the decoders, preprocessor and/or rules engine implementing an intrusion rule would extract the protocol, source and destination IP addresses, and source and destination ports (<i>i.e.</i>, packet header field values) from the header for at least a portion of received packets, and compare those values to the 5-tuple values in the rule header. 260: <pre> graph LR subgraph "3D Sensor with IPS" subgraph " " PD[Packet Decoder] --> P[Preprocessors] P --> RE[Rules Engine] end ED[Event Database] PD --> ED P --> ED RE --> ED end </pre> 256: “Intrusion events can be generated by: - o a link layer decoder such as the Ethernet II decoder - o a network layer decoder such as the IP decoder - o a transport layer decoder such as the TCP decoder - o an application layer decoder or preprocessor such as the HTTP Inspect preprocessor - o the rules engine”

Claim Language	Sourcefire										
	621: “Before sending captured packets to a preprocessor, the detection engine first sends the packets to the packet decoder. The packet decoder converts packet headers and payloads into a format that can be easily used by the preprocessors and the rules engine. Each stack layer is decoded in turn, beginning with the data link layer and continuing through the network and transport layers.” 258: <table> <tr> <th colspan="2">Decoded Packets</th></tr> <tr> <th>In this TCP/IP layer...</th><th>These packets are decoded...</th></tr> <tr> <td>Data Link</td><td> - raw packets - loopback interface - Ethernet - Token Ring - Fiber Distributed Data Interface (FDDI) - Linux SLL (cooked mode) - IEEE 802.11 - Point-to-Point Protocol (PPP) - Point-to-Point Protocol over Ethernet (PPPOE) - Serial Line Internet Protocol (SLIP) - ISDN for Linux (I4L) - Address Resolution Protocol (ARP) - Extensible Authentication Protocol (EAP) - EAP over LAN (EAPOL) </td></tr> <tr> <td>Network</td><td> - Internet Protocol (IP) - Internet Control Message Protocol (ICMP) </td></tr> <tr> <td>Transport</td><td> - Transmission Control Protocol (TCP) - User Datagram Protocol (UDP) </td></tr> </table> 258-259: “After the packets are decoded through the first three TCP/IP layers, they are sent to preprocessors, which normalize traffic at the application layer and detect protocol anomalies. After the packets have passed through the preprocessors, they are sent to the rules engine. The rules engine inspects the packet headers and payloads to determine whether they trigger any of the ... rules.” 602: “Before packets are sent to preprocessors, the packet decoder converts packet headers and payloads into a format that can be easily used by the	Decoded Packets		In this TCP/IP layer...	These packets are decoded...	Data Link	- raw packets - loopback interface - Ethernet - Token Ring - Fiber Distributed Data Interface (FDDI) - Linux SLL (cooked mode) - IEEE 802.11 - Point-to-Point Protocol (PPP) - Point-to-Point Protocol over Ethernet (PPPOE) - Serial Line Internet Protocol (SLIP) - ISDN for Linux (I4L) - Address Resolution Protocol (ARP) - Extensible Authentication Protocol (EAP) - EAP over LAN (EAPOL)	Network	- Internet Protocol (IP) - Internet Control Message Protocol (ICMP)	Transport	- Transmission Control Protocol (TCP) - User Datagram Protocol (UDP)
Decoded Packets											
In this TCP/IP layer...	These packets are decoded...										
Data Link	- raw packets - loopback interface - Ethernet - Token Ring - Fiber Distributed Data Interface (FDDI) - Linux SLL (cooked mode) - IEEE 802.11 - Point-to-Point Protocol (PPP) - Point-to-Point Protocol over Ethernet (PPPOE) - Serial Line Internet Protocol (SLIP) - ISDN for Linux (I4L) - Address Resolution Protocol (ARP) - Extensible Authentication Protocol (EAP) - EAP over LAN (EAPOL)										
Network	- Internet Protocol (IP) - Internet Control Message Protocol (ICMP)										
Transport	- Transmission Control Protocol (TCP) - User Datagram Protocol (UDP)										

Claim Language	Sourcefire
	<i>preprocessors and the rules engine</i> and detects various anomalous behaviors in packet headers.” • See also 35, 276-277, 428-430, 465, 484, 488-89, 491, 575, 596-601, 608, 625-627, 761-768, 861-864, 2084, 2088-2089, 2093.

[1.c], [8.e], [15.c]: responsive to a determination [by the computing system] that a packet header field value of [the/a] first portion of packets comprises data corresponding to the first criterion specified by at least one matching packet-filtering rule, [applying/ apply], [by the computing system and] to each packet in the first portion of packets, one or more operators specified by the at least one matching packet-filtering rule;

119. Sourcefire discloses [1.c], [8.e], and [15.c].

Claim Language	Sourcefire
[1.c]/[8.e]/[15.c] responsive to a determination [by the computing system] that a packet header field value of [the/a] first portion of packets comprises data corresponding to the first criterion specified by at least one matching packet-filtering	Sourcefire discloses responsive to a determination (<i>e.g.</i>, “if the packet data matches all the conditions specified in a rule, the rule triggers”) by the computing system that a packet header field value of the first portion of packets comprises data corresponding to the first criterion specified by at least one matching packet-filtering rule (<i>e.g.</i>, see [1.b]), applying, by the computing system and to each packet in the first portion of packets, one or more operators (<i>e.g.</i>, applying “rule keywords and arguments” and the “rule action”) specified by the at least one matching packet-filtering rule (<i>e.g.</i>, specified by “intrusion rules”). As discussed above with respect to [1.b], Sourcefire discloses comparing packet header information, <i>e.g.</i>, 5-tuple values as part of the first stage of packet filtering (shown in orange, below), which determines whether to apply the “rule keywords and arguments” to a given packet. The keywords

Claim Language	Sourcefire
rule, [applying/ apply], [by the computing system and] to each packet in the first portion of packets, one or more operators specified by the at least one matching packet-filtering rule;	and arguments are applied to determine whether to apply the “rule action,” such as “pass” or “drop,” to the packet being inspected. 763: 769 (“Using the rules language, you can specify the behavior of a rule by combining keywords. Keywords and their associated values (called arguments) dictate how the system evaluates packets and packet-related values that the rules engine tests. The Sourcefire 3D System currently supports keywords that allow you to perform inspection functions, such as content matching, protocol-specific pattern matching, and state-specific matching. You can combine any number of compatible keywords to create highly specific rules.”) 761: “An intrusion rule is a specified set of keywords and arguments on a 3D Sensor with the IPS component that detects attempts to exploit vulnerabilities on your network by analyzing network traffic to check if it matches the criteria in the rule. IPS compares packets against the conditions specified in each rule and, if the packet data matches all the conditions specified in a rule, the rule triggers.” 765: “Specifying Rule Actions ... Each rule header includes a parameter that specifies the action the system takes when a packet triggers a rule.

Claim Language	Sourcefire
	... Rules with the <i>action set to pass</i> do not generate an event against, or log the details of, the packet that triggered the rule. ... In an inline deployment, rules with the rule state <i>set to Drop</i> and Generate Events generate an intrusion event against the packet that triggered the rule.” • See also 259, 358-359, 399, 403-406, 435-439, 464-466, 599, 762, 765-769. Sourcefire thus discloses that the 3D Sensor applies “keywords and arguments” to each packet in the first portion of packets (<i>i.e.</i>, to packets that match the 5-tuple in the rule header), responsive to a determination by the 3D Sensor that a packet header field value (<i>e.g.</i>, protocol, source and destination IP address, source and destination ports) of the first portion of packets comprises data corresponding to the first criterion specified by at least one matching packet-filtering rule (<i>i.e.</i>, corresponding to the 5-tuple values specified by the rule header). Only if the first stage criterion is met (<i>e.g.</i>, 5-tuple values) are remaining criteria (<i>i.e.</i>, rule keywords and arguments) in the rule tested. Then the rule’s action or type (<i>e.g.</i>, allow, drop) is applied only to the packets that satisfy the keywords and arguments. Sourcefire’s teaching of intrusion rules that apply “keywords and arguments” is a teaching of applying an “operator” as that term is used in the Challenged Claims. As explained in Section VIII.C.1, an “operator” as used in the Challenged Claims specifies one or more application-header-field-value criteria. Sourcefire discloses applying an “operator” that meets this requirement, as set forth below with respect to [1.d] (requiring applying a second criteria “based on an application header field value”).

[1.d], [8.f], [15.d]: [determining/ determine], based on an application header field value, [the/a] second portion of packets based on whether the first portion of packets comprises data corresponding to second criterion specified by one or more operators specified by the at least one matching packet-filtering rule; and

120. Sourcefire discloses [1.d], [8.f], and [15.d].

Claim Language	Sourcefire
[1.d]/[8.f]/[15.d] [determining/ determine], based on an application header field value, [the/a] second portion of packets based on whether the first portion of packets comprises data corresponding to second criterion specified by one or more operators specified by the at least one matching packet-filtering rule; and	Sourcefire discloses determining, based on an application header field value (e.g., “application layer protocol values,” HTTP “request method field, which identifies the action such as GET, PUT, CONNECT,” “SSL_version”), the second portion of packets (e.g., packets matching the first criterion and the second criterion) based on whether the first portion of packets (e.g., see [1.c]) comprises data corresponding to second criterion specified by one or more operators (e.g., “rule keywords and arguments”) specified by the at least one matching packet-filtering rule (e.g., specified by “intrusion rules,” see [1.c]). Sourcefire discloses at least two examples of rule keywords/arguments for identifying matching packets based on an application-layer field value (e.g., rules examining HTTP methods and SSL/TLS versions). As a first example, Sourcefire discloses an HTTP Method option in the HTTP content keywords specification list for checking HTTP methods, such as GET, PUT, and CONNECT, in HTTP request packets. Information about HTTP packets is decoded by the HTTP inspect preprocessor, which is an application-layer preprocessor. • 560-561: “The HTTP Inspect preprocessor is responsible for:

Claim Language	Sourcefire
	- o <i>decoding and normalizing HTTP requests</i> sent to and HTTP responses received from web servers on your network - o ... - o making the normalized data available for additional rule processing • 568: “HTTP Methods Specifies <i>HTTP request methods for the preprocessor to inspect in addition to GET and POST</i>. Use a comma to separate multiple values.” • 786: “HTTP content Keyword Options Select this option... <i>HTTP Method</i> To <i>search for content matches in... the request method field, which identifies the action such as GET, PUT, CONNECT</i>, and so on to take on the resource identified in the URI.” • 806: “The HTTP preprocessor must be enabled to allow processing of rules using the C, H, U, M, or P expression modifiers.” • 807:

Claim Language	Sourcefire														
	Snort-Specific Post Regular Expression Modifiers (Continued) <table> <tr> <th>Option</th><th>Description</th></tr> <tr> <td>H</td><td>Searches for the content within the header, excluding cookies, of an HTTP request or response message decoded by the HTTP Inspect preprocessor. Note that you cannot use this option in combination with the content keyword HTTP Header option to search the same content. See the HTTP content Keyword Options table on page 785 for more information.</td></tr> <tr> <td>D</td><td>Searches for the content within the header, excluding cookies, of a raw HTTP request or response message decoded by the HTTP Inspect preprocessor. Note that you cannot use this option in combination with the content keyword HTTP Raw Header option to search the same content. See the HTTP content Keyword Options table on page 785 for more information.</td></tr> <tr> <td>M</td><td>Searches for the content within the method field of a normalized HTTP request message decoded by the HTTP Inspect preprocessor; the method field identifies the action such as GET, PUT, CONNECT, and so on to take on the resource identified in the URI. See the content keyword HTTP Method option in the HTTP content Keyword Options table on page 785 for more information.</td></tr> <tr> <td>C</td><td>When the HTTP Inspect preprocessor Inspect HTTP Cookies option is enabled, searches for the normalized content within any cookie in an HTTP request header, and also within any set-cookie in an HTTP response header when the preprocessor Inspect HTTP Responses option is enabled. When Inspect HTTP Cookies is not enabled, searches the entire header, including the cookie or set-cookie data. Note that cookies included in the message body are treated as body content. Note also that you cannot use this option in combination with the content keyword HTTP Cookie option to search the same content. See the HTTP content Keyword Options table on page 785 for more information.</td></tr> <tr> <td>K</td><td>When the HTTP Inspect preprocessor Inspect HTTP Cookies option is enabled, searches for the raw content within any cookie in an HTTP request header, and also within any set-cookie in an HTTP response header when the preprocessor Inspect HTTP Responses option is enabled. When Inspect HTTP Cookies is not enabled, searches the entire header, including the cookie or set-cookie data. Note that cookies included in the message body are treated as body content. Note also that you cannot use this option in combination with the content keyword HTTP Raw Cookie option to search the same content. See the HTTP content Keyword Options table on page 785 for more information.</td></tr> <tr> <td>S</td><td>Searches the 3-digit status code in an HTTP response. See the content keyword HTTP Status Code option in the HTTP content Keyword Options table on page 785 for more information.</td></tr> </table> • See also 480, 484, 513, 569, 621, 806, 783-786, 826-837, 859, 880. To the extent Sourcefire does not explicitly describe a rule in which an HTTP method such as GET, PUT, or CONNECT is specifically included as a keyword, a person of ordinary skill in the art would have found obvious a rule using the HTTP Method as part of the criteria to invoke the HTTP inspect preprocessor to identify traffic using that HTTP method (<i>e.g.</i>, PUT), and either pass or drop those packets based on the inspection. For example, Sourcefire discloses creating “your own Sourcefire 3D System rules.” Sourcefire, 377, 762.	Option	Description	H	Searches for the content within the header, excluding cookies, of an HTTP request or response message decoded by the HTTP Inspect preprocessor. Note that you cannot use this option in combination with the content keyword HTTP Header option to search the same content. See the HTTP content Keyword Options table on page 785 for more information.	D	Searches for the content within the header, excluding cookies, of a raw HTTP request or response message decoded by the HTTP Inspect preprocessor. Note that you cannot use this option in combination with the content keyword HTTP Raw Header option to search the same content. See the HTTP content Keyword Options table on page 785 for more information.	M	Searches for the content within the method field of a normalized HTTP request message decoded by the HTTP Inspect preprocessor; the method field identifies the action such as GET, PUT, CONNECT, and so on to take on the resource identified in the URI. See the content keyword HTTP Method option in the HTTP content Keyword Options table on page 785 for more information.	C	When the HTTP Inspect preprocessor Inspect HTTP Cookies option is enabled, searches for the normalized content within any cookie in an HTTP request header, and also within any set-cookie in an HTTP response header when the preprocessor Inspect HTTP Responses option is enabled. When Inspect HTTP Cookies is not enabled, searches the entire header, including the cookie or set-cookie data. Note that cookies included in the message body are treated as body content. Note also that you cannot use this option in combination with the content keyword HTTP Cookie option to search the same content. See the HTTP content Keyword Options table on page 785 for more information.	K	When the HTTP Inspect preprocessor Inspect HTTP Cookies option is enabled, searches for the raw content within any cookie in an HTTP request header, and also within any set-cookie in an HTTP response header when the preprocessor Inspect HTTP Responses option is enabled. When Inspect HTTP Cookies is not enabled, searches the entire header, including the cookie or set-cookie data. Note that cookies included in the message body are treated as body content. Note also that you cannot use this option in combination with the content keyword HTTP Raw Cookie option to search the same content. See the HTTP content Keyword Options table on page 785 for more information.	S	Searches the 3-digit status code in an HTTP response. See the content keyword HTTP Status Code option in the HTTP content Keyword Options table on page 785 for more information.
Option	Description														
H	Searches for the content within the header, excluding cookies, of an HTTP request or response message decoded by the HTTP Inspect preprocessor. Note that you cannot use this option in combination with the content keyword HTTP Header option to search the same content. See the HTTP content Keyword Options table on page 785 for more information.														
D	Searches for the content within the header, excluding cookies, of a raw HTTP request or response message decoded by the HTTP Inspect preprocessor. Note that you cannot use this option in combination with the content keyword HTTP Raw Header option to search the same content. See the HTTP content Keyword Options table on page 785 for more information.														
M	Searches for the content within the method field of a normalized HTTP request message decoded by the HTTP Inspect preprocessor; the method field identifies the action such as GET, PUT, CONNECT, and so on to take on the resource identified in the URI. See the content keyword HTTP Method option in the HTTP content Keyword Options table on page 785 for more information.														
C	When the HTTP Inspect preprocessor Inspect HTTP Cookies option is enabled, searches for the normalized content within any cookie in an HTTP request header, and also within any set-cookie in an HTTP response header when the preprocessor Inspect HTTP Responses option is enabled. When Inspect HTTP Cookies is not enabled, searches the entire header, including the cookie or set-cookie data. Note that cookies included in the message body are treated as body content. Note also that you cannot use this option in combination with the content keyword HTTP Cookie option to search the same content. See the HTTP content Keyword Options table on page 785 for more information.														
K	When the HTTP Inspect preprocessor Inspect HTTP Cookies option is enabled, searches for the raw content within any cookie in an HTTP request header, and also within any set-cookie in an HTTP response header when the preprocessor Inspect HTTP Responses option is enabled. When Inspect HTTP Cookies is not enabled, searches the entire header, including the cookie or set-cookie data. Note that cookies included in the message body are treated as body content. Note also that you cannot use this option in combination with the content keyword HTTP Raw Cookie option to search the same content. See the HTTP content Keyword Options table on page 785 for more information.														
S	Searches the 3-digit status code in an HTTP response. See the content keyword HTTP Status Code option in the HTTP content Keyword Options table on page 785 for more information.														

Claim Language	Sourcefire
	Sourcefire further discloses that intrusion rules can be implemented by the HTTP inspect preprocessor and by the rules engine and provides a specific keyword option to access the HTTP method. Sourcefire, 568 (“Specifies HTTP request methods for the preprocessor to inspect in addition to GET and POST.”); <i>see also</i> 435-439, 491, 569, 785-786, 807. Additionally, a person of ordinary skill in the art would have understood that HTTP methods such as PUT, POST, and CONNECT were well-known methods of network exfiltration, because a malicious actor can use such methods to send data out of a network. This is the reason that Sourcefire teaches incorporating keywords into packet filtering rules to identify these methods.⁶ Indeed, existing packet-filtering firewalls used application methods as part of rule criteria to block and pass packets, <i>e.g.</i>, a known “stateful inspection firewall allowed the FTP ‘GET’ command but disallowed the ‘PUT’ command” (Ex. 1008, 4:31-41), and HTTP methods were used in IDS rules at least as early as 1997 (Ex. 1028, 4-80). Thus, in view of the teachings of Sourcefire, a person of ordinary skill in the art would have been motivated and found it obvious to write an intrusion rule in which the HTTP Method option is specifically included in a rule in order to detect packets containing certain HTTP methods, such as POST, PUT, or CONNECT. Implementing such a rule, would have had the predictable benefit of achieving increased network security by protecting a network against known vulnerabilities.

⁶ That these HTTP methods were known exfiltration methods is confirmed by the ’343 Patent, which similarly recognized that it was well known that HTTP methods such as PUT and POST correspond to an exfiltration method. *See e.g.*, ’343 Patent, 7:21-23, 7:55-60, 2:47-58.

Claim Language	Sourcefire																		
	As a second example, Sourcefire discloses using SSL/TLS rule keywords/arguments to invoke the application-layer SSL preprocessor and extract information on the SSL or TLS version from application packet headers. SSL and TLS are encryption protocols used to secure packets during communication across and/or between networks. 827: “The ssl_version keyword can be used to match against version information for an encrypted session. <i>When a rule uses the ssl_version keyword, the rules engine invokes the SSL preprocessor to check traffic for SSL version information.</i> For example, if you know there is a buffer overflow vulnerability in SSL version 2, you could use the ssl_version keyword with the sslv2 argument to identify traffic using that version of SSL.” 827-828: <table> <tr> <th colspan="2">ssl_version Arguments</th></tr> <tr> <th>Argument</th><th>Purpose</th></tr> <tr> <td>sslv2</td><td>Matches against traffic encoded using Secure Sockets Layer (SSL) Version 2.</td></tr> <tr> <td>sslv3</td><td>Matches against traffic encoded using Secure Sockets Layer (SSL) Version 3.</td></tr> </table> <table> <tr> <th colspan="2">ssl_version Arguments (Continued)</th></tr> <tr> <th>Argument</th><th>Purpose</th></tr> <tr> <td>tlsv1.0</td><td>Matches against traffic encoded using Transport Layer Security (TLS) Version 1.0.</td></tr> <tr> <td>tlsv1.1</td><td>Matches against traffic encoded using Transport Layer Security (TLS) Version 1.1.</td></tr> <tr> <td>tlsv1.2</td><td>Matches against traffic encoded using Transport Layer Security (TLS) Version 1.2.</td></tr> </table> 697: Adaptive profiles can also make the... <i>SSL preprocessors</i> more efficient by checking traffic for service identification.” 825: “Extracting SSL Information from a Session - You can <i>use SSL rule keywords to invoke the Secure Sockets Layer (SSL) preprocessor and</i>	ssl_version Arguments		Argument	Purpose	sslv2	Matches against traffic encoded using Secure Sockets Layer (SSL) Version 2.	sslv3	Matches against traffic encoded using Secure Sockets Layer (SSL) Version 3.	ssl_version Arguments (Continued)		Argument	Purpose	tlsv1.0	Matches against traffic encoded using Transport Layer Security (TLS) Version 1.0.	tlsv1.1	Matches against traffic encoded using Transport Layer Security (TLS) Version 1.1.	tlsv1.2	Matches against traffic encoded using Transport Layer Security (TLS) Version 1.2.
ssl_version Arguments																			
Argument	Purpose																		
sslv2	Matches against traffic encoded using Secure Sockets Layer (SSL) Version 2.																		
sslv3	Matches against traffic encoded using Secure Sockets Layer (SSL) Version 3.																		
ssl_version Arguments (Continued)																			
Argument	Purpose																		
tlsv1.0	Matches against traffic encoded using Transport Layer Security (TLS) Version 1.0.																		
tlsv1.1	Matches against traffic encoded using Transport Layer Security (TLS) Version 1.1.																		
tlsv1.2	Matches against traffic encoded using Transport Layer Security (TLS) Version 1.2.																		

Claim Language	Sourcefire
	<i>extract information about SSL version</i> and session state from packets in an encrypted session. - o ... - o The <i>SSL preprocessor extracts state and version information from specific handshake fields. Two fields within the handshake indicate the version of SSL or TLS used to encrypt the session and the stage of the handshake.</i> • 826: “The ssl_state keyword can be used to match against state information for an encrypted session. When a rule uses the ssl_state keyword, the rules engine invokes the SSL preprocessor to check traffic for SSL state information.” • See also 491, 597-601. To the extent Sourcefire does not explicitly describe a rule in which the SSL or TLS version is specifically included as a keyword, a person of ordinary skill in the art would have found obvious a rule using the ssl_version keyword to identify traffic using particular versions of SSL or TLS, and either pass or drop those packets. Sourcefire discloses that intrusion rules can be implemented by the SSL preprocessor and the rules engine and provides a specific keyword option to access the SSL and TLS version. Sourcefire, 827-828. For example, Sourcefire discloses that if there is a known vulnerability in SSL version 3, you could use the ssl_version keyword with the sslv2 argument to identify traffic using that version of SSL. <i>Id.</i> The security vulnerabilities of SSLv2 were widely known. Ex. 1016; see also Exs. 1037, 1039. Thus, a person of ordinary skill in the art would have understood that certain SSL/TLS versions had security vulnerabilities (SSLv2, SSLv3, TLS 1.0) and that later TLS versions (TLS 1.1, 1.2) were more secure. These vulnerabilities in early versions of TLS and SSL exposed packets to potential exfiltration, as a malicious actor exploiting the vulnerabilities could circumvent the encryption

Claim Language	Sourcefire
	and gain access to the data. In view of the teachings of Sourcefire, a person of ordinary skill in the art would have been motivated and found it obvious to write an intrusion rule in which the ssl_version keyword is specifically included as a rule in order to detect and drop packets containing certain SSL/TLS versions. Implementing such a rule, would have had the predictable benefit of achieving increased network security by protecting a network against known vulnerabilities. Implementation of the above-described rules involves examining application layer packet headers because SSL, TLS, and HTTP are application layer protocols. Moreover, rule keywords and arguments specifying certain HTTP methods and SSL versions require checking application layer packet headers because HTTP methods and SSL versions are specified in application header fields. For example, a person of ordinary skill in the art would have recognized that HTTP methods are identified within the method field of an HTTP request header.

[1.e], [8.g], [15.e]: responsive to determining the second portion of packets that comprises data corresponding to the second criterion specified by one or more operators specified by the at least one matching packet-filtering rule, [applying/apply], [by the computing system and] to each packet in the second portion of packets [that match the second criterion], at least one packet transformation function configured to prevent an exfiltration operation, wherein the at least one packet transformation function indicates whether each packet in the second portion of packets is allowed to continue toward its destination.

121. Sourcefire discloses [1.e], [8.g], and [15.e].

Claim Language	Sourcefire
[1.e]/[8.g]/[15.e] responsive to determining the second portion of packets that comprises data corresponding to the second criterion specified by one or more operators specified by the at least one matching packet-filtering rule, [applying/apply], [by the computing system and] to each packet in the second portion of packets [that match the second criterion], at least one packet transformation function configured to	Sourcefire discloses responsive to determining (<i>e.g.</i>, “if the packet data matches all the conditions specified in a rule, the rule triggers”) the second portion of packets that comprises data corresponding to the second criterion specified by one or more operators specified by the at least one matching packet-filtering rule, applying, by the computing system and to each packet in the second portion of packets that match the second criterion (<i>e.g.</i>, see [1.d]), at least one packet transformation function (<i>e.g.</i>, “drop,” “pass”) configured to prevent an exfiltration operation (<i>e.g.</i>, to stop certain “HTTP methods” such as “PUT,” “POST,” and “CONNECT” or use of certain unsecure versions of SSL/TLS), wherein the at least one packet transformation function indicates whether each packet in the second portion of packets is allowed to continue toward its destination (<i>e.g.</i>, indicates “drop” or “pass”). Sourcefire discloses rules preventing exfiltration. For example, as discussed above with respect to [1.d], Sourcefire discloses rules including the most common HTTP methods of PUT, POST, and CONNECT as one or more of the rule criteria, and rules including SSL/TLS version as the rule criteria. Sourcefire, 568, 786. Also as discussed above with respect to [1.d], it was known that certain HTTP methods, such as PUT, POST, and CONNECT, and certain SSL/TLS versions, such as SSLv2, had known vulnerabilities and could be used for exfiltration, and it would have been obvious to create rules using the corresponding keywords to identify and drop such packets.⁷

⁷ This is consistent with the ’343 Patent, which subsequently recognized that HTTP methods such as PUT and POST correspond to an exfiltration method. See *e.g.*, ’343 Patent, 7:21-23, 7:55-60, 2:47-58.

Claim Language	Sourcefire
prevent an exfiltration operation, wherein the at least one packet transformation function indicates whether each packet in the second portion of packets is allowed to continue toward its destination.	Sourcefire discloses a packet transformation function that indicates whether each packet that has met the first and second criteria (<i>i.e.</i>, a second portion of packets) is passed or dropped (<i>i.e.</i>, whether it is allowed to continue toward its destination): As discussed in [1.d], Sourcefire discloses that preprocessors and the rule engine compare application header field values of each packet against keywords and arguments specified in the rule arguments and that rule headers specify the rule's action (<i>e.g.</i>, pass, drop). Therefore, responsive to a determination that an application header field value matches a second criterion in the rule, the 3D Sensor applies the rule's action to each packet. • 435: "You can set a rule to Generate Events, Drop and Generate Events, or to Disable individually or you can filter the rules by a variety of factors to select the rules for which you want to modify the state. If you assign inline or inline with fail open interface sets to your IPS detection engines, you can <i>use the Drop and Generate Events rule state in inline intrusion deployments to drop malicious packets.</i>" • 761: "If it is a <i>pass rule</i>, it ignores the traffic. ... For a <i>drop rule</i> in an inline deployment, IPS drops the packet and generates an event." • 765: "Specifying <i>Rule Actions</i> ... Each rule header includes a parameter that <i>specifies the action the system takes when a packet triggers a rule.</i> ... Rules with the <i>action set to pass</i> do not generate an event against, or log the details of, the packet that triggered the rule. ... In an inline deployment, rules with the rule state <i>set to Drop</i> and Generate Events generate an intrusion event against the packet that triggered the rule."

Claim Language	Sourcefire
	• See also 32, 560, 863.

B. Dependent Claims 2, 9, and 16

[2.pre], [9.pre], [16.pre]: The method of claim 1, wherein: / The apparatus of claim 8, wherein / The one or more non-transitory computer-readable media of claim 15, wherein:

122. Sourcefire discloses the preambles of claims 2, 9, and 16. See Claims 1, 8, and 15.

[2.a], [9.a], [16.a]: the first portion of packets comprises a packet header field value indicating a protocol type associated with a particular type of data transfer, and corresponding to the first criterion specified by the at least one matching packet - filtering rule.

123. Sourcefire discloses [2.a], [9.a], and [16.a].

Claim Language	Sourcefire
[2.a]/[9.a]/[16.a] the first portion of packets comprises a packet header field value indicating a protocol type associated with a particular type of data transfer, and corresponding to the first criterion specified by the	Sourcefire discloses the first portion of packets (e.g., see [1.b]) comprises a packet header field value indicating a protocol type (e.g., “IP,” “TCP,” “UDP”) associated with a particular type of data transfer (e.g., “HTTP”), and corresponding to the first criterion specified by the at least one matching packet - filtering rule (e.g., see [1.b]). Sourcefire discloses that the protocol type is indicated in a packet header field value. Sourcefire, 766. As discussed in [1.b], the field corresponds to a first criterion in the packet-filtering rule. A protocol type such as TCP is associated with a type of data transfer such as HTTP because HTTP packets are encapsulated in TCP/IP packets under the OSI model. • 764:

Claim Language	Sourcefire
at least one matching packet - filtering rule.	766: <i>“In each rule header, you must specify the protocol of the traffic the rule inspects.</i> You can specify the following network protocols for analysis: - o ICMP (Internet Control Message Protocol) - o IP (Internet Protocol) - o TCP (Transmission Control Protocol) - o UDP (User Datagram Protocol) Use IP as the protocol type to examine all protocols assigned by IANA, including TCP, UDP, ICMP, IGMP, and many more.”

C. Dependent Claims 3, 10, and 17

[3.pre], [10.pre], [17.pre]: The method of claim 1, wherein: / The apparatus of claim 8, wherein: / The one or more non-transitory computer-readable media of claim 15, wherein:

124. Sourcefire discloses the preambles of claims 3, 10, and 17. See Claims 1, 8, and 15.

[3.a], [10.a], [17.a]: the first portion of packets comprises data indicating a destination port number associated with a particular type of data transfer, and corresponding to the first criterion specified by the at least one matching packet filtering rule.

125. Sourcefire discloses [3.a], [10.a], and [17.a].

Claim Language	Sourcefire												
[3.a]/[10.a]/[17.a] the first portion of packets comprises data indicating a destination port number associated with a particular type of data transfer, and corresponding to the first criterion specified by the at least one matching packet filtering rule.	Sourcefire discloses the first portion of packets (<i>e.g.</i>, see [1.b]) comprises data indicating a destination port number (<i>e.g.</i>, “port number,” “destination ports”) associated with a particular type of data transfer (<i>e.g.</i>, “HTTP”), and corresponding to the first criterion specified by the at least one matching packet filtering rule (<i>e.g.</i>, see [1.b]). Sourcefire discloses that destination ports, which are fields in packet headers, are associated with data transfer protocols such as HTTP. Sourcefire further discloses that “destination ports are usually tied to a service” and that HTTP is such a service. Sourcefire, 1577, 427. This is consistent with the ’343 Patent, which acknowledges that “HTTP and HTTPS” are examples of “data transfer protocols.” ’343 Patent, 8:59-65. Sourcefire additionally discloses rules that specify destination ports. Sourcefire, 764, 768. 425: “You can filter the rules listed in the Rules page by several rule content items. For example, you can quickly retrieve a rule by searching for the rule SID. You can also <i>find all rules that inspect traffic going to a specific destination port.</i>” 427: <table border="1"> <caption>Rule Content Filters (Continued)</caption> <thead> <tr> <th>To use this filter, click...</th> <th>Then...</th> <th>Result</th> </tr> </thead> <tbody> <tr> <td>Destination port</td> <td>Type the destination port to filter by. The port value must be an integer between 1 and 65535 or a port variable.</td> <td>Finds rules that include the specified destination port.</td> </tr> <tr> <td>Rule Overhead</td> <td>Select the amount of rule overhead to filter by.</td> <td>Finds rules with the selected rule overhead.</td> </tr> <tr> <td>Metadata</td> <td>Type the metadata key-value pair to filter by, separated by a space. For example, type <code>metadata:"service http"</code> to locate rules with metadata relating to the HTTP service.</td> <td>Find rules with metadata containing the matching key-value pair.</td> </tr> </tbody> </table> 627: “The TCP protocol defines various states in which connections can exist. Each <i>TCP connection is identified by the source and destination IP addresses and source and destination ports.</i> TCP permits only	To use this filter, click...	Then...	Result	Destination port	Type the destination port to filter by. The port value must be an integer between 1 and 65535 or a port variable.	Finds rules that include the specified destination port.	Rule Overhead	Select the amount of rule overhead to filter by.	Finds rules with the selected rule overhead.	Metadata	Type the metadata key-value pair to filter by, separated by a space. For example, type <code>metadata:"service http"</code> to locate rules with metadata relating to the HTTP service.	Find rules with metadata containing the matching key-value pair.
To use this filter, click...	Then...	Result											
Destination port	Type the destination port to filter by. The port value must be an integer between 1 and 65535 or a port variable.	Finds rules that include the specified destination port.											
Rule Overhead	Select the amount of rule overhead to filter by.	Finds rules with the selected rule overhead.											
Metadata	Type the metadata key-value pair to filter by, separated by a space. For example, type <code>metadata:"service http"</code> to locate rules with metadata relating to the HTTP service.	Find rules with metadata containing the matching key-value pair.											

Claim Language	Sourcefire																											
	one connection with the same connection parameter values to exist at a time.” 680: “Individual data types identify the sensitive data you can detect and generate events on in your specified destination network traffic. You can modify default settings for data type options that specify the following: 0 ... 0 the <i>destination ports to monitor for each data type</i> 764: “Every standard text rule and shared object rule has parameters and arguments that make up the rule header. For example, the following illustrates the parts of a rule header: Type Source IP Operator Destination Port ↓ ↓ ↓ ↓ alert tcp \$EXTERNAL_NET any -> \$HTTP_SERVERS \$HTTP_PORTS ↑ ↑ ↑ Protocol Source Port Destination IP <table><tr><th colspan="3">Rule Header Values</th></tr><tr><th>Rule Header Component</th><th>Example Value</th><th>This Value...</th></tr><tr><td>Action</td><td>alert</td><td>Generates an intrusion event when triggered.</td></tr><tr><td>Protocol</td><td>tcp</td><td>Tests TCP traffic only.</td></tr><tr><td>Source IP Address</td><td>\$EXTERNAL_NET</td><td>Tests traffic coming from any host that is not on your internal network.</td></tr><tr><td>Source Ports</td><td>any</td><td>Tests traffic coming from any port on the originating host.</td></tr><tr><td>Operator</td><td>-></td><td>Tests external traffic (destined for the web servers on your network).</td></tr><tr><td>Destination IP Address</td><td>\$HTTP_SERVERS</td><td>Tests traffic to be delivered to any host specified as a web server on your internal network.</td></tr><tr><td>Destination Ports</td><td>\$HTTP_PORTS</td><td>Tests traffic delivered to an HTTP port on your internal network.</td></tr></table> 768: “Rule headers specify the source and destination ports of the traffic the rule Inspects. You can <i>restrict packet inspection to packets originating from or destined to specific ports.</i>”	Rule Header Values			Rule Header Component	Example Value	This Value...	Action	alert	Generates an intrusion event when triggered.	Protocol	tcp	Tests TCP traffic only.	Source IP Address	\$EXTERNAL_NET	Tests traffic coming from any host that is not on your internal network.	Source Ports	any	Tests traffic coming from any port on the originating host.	Operator	->	Tests external traffic (destined for the web servers on your network).	Destination IP Address	\$HTTP_SERVERS	Tests traffic to be delivered to any host specified as a web server on your internal network.	Destination Ports	\$HTTP_PORTS	Tests traffic delivered to an HTTP port on your internal network.
Rule Header Values																												
Rule Header Component	Example Value	This Value...																										
Action	alert	Generates an intrusion event when triggered.																										
Protocol	tcp	Tests TCP traffic only.																										
Source IP Address	\$EXTERNAL_NET	Tests traffic coming from any host that is not on your internal network.																										
Source Ports	any	Tests traffic coming from any port on the originating host.																										
Operator	->	Tests external traffic (destined for the web servers on your network).																										
Destination IP Address	\$HTTP_SERVERS	Tests traffic to be delivered to any host specified as a web server on your internal network.																										
Destination Ports	\$HTTP_PORTS	Tests traffic delivered to an HTTP port on your internal network.																										

Claim Language	Sourcefire																								
	<table><tr><th colspan="3">Source/Destination Port Syntax</th></tr><tr><th>To Specify...</th><th>Use</th><th>Example</th></tr><tr><td>any port</td><td>any</td><td>any</td></tr><tr><td>a specific port</td><td>the port number</td><td>80</td></tr><tr><td>a range of ports</td><td>a dash between the first and last port number in the range</td><td>80-443</td></tr><tr><td>all ports less than or equal to a specific port</td><td>a dash before the port number</td><td>-21</td></tr><tr><td>all ports greater than or equal to a specific port</td><td>a dash after the port number</td><td>80-</td></tr><tr><td>all ports except a specific port or range of ports</td><td>the ! character before the port, port list, or range of ports you want to negate Note that you can logically use negation with all port designations except <i>any</i>; which if negated would indicate <i>no port</i>.</td><td>!20</td></tr></table>	Source/Destination Port Syntax			To Specify...	Use	Example	any port	any	any	a specific port	the port number	80	a range of ports	a dash between the first and last port number in the range	80-443	all ports less than or equal to a specific port	a dash before the port number	-21	all ports greater than or equal to a specific port	a dash after the port number	80-	all ports except a specific port or range of ports	the ! character before the port, port list, or range of ports you want to negate Note that you can logically use negation with all port designations except <i>any</i> ; which if negated would indicate <i>no port</i> .	!20
Source/Destination Port Syntax																									
To Specify...	Use	Example																							
any port	any	any																							
a specific port	the port number	80																							
a range of ports	a dash between the first and last port number in the range	80-443																							
all ports less than or equal to a specific port	a dash before the port number	-21																							
all ports greater than or equal to a specific port	a dash after the port number	80-																							
all ports except a specific port or range of ports	the ! character before the port, port list, or range of ports you want to negate Note that you can logically use negation with all port designations except <i>any</i> ; which if negated would indicate <i>no port</i> .	!20																							
	831: “Target ports are always HTTP ports.” 1577: “Because destination ports are usually tied to a service, this workflow can help you detect services that are experiencing an uncommonly high volume of alerts. The Destination Port column can also help you identify services that should not be present on your network.” 1608: “With IPS, if you are viewing intrusion events and want to constrain the list to only events where the destination port is 80, click 80 (http)/tcp in the DST Port/ICMP Code column.” - See also 401, 427, 465, 768-769, 870, 1222, 1275.																								

D. Dependent Claims 4, 11, and 18

[4.pre], [11.pre], [18.pre]: The method / apparatus / one or more non-transitory computer-readable media of claim 1/8/15, wherein

126. Sourcefire discloses the preambles of claims 4, 11, and 18. See Claims 1, 8 and 15.

[4.a], [11.a], [18.a]: the application header field value identifies one or more hypertext transfer protocol (HTTP) methods, wherein the second portion of packets comprises a plurality of HTTP packets, wherein the second portion of packets comprises a third portion of packets and a fourth portion of packets, and wherein each packet of the third portion of packets comprises at least one GET method value, and each packet of the fourth portion of packets comprises at least one of a PUT method value, a POST method value, or a CONNECT method value, [the method further comprising:/and wherein the instructions, when executed by the at least one processor cause the apparatus to:/ and wherein the instructions, when executed by the one or more computing devices, cause the one or more computing devices to:]

127. Sourcefire discloses [4.a], [11.a], and [18.a].

Claim Language	Sourcefire
[4.a]/[11.a]/[18.a] the application header field value identifies one or more hypertext transfer protocol (HTTP) methods, wherein the second portion of packets comprises a plurality of HTTP packets, wherein the second portion of packets comprises a third portion of packets and a	Sourcefire discloses wherein the application header field value (e.g., see [1.d]) identifies one or more hypertext transfer protocol (HTTP) methods (e.g., “HTTP Methods”), wherein the second portion of packets (e.g., see [1.d]) comprises a plurality of HTTP packets, wherein the second portion of packets comprises a third portion of packets and a fourth portion of packets, and wherein each packet of the third portion of packets comprises at least one GET method value (e.g., “GET”), and each packet of the fourth portion of packets comprises at least one of a PUT method value (e.g., “PUT”), a POST method value (e.g., “POST”), or a CONNECT method value (e.g., “CONNECT”). As discussed in [1.d], Sourcefire discloses invoking the HTTP Inspect preprocessor to identify HTTP methods such as GET, PUT, POST, and CONNECT in application header field values. A person of ordinary skill in the art would have understood that in a plurality of HTTP method packets, one or more packets may variously specify HTTP method options such as GET, PUT, POST, or CONNECT. Thus, the second

Claim Language	Sourcefire
fourth portion of packets, and wherein each packet of the third portion of packets comprises at least one GET method value, and each packet of the fourth portion of packets comprises at least one of a PUT method value, a POST method value, or a CONNECT method value, [the method further comprising:/and wherein the instructions, when executed by the at least one processor cause the apparatus to:/ and wherein the instructions, when executed by the one or more computing devices, cause	portion of packets would comprise a third portion of packets (<i>i.e.</i>, packets having GET method values) and fourth portions of packets (<i>i.e.</i>, packets having PUT, POST, or CONNECT method values). • 560-561: “The <i>HTTP Inspect preprocessor</i> is responsible for: ○ <i>decoding and normalizing HTTP requests</i> sent to and HTTP responses received from web servers on your network ○ ... ○ making the normalized data available for additional rule processing • 568: “HTTP Methods Specifies <i>HTTP request methods for the preprocessor to inspect in addition to GET and POST.</i> Use a comma to separate multiple values.” • 786: “HTTP content Keyword Options Select this option... <i>HTTP Method</i> To <i>search for content matches in... the request method field, which identifies the action such as GET, PUT, CONNECT,</i> and so on to take on the resource identified in the URI.” • 807:

Claim Language	Sourcefire														
the one or more computing devices to:]	Snort-Specific Post Regular Expression Modifiers (Continued) <table> <tr> <th>Option</th><th>Description</th></tr> <tr> <td>H</td><td>Searches for the content within the header, excluding cookies, of an HTTP request or response message decoded by the HTTP Inspect preprocessor. Note that you cannot use this option in combination with the content keyword HTTP Header option to search the same content. See the HTTP content Keyword Options table on page 785 for more information.</td></tr> <tr> <td>D</td><td>Searches for the content within the header, excluding cookies, of a raw HTTP request or response message decoded by the HTTP Inspect preprocessor. Note that you cannot use this option in combination with the content keyword HTTP Raw Header option to search the same content. See the HTTP content Keyword Options table on page 785 for more information.</td></tr> <tr> <td>M</td><td>Searches for the content within the method field of a normalized HTTP request message decoded by the HTTP Inspect preprocessor; the method field identifies the action such as GET, PUT, CONNECT, and so on to take on the resource identified in the URI. See the content keyword HTTP Method option in the HTTP content Keyword Options table on page 785 for more information.</td></tr> <tr> <td>C</td><td>When the HTTP Inspect preprocessor Inspect HTTP Cookies option is enabled, searches for the normalized content within any cookie in an HTTP request header, and also within any set-cookie in an HTTP response header when the preprocessor Inspect HTTP Responses option is enabled. When Inspect HTTP Cookies is not enabled, searches the entire header, including the cookie or set-cookie data. Note that cookies included in the message body are treated as body content. Note also that you cannot use this option in combination with the content keyword HTTP Cookie option to search the same content. See the HTTP content Keyword Options table on page 785 for more information.</td></tr> <tr> <td>K</td><td>When the HTTP Inspect preprocessor Inspect HTTP Cookies option is enabled, searches for the raw content within any cookie in an HTTP request header, and also within any set-cookie in an HTTP response header when the preprocessor Inspect HTTP Responses option is enabled. When Inspect HTTP Cookies is not enabled, searches the entire header, including the cookie or set-cookie data. Note that cookies included in the message body are treated as body content. Note also that you cannot use this option in combination with the content keyword HTTP Raw Cookie option to search the same content. See the HTTP content Keyword Options table on page 785 for more information.</td></tr> <tr> <td>S</td><td>Searches the 3-digit status code in an HTTP response. See the content keyword HTTP Status Code option in the HTTP content Keyword Options table on page 785 for more information.</td></tr> </table>	Option	Description	H	Searches for the content within the header, excluding cookies, of an HTTP request or response message decoded by the HTTP Inspect preprocessor. Note that you cannot use this option in combination with the content keyword HTTP Header option to search the same content. See the HTTP content Keyword Options table on page 785 for more information.	D	Searches for the content within the header, excluding cookies, of a raw HTTP request or response message decoded by the HTTP Inspect preprocessor. Note that you cannot use this option in combination with the content keyword HTTP Raw Header option to search the same content. See the HTTP content Keyword Options table on page 785 for more information.	M	Searches for the content within the method field of a normalized HTTP request message decoded by the HTTP Inspect preprocessor; the method field identifies the action such as GET, PUT, CONNECT, and so on to take on the resource identified in the URI. See the content keyword HTTP Method option in the HTTP content Keyword Options table on page 785 for more information.	C	When the HTTP Inspect preprocessor Inspect HTTP Cookies option is enabled, searches for the normalized content within any cookie in an HTTP request header, and also within any set-cookie in an HTTP response header when the preprocessor Inspect HTTP Responses option is enabled. When Inspect HTTP Cookies is not enabled, searches the entire header, including the cookie or set-cookie data. Note that cookies included in the message body are treated as body content. Note also that you cannot use this option in combination with the content keyword HTTP Cookie option to search the same content. See the HTTP content Keyword Options table on page 785 for more information.	K	When the HTTP Inspect preprocessor Inspect HTTP Cookies option is enabled, searches for the raw content within any cookie in an HTTP request header, and also within any set-cookie in an HTTP response header when the preprocessor Inspect HTTP Responses option is enabled. When Inspect HTTP Cookies is not enabled, searches the entire header, including the cookie or set-cookie data. Note that cookies included in the message body are treated as body content. Note also that you cannot use this option in combination with the content keyword HTTP Raw Cookie option to search the same content. See the HTTP content Keyword Options table on page 785 for more information.	S	Searches the 3-digit status code in an HTTP response. See the content keyword HTTP Status Code option in the HTTP content Keyword Options table on page 785 for more information.
Option	Description														
H	Searches for the content within the header, excluding cookies, of an HTTP request or response message decoded by the HTTP Inspect preprocessor. Note that you cannot use this option in combination with the content keyword HTTP Header option to search the same content. See the HTTP content Keyword Options table on page 785 for more information.														
D	Searches for the content within the header, excluding cookies, of a raw HTTP request or response message decoded by the HTTP Inspect preprocessor. Note that you cannot use this option in combination with the content keyword HTTP Raw Header option to search the same content. See the HTTP content Keyword Options table on page 785 for more information.														
M	Searches for the content within the method field of a normalized HTTP request message decoded by the HTTP Inspect preprocessor; the method field identifies the action such as GET, PUT, CONNECT, and so on to take on the resource identified in the URI. See the content keyword HTTP Method option in the HTTP content Keyword Options table on page 785 for more information.														
C	When the HTTP Inspect preprocessor Inspect HTTP Cookies option is enabled, searches for the normalized content within any cookie in an HTTP request header, and also within any set-cookie in an HTTP response header when the preprocessor Inspect HTTP Responses option is enabled. When Inspect HTTP Cookies is not enabled, searches the entire header, including the cookie or set-cookie data. Note that cookies included in the message body are treated as body content. Note also that you cannot use this option in combination with the content keyword HTTP Cookie option to search the same content. See the HTTP content Keyword Options table on page 785 for more information.														
K	When the HTTP Inspect preprocessor Inspect HTTP Cookies option is enabled, searches for the raw content within any cookie in an HTTP request header, and also within any set-cookie in an HTTP response header when the preprocessor Inspect HTTP Responses option is enabled. When Inspect HTTP Cookies is not enabled, searches the entire header, including the cookie or set-cookie data. Note that cookies included in the message body are treated as body content. Note also that you cannot use this option in combination with the content keyword HTTP Raw Cookie option to search the same content. See the HTTP content Keyword Options table on page 785 for more information.														
S	Searches the 3-digit status code in an HTTP response. See the content keyword HTTP Status Code option in the HTTP content Keyword Options table on page 785 for more information.														

[4.b], [11.b], [18.b]: [allowing/ allow] each packet of the third portion of packets to continue toward its respective destination; and

128. Sourcefire discloses [4.b], [11.b], and [18.b].

Claim Language	Sourcefire
[4.b]/[11.b]/[18.b] [allowing/allow] each packet of the third portion of packets to continue toward its respective destination; and	Sourcefire discloses allowing (e.g., “pass”) each packet of the third portion of packets (e.g., packets indicating “GET”) to continue toward its respective destination (e.g., “pass”). As discussed in [1.d], [1.e], and [4.a], Sourcefire discloses invoking the HTTP Inspect preprocessor to identify HTTP methods such as GET, PUT, POST, and CONNECT in application header field values, and passing or dropping (depending on how the rule is configured) packets that match the specified criteria. At a minimum, it would have been obvious in view of these disclosures to write a rule using the HTTP Method option of the HTTP content keyword as part of the application-layer rule criteria to invoke the HTTP inspect preprocessor to identify a packet using the “GET” HTTP method (third portion of packets) and to allow such a packet to continue towards its destination. A person of ordinary skill in the art would have been motivated to write such a rule because it was well-known that the HTTP GET option is typically not exfiltration related. Ex. 1019, 13:65-67. Moreover, as discussed in [1.e], writing such a rule merely involves selecting from a finite number of criteria that are relevant to a selected network vulnerability, and would yield predictable results of increased network security. Additionally, implementing the rule would involve nothing more than applying a known technique to a known device.

[4.c], [11.c], [18.c]: [blocking/block] each packet of the fourth portion of packets from continuing toward its respective destination.

129. Sourcefire discloses [4.c], [11.c], and [18.c].

Claim Language	Sourcefire
[4.c]/[11.c]/[18.c] [blocking/block] each packet of the fourth portion of packets from continuing toward its respective destination.	Sourcefire discloses blocking (e.g., “drop”) each packet of the fourth portion of packets from continuing toward its respective destination. Similarly, at a minimum, it would have been obvious in view of these disclosures to write a rule using the HTTP Method option of the HTTP content keyword as part of the application-layer rule criteria to invoke the HTTP inspect preprocessor to identify a packet using the “PUT,” “POST,” or “CONNECT” HTTP methods (fourth portion of packets) and to block such a packet from continuing towards its destination. A person of ordinary skill in the art would have been motivated to write such a rule because it was well known that PUT, POST, and CONNECT methods transmit data outside the network, as discussed in [1.d]. Sourcefire, 934; Moreover, as discussed in [1.e], writing such a rule merely involves selecting from a finite number of criteria that are relevant to a selected network vulnerability, and would yield predictable results of increased network security. Additionally, implementing the rule would involve nothing more than applying a known technique to a known device.

E. Dependents Claim 5 and 12

[5.pre], [12.pre]: The method of claim 1, the method further comprising: / The apparatus of claim 8, wherein the instructions, when executed by the one or more computing devices, cause the apparatus to:

130. Sourcefire discloses the preambles of claims 5 and 12. See Claims 1 and 8.

[5.a], [12.a]: [allowing/ allow], based on a determination that the application header field value indicates a request for [specified resource data/data from a specified resource], each packet of the second portion of packets indicating a request for [specified resource data/data from a specified resource] to continue toward its respective destination; and

131. Sourcefire discloses [5.a] and [12.a].

Claim Language	Sourcefire
[5.a]/[12.a] [allowing/ allow], based on a determination that the application header field value indicates a request for [specified resource data/data from a specified resource], each packet of the second portion of packets indicating a request for [specified resource data/data from a specified resource] to continue toward its respective destination; and	Sourcefire discloses allowing (e.g., “pass”), based on a determination that the application header field value (e.g., see [1.d]) indicates a request for specified resource data (e.g., “GET”), each packet of the second portion of packets indicating a request for specified resource data (e.g., packets indicating “GET”) to continue toward its respective destination (e.g., “pass”). See [4.a]/[11.a] and [4.b]/[11.b].

[5.b], [12.b]: [blocking/ block], based on a determination that the application header field value indicates a request to submit data to be processed by a specified resource, each packet of the second portion of packets indicating a request to submit data to be processed by [the/a] specified [resource] from continuing toward its respective destination.

132. Sourcefire discloses [5.b] and [12.b].

Claim Language	Sourcefire
[5.b]/[12.b] [blocking/ block], based on a determination that the application header field value indicates a request to submit data to be processed by a specified resource, each packet of the second portion of packets indicating a request to submit data to be processed by [the/a] specified [resource] from continuing toward its respective destination.	Sourcefire discloses blocking (e.g., “drop”), based on a determination that the application header field value (e.g., see [1.d]) indicates a request to submit data to be processed by a specified resource (e.g., “PUT,” POST,” “CONNECT”), each packet of the second portion of packets indicating a request to submit data to be processed by the specified resource (e.g., packets indicating “PUT,” “POST,” “CONNECT”) from continuing toward its respective destination (e.g., “drop”). See [4.a]/[11.a] and [4.c]/[11.c]

F. Dependent Claims 6, 13, and 19

[6.pre], [13.pre], [19.pre]: The method of claim 1, further comprising / The apparatus of claim 8, wherein the instructions, when executed by the one or more computing devices, cause the apparatus to / The one or more non-transitory computer-readable media of claim 15, further comprising

133. Sourcefire discloses the preambles of claims 6, 13, and 19. See Claims 1, 8, and 15.

[6.a], [13.a], [19.a]: [dropping/drop] packets associated with a particular type of data transfer responsive to a determination that the packets associated with the particular type of data transfer comprise data corresponding to a particular transport layer security (TLS) version value

134. Sourcefire discloses [6.a], [13.a], and [19.a].

Claim Language	Sourcefire
[6.a]/[13.a]/[19.a] [dropping/drop] packets associated with a particular type of data transfer responsive to a determination that the packets associated with the particular type of data transfer comprise data corresponding to a particular transport layer	Sourcefire discloses dropping packets (e.g., “drop”) associated with a particular type of data transfer (e.g., “SSL”) responsive to a determination (e.g., “if the packet data matches all the conditions specified in a rule, the rule triggers”) that the packets associated with the particular type of data transfer comprise data corresponding to a particular transport layer security (TLS) version value (e.g., “TLS Version 1.0,” “TLS Version 1.1,” “TLS Version 1.2”). SSL is a type of data transfer. As discussed above with respect to [1.d] and [1.e], Sourcefire discloses a rule using an SSL keyword to identify SSL or TLS versions in packets using the SSL preprocessor to drop or allow packets, and passing or dropping (depending on how the rule is configured) packets that match the specified criteria.

Claim Language	Sourcefire														
security (TLS) version value	827-828: “The ssl_version keyword can be used to match against version information for an encrypted session. When a rule uses the ssl_version keyword, the rules engine invokes the SSL preprocessor to check traffic for SSL version information. For example, if you know there is a buffer overflow vulnerability in SSL version 2, you could use the ssl_version keyword with the sslv2 argument to identify traffic using that version of SSL. Use a comma-separated list to specify multiple arguments for the SSL version. When you list multiple arguments, IPS evaluates them using the OR operator. For example, if you wanted to identify any encrypted traffic that was not using SSLv2, you could add ssl_version:ssl_v3, tls1.0, tls1.1, tls1.2 to a rule. The rule would evaluate any traffic using SSL Version 3, TLS Version 1.0, TLS Version 1.1, or TLS Version 1.2.” ssl_version Arguments (Continued) <table> <tr> <th>Argument</th><th>Purpose</th></tr> <tr> <td>tls1.0</td><td>Matches against traffic encoded using Transport Layer Security (TLS) Version 1.0.</td></tr> <tr> <td>tls1.1</td><td>Matches against traffic encoded using Transport Layer Security (TLS) Version 1.1.</td></tr> <tr> <td>tls1.2</td><td>Matches against traffic encoded using Transport Layer Security (TLS) Version 1.2.</td></tr> </table> ssl_version Arguments <table> <tr> <th>Argument</th><th>Purpose</th></tr> <tr> <td>sslv2</td><td>Matches against traffic encoded using Secure Sockets Layer (SSL) Version 2.</td></tr> <tr> <td>sslv3</td><td>Matches against traffic encoded using Secure Sockets Layer (SSL) Version 3.</td></tr> </table> As discussed in [1.d], at a minimum, it would have been obvious in view of Sourcefire to write a rule using the ssl_version keyword as part of the application-layer rule criteria to invoke the SSL preprocessor to identify a packet using particular transport layer security (TLS) version and to drop such a packet.	Argument	Purpose	tls1.0	Matches against traffic encoded using Transport Layer Security (TLS) Version 1.0.	tls1.1	Matches against traffic encoded using Transport Layer Security (TLS) Version 1.1.	tls1.2	Matches against traffic encoded using Transport Layer Security (TLS) Version 1.2.	Argument	Purpose	sslv2	Matches against traffic encoded using Secure Sockets Layer (SSL) Version 2.	sslv3	Matches against traffic encoded using Secure Sockets Layer (SSL) Version 3.
Argument	Purpose														
tls1.0	Matches against traffic encoded using Transport Layer Security (TLS) Version 1.0.														
tls1.1	Matches against traffic encoded using Transport Layer Security (TLS) Version 1.1.														
tls1.2	Matches against traffic encoded using Transport Layer Security (TLS) Version 1.2.														
Argument	Purpose														
sslv2	Matches against traffic encoded using Secure Sockets Layer (SSL) Version 2.														
sslv3	Matches against traffic encoded using Secure Sockets Layer (SSL) Version 3.														

G. Dependent Claims 7, 14, and 20

[7.pre], [14.pre], [20.pre]: The method of claim 1, further comprising / The apparatus of claim 8, wherein the instructions, when executed by the one or more computing devices, cause the apparatus to / The one or more non-transitory computer-readable media of claim 15, further comprising.

135. Sourcefire discloses the preambles of claims 7, 14, and 20. See Claim 1, 8, and 15.

[7.a], [14.a], [20.a]: [dropping/drop] packets associated with a particular type of data transfer responsive to a determination that the packets associated with the particular type of data transfer comprise data corresponding to a particular real-time transport protocol value.

136. Sourcefire in view of Mantripragada renders obvious [7.a], [14.a], and [20.a].

137. U.S. Pat. App. Publication 2009/0103524 (“Mantripragada”) is directed to enforcing security rules for Voice-over-IP (VoIP) and Unified Communications (UC) technology in IP communication networks. Mantripragada, Abstract, ¶¶2-3. For example, Mantripragada teaches monitoring Real-Time Transport Protocol (RTP) streams to ensure that various RTP values are not tampered with while the RTP session is active. *Id.* ¶¶69-70. RTP values that are checked include TP protocol header sequence number (16 bits), RTP protocol header timestamp (32 bits), RTP protocol header synchronization source identification (32 bits), version (2 bits), padding (1 bit), extension (1 bit), CSRC (Contributing Source) count (4 bits), marker

(1 bit), and payload type (7 bits). *Id.* ¶70. A person of ordinary skill in the art would have understood that each of these variables is located within the header of an RTP packet. Ex. 1012, 24:55-67 (showing the structure of an RTP header). Mantripragada further teaches rejecting (*i.e.*, dropping) packets that have RTP values that are out of compliance. *Id.* ¶70 (“Media streams that are not compliant with existing session flows ... are rejected....”).

138. Thus, Mantripragada discloses dropping packets (*e.g.*, “reject[ing]”) associated with a particular type of data transfer (*e.g.*, associated with “RTP ... streams”) responsive to a determination (*e.g.*, “are not compliant with existing session flows”) that the packets associated with the particular type of data transfer comprise data corresponding to a particular real-time transport protocol value (*e.g.*, corresponding to RTP values, including “version” and Contributing Source,” that are “out of compliance”).

139. Sourcefire and Mantripragada are in the same field of art and are analogous art to the ’343 Patent—all are in the same fields of packet filtering and network security. *E.g.*, Sourcefire, 33-35; Mantripragada, Abstract, ¶¶2, 69-70; ’343, Abstract, 1:18-51.

140. A person of ordinary skill in the art would have been motivated to apply Mantripragada’s teachings of dropping packets based on non-compliant RTP values in implementing an intrusion rule of Sourcefire in order to prevent attacks on RTP

streams. For example, Mantripragada discloses that “[a]ttacks against RTP are applicable to virtually any VoIP or SIP environment” and identifies “Man-in-the-Middle (MITM) attacks” as an example. Mantripragada ¶69. The solution taught by Mantripragada—dropping packets based on out of compliant RTP values—prevents those attacks from succeeding by “ensur[ing] that [the packets] are not tampered while the session is active.” *Id.*

141. A person of ordinary skill in the art would have had a reasonable expectation of success in using Mantripragada’s teachings of dropping packets based on non-compliant RTP values in implementing an intrusion rule of Sourcefire, and would have known that such a combination (yielding the claimed limitations) would predictably work and provide the expected functionality. For example, Sourcefire discloses decoding UDP traffic (Sourcefire, 258)—the transport layer that RTP sits atop—and discloses application layer decoders and preprocessors (*id.*, 256), which would be used to decode RTP packets. Indeed, the 3D Sensor is able to detect “Real-time Transport Protocol.” *Id.*, 2034. The device will also “identify a session” of a UDP stream (*id.*, 645-46), and the “flow” keyword is used to “select packets for inspection by a rule based on session characteristics” (*id.*, 820). Sourcefire further discloses that “you can create your own custom standard text rules.” *Id.*, 861. To the extent a specific rule keyword is not pre-configured to target the RTP values identified by Mantripragada, Sourcefire discloses a “byte_extract keyword” that

reads a specified number of bytes from a packet with a specified offset (*id.*, 840-41), which a person of ordinary skill in the art would have known to use in conjunction with RTP's standardized, fixed-size header values (Ex. 1012, 24:55-67) to reliably extract the required values.

X. Secondary Considerations

142. I understand that Patent Owner raised evidence of alleged secondary considerations in IPRs challenging the related '552 and '193 Patents, and that the Board rejected Patent Owner's evidence. Ex. 1042, 56-62. I further understand that the Federal Circuit affirmed the Board's determination, holding that "the objective indicia of nonobviousness were not entitled to substantial weight." Ex. 1043, 9.

143. At this time, after review of the documents and materials cited herein, such as the prosecution history of the '343 Patent, I have found no evidence of secondary considerations or objective factors of non-obviousness relevant to the Challenged Claims, and it is my expert opinion that Challenged Claims of the '343 Patent are obvious over the prior art. To the extent Patent Owner alleges secondary considerations in this proceeding, it is my expert opinion that they would not outweigh the strong evidence of obviousness of the '343 Patent claims, as set forth herein. I have been informed that if the Patent Owner presents in this proceeding evidence of objective factors that it argues support a finding of non-obviousness, I may be asked to review such evidence and to formulate an opinion and reserve the

right to address any such evidence and/or arguments that Patent Owner may raise in this regard.

XI. Conclusion

144. In summary, I have concluded that each of the Challenged Claims is unpatentable as obvious under 35 U.S.C. § 103 in light of the prior art references as discussed above and the knowledge of one of ordinary skill in the art, as described in this declaration.

145. Specifically, in my expert opinion, for at least the above reasons:

- Sourcefire renders obvious claims 1-6, 8-13, and 15-19.
- Sourcefire in view of Mantripragada renders obvious claims 7, 14, and 20.

146. Furthermore, as discussed in Section II above, I have used my background and experience in forming my expert opinions that the Challenged Claims are unpatentable as obvious over the prior art.

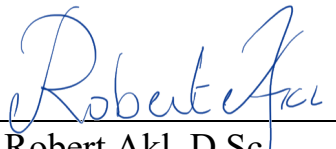
147. Even if it were argued that any further disclosure is required for a limitation in the Challenged Claims that I have identified above as having been disclosed by the cited references, a person of ordinary skill in the art would have found that limitation obvious to include based on the same disclosure and analysis I have identified above.

148. I currently hold the opinions expressed in this declaration. But my analysis may continue, and I reserve the right to supplement my expert opinions in

the future to respond to any arguments that Patent Owner or its expert(s) may raise and to take into account new information as it becomes available to me.

I declare under penalty of perjury that the foregoing is true and correct. I declare that all statements made herein of my knowledge are true, and 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 so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code.

Dated: July 6, 2021

By: 
Dr. Robert Ak1, D.Sc.
Dallas, Texas

Robert Akl, D.Sc.



Professional Summary

Dr. Akl has over 25 years of industry and academic experience. He is currently a Tenured Associate Professor at the University of North Texas and a Senior Member of IEEE. He has designed, implemented, and optimized both hardware and software aspects of several wireless communication systems for CDMA, Wi-Fi, and sensor networks. Dr. Akl has broad expertise in wireless communication, Bluetooth, CDMA/WCDMA network optimization, GSM, LTE, VoIP, telephony, computer architecture, and computer networks. He is a very active researcher and is well published and cited. He has been awarded many research grants by leading companies in the industry and the National Science Foundation. He has developed and taught over 100 courses in his field. Dr. Akl has received several awards and commendation for his work, including the 2008 IEEE Professionalism Award and was the winner of the 2010 Tech Titan of the Future Award.

Dr. Akl has extensive experience with patents in the wireless and networking industry. In the past ten years, he has worked as a technical expert in dozens of patent related matters, involving thousands of hours of research, investigation, and study. He has repeatedly been qualified as an expert by Courts, and has provided numerous technology tutorials to Courts, and given testimony by deposition and at trial. He has worked with companies large and small, both for and against the validity and infringement of patents, and has also helped counsel and Courts to understand technology that often seems complex. In doing so, he has become familiar with, and actively worked with, the legal principles that underlie patentability and validity and claim interpretation in the wireless and networking industries.

Areas of Expertise

2G, 3G, 4G, 5G, CDMA/WCDMA, GPS, GSM, UMTS, LTE, Ad-hoc Networks, Antenna Design, Bluetooth, Call Admission Control, Channel Coding, Communication Interfaces and Standards, Compression, Computer Architecture, MIMO Systems, Multi-cell Network Optimization, Network Security, Packet-networks, Telephony, VoIP, Wi-Fi, Wireless Communication, Wireless Sensors.

Education

<u>Year</u>	<u>College/University</u>	<u>Degree</u>	<u>GPA</u>
2000	Washington University in Saint Louis	D.Sc. in Electrical Engineering	4.0 / 4.0
1996	Washington University in Saint Louis	M.Sc. in Electrical Engineering	4.0 / 4.0
1994	Washington University in Saint Louis	B.Sc. in Electrical Engineering	4.0 / 4.0
1994	Washington University in Saint Louis	B.Sc. in Computer Science	4.0 / 4.0

Graduated *summa cum laude* and ranked first in undergraduate class.

Dissertation: “Cell Design to Maximize Capacity in Cellular Code Division Multiple Access (CDMA) Networks.” Advisors: Dr. Manju Hegde and Dr. Paul Min.

Litigation Support and Expert Witness Experience

- L1. 2021 **Ropes & Gray LLP**
Case: Palo Alto Networks Inc. v. Centripetal Networks, Inc.
IPR2021-01155
Matter: *Inter Partes* Review, network security systems
Project: Declarations to support IPR petition
- L2. 2021 **Devlin Law Firm**
Case: CDN Innovations, LLC v. Grande Communications Networks, LLC
District of Delaware, Case No. 4:20-cv-653-SDJ
Matter: Patent infringement, telecommunication systems
Project: Declaration to support infringement contentions
- L3. 2021 **Banner Witcoff**
Case: Sisvel International S.A., 3G Licensing S.A. v. ZTE (USA) Inc. et al.
Northern district of Texas, Case No. 3:20-cv-01289-M
Matter: Patent infringement, wireless communication systems
Project: Declaration to support claim construction, deposition
- L4. 2021 **Paul Hastings LLP**
Case: G. Holdings Ltd. v. Samsung Electronics Co., et al.
Eastern district of Texas, Marshall division, Case No. 2:20-cv-00342-JRG
Matter: Patent infringement, electronic payment systems
Project: Declaration to support claim construction
- L5. 2021 **Morgan, Lewis & Bockius LLP**
Case: Aruba Networks, LLC and Hewlett Packard Enterprise Company v. SIPCO, LLC
IPR2021-00787
Matter: *Inter Partes* Review, wireless communication systems
Project: Declaration to support IPR petition
- L6. 2021 **Jenner & Block LLP**
Case: Virentem Ventures LLC D/B/A Enounce v. TiVo Corp and Xperi Holding Corporation
District of Delaware, Case No. 20-787-MN
Matter: Patent infringement, telecommunication systems
Project: Declaration to support claim construction

- L7. 2021 **Carter Arnett**
Case: Correct Transmission LLC v. Adtran, Inc. and Juniper Networks, Inc.
Western district of Texas, Waco division, Case No. 6:20-cv-669-ADA
Matter: Patent infringement, telecommunication systems
Project: Declaration to support claim construction
- L8. 2021 **Fish & Richardson, P.C.**
Case: Quectel Wireless Solutions Co. Ltd. v. Koninklijke Philips N.V.
IPR2021-00558, IPR2021-00559, IPR2021-00561
Matter: *Inter Partes* Review, wireless communication systems
Project: Three declarations to support 3 IPR petitions
- L9. 2021 **Fish & Richardson, P.C.**
Case: Samsung Electronics Co. Ltd. v. Ericsson Inc.
IPR2021-00447, IPR2021-00588, IPR2021-00613, IPR2021-00614,
IPR2021-00643, IPR2021-00645, IPR2021-00684
Matter: *Inter Partes* Review, wireless communication systems
Project: Seven declarations to support 7 IPR petitions
- L10. 2020 **Kilpatrick Townsend & Stockton LLP**
Case: GREE Inc. v. Supercell Oy
Eastern district of Texas, Marshall division, Case No. 2:19-cv-00413-
JRG-RSP
Matter: Patent infringement, mobile gaming
Project: Source code review, infringement expert report, supplemental expert
report, validity expert report, deposition
- L11. 2020 **Banner Witcoff**
Case: Sisvel International S.A., 3G Licensing S.A. v. ZTE (USA) Inc. et al.
Northern district of Texas, Case No. 3:19-cv-01694-N
Matter: Patent infringement, wireless communication systems
Project: Declaration to support claim construction, deposition
- L12. 2020 **Fish & Richardson, P.C.**
Case: Cellco Partnership D/B/A Verizon Wireless v. Huawei Technologies
Co., Ltd
IPR2020-01352, IPR202-01356, IPR2020-01357
Matter: *Inter Partes* Review, network communication systems
Project: Three declarations to support three Patent Owner responses
- L13. 2020 **Kilpatrick Townsend & Stockton LLP**
Case: GREE Inc. v. Supercell Oy
Eastern district of Texas, Marshall division, Case No. 2:20-cv-00113-
JRG-RSP
Matter: Patent infringement, mobile gaming

- Project: Source code review, declaration supporting claim construction, infringement expert report, validity expert report, supplemental report, deposition
- L14. 2020 **Kilpatrick Townsend & Stockton LLP**
Case: GREE Inc. v. Supercell Oy
Eastern district of Texas, Marshall division, Case No. 2:19-cv-00200-JRG-RSP, Case No. 2:19-cv-00237-JRG-RSP, 2:19-cv-00310-JRG-RSP; 2:19-cv-00311-JRG-RSP
Matter: Patent infringement, mobile gaming
Project: Source code review, four infringement expert reports, two supplemental infringement expert reports, four validity expert reports, two supplemental expert reports, two second supplemental expert reports, 3-day deposition, jury trial testimony
- L15. 2020 **Sheridan Ross P.C.**
Case: Justservice.net LLC v. Dropbox, Inc.
Western district of Texas, Waco division, Case No. 6:20-CV-00070-ADA
Matter: Patent infringement, computer systems and networking
Project: Declaration regarding claim construction
- L16. 2020 **Perkins Coie LLP**
Case: Huizhou TCL Mobile Communication Co. Ltd., TCT Mobile (US) Inc., and TCL Mobile Communication (HK) Co., Ltd. v. Wi-LAN Inc.
Matter: *Ex Partes* Reexamination, QoS enhancements for wireless IP networks
Project: Declaration to support Requesters
- L17. 2020 **Perkins Coie LLP**
Case: Intel Corporation v. UNM Rainforest Innovations
IPR2020-01576, IPR2020-01578, IPR2020-*to be assigned*
Matter: *Inter Partes* Review, wireless broadband
Project: Three declarations to support three IPR petitions
- L18. 2020 **Calfee, Halter & Griswold LLP**
Case: Motorola Solutions, Inc. v. Hytera Communications Corp. Ltd. et al.
Northern district of Illinois, Case No. 1:17-cv-01972
Matter: Patent infringement, two-way radios
Project: Declaration regarding claim construction, deposition
- L19. 2020 **Fish & Richardson, P.C.**
Case: Huawei Technologies Co., Ltd et al. v. Verizon Communications, Inc. et al.
Western district of Texas, Waco division, Case No. 6-20-cv-00090
Matter: Patent infringement, video communication

- Project: Source code review, declaration regarding claim construction, infringement expert report, validity expert report, deposition
- L20. 2020 **Kilpatrick Townsend & Stockton LLP**
Case: GREE Inc. v. Supercell Oy
Eastern district of Texas, Marshall division, Case No. 2:19-cv-00070-JRG-RSP, 2:19-cv-00071-JRG-RSP
Matter: Patent infringement, mobile gaming
Project: Source code review, two infringement expert reports, two supplemental infringement expert reports, two second supplemental infringement expert reports, two rebuttal expert reports on validity, two-day deposition, seven declarations supporting Gree's opposition to Supercell's motions for summary judgement, jury trial testimony
- L21. 2020 **Cooley LLP**
Case: Saint Lawrence Communications, LLC v. Amazon.com, Inc., et al.
Eastern district of Texas, Marshall division, Case No. 2:19-cv-00027-JRG
Matter: Patent infringement, AMR-WB, speech compression, coding and decoding
Project: Invalidity expert report
- L22. 2020 **Prince Lobel Tye LLP**
Case: Intellectual Ventures I and II LLC v. VMware, Inc.
Western district of Texas, Austin division, Case No. 1:19-cv-01075-ADA
Matter: Patent infringement, networking systems
Project: Declaration to support claim construction
- L23. 2020 **Faegre Baker Daniels LLP**
Case: CommScope, Inc. v. Rosenberger Technology, et al.
District of New Jersey, Case No. 19-cv-15962-MCA-LDW
Matter: Trade secret software, base station antenna design
Project: Declaration, deposition
- L24. 2020 **Ropes & Gray LLP**
Case: Canon, Inc. v. TCL Electronics Holdings Ltd., et al.
Eastern district of Texas, Marshall division, Case No. 2:18-cv-546-JRG
Matter: Patent infringement, communication interfaces
Project: Source code review, declaration to support claim construction, deposition

- L25. 2019 **Perkins Coie LLP**
Case: Huizhou TCL Mobile Communication Co. Ltd., TCT Mobile (US) Inc., and TCL Mobile Communication (HK) Co., Ltd. v. Wi-LAN Inc.
IPR2020-00302, IPR2020-00303
Matter: *Inter Partes* Review, QoS enhancements for wireless IP networks
Project: Two declarations to support two IPR petitions
- L26. 2019 **K & L Gates LLP**
Case: EVS CODEC Technologies, LLC and Saint Lawrence Communications, LLC v. ZTE Corporation, et al.
Northern district of Texas, Dallas division, Case No. 3:19-cv-00385-MBH
Matter: Patent infringement, EVS, speech compression, coding and decoding
Project: Invalidity expert report
- L27. 2019 **Feinberg Day Alberti Lim & Belloli LLP**
Case: Uniloc 2017 LLC v. AT&T Mobility LLC, et al.
Eastern district of Texas, Marshall division, Case No. 2:18-cv-00514-JRG
Matter: Patent infringement, wireless frequency bands and devices
Project: Two declarations to support claim construction, deposition
- L28. 2019 **Ropes & Gray LLP**
Case: Huawei Technologies Co. Ltd. v. Harris Global Communications, Inc.
IPR2019-01512, IPR2019-01631
Matter: *Inter Partes* Review, routing and security in wireless networks
Project: Two declarations to support two IPR petitions
- L29. 2019 **Ropes & Gray LLP**
Case: Harris Corporation v. Huawei Device USA, Inc. et al.
Eastern district of Texas, Marshall division, Case No. 2:18-cv-00439-JRG
Matter: Patent infringement, routing and security in wireless networks
Project: Declaration to support claim construction
- L30. 2019 **Erise IP**
Case: Semcon IP Inc. v. ASUSTeK Computer Inc.
Eastern district of Texas, Marshall division, Case No. 2:18-cv-00193-JRG
Matter: Patent infringement, adaptive power control
Project: Non-infringement expert report

- L31. 2019 **Cooley LLP**
Case: Facebook Inc. v. BlackBerry Corp. et al.
Northern District of California, Oakland division, Case No. 4:18-cv-05434-JSW
Matter: Patent infringement, mobile computing
Project: Declaration to support claim construction
- L32. 2019 **Sidley Austin LLP**
Case: Semcon IP Inc. v. Amazon.com, Inc.
Eastern district of Texas, Marshall division, Case No. 2:18-cv-00192-JRG
Matter: Patent infringement, adaptive power control
Project: Expert report regarding patent marking, rebuttal report regarding patent marking, deposition
- L33. 2019 **Oblon, McClelland, Maier & Neustadt, LLP**
Case: MV3 Partners LLC v. Roku, Inc.
Western district of Texas, Waco division, Case No. 6:18-cv-308-ADA
Matter: Patent infringement, mobile set top box
Project: Declaration to support claim construction, deposition, Markman hearing testimony
- L34. 2019 **Banner & Witcoff, LTD.**
Case: Kathrein USA, Inc. v. Fractus S.A.
IPR2019-00954, IPR2019-00955, IPR2019-00956, IPR2019-00957
Matter: *Inter Partes* Review, multiband antenna arrays
Project: Four declarations to support four IPR petitions
- L35. 2019 **Fish & Richardson, P.C.**
Case: LG Electronics Inc. v. Saint Lawrence Communications LLC
Southern district of New York, Case No. 1:18-cv-11082-DLC
Matter: Patent infringement, EVS, speech compression, coding and decoding
Project: Declaration relating to motion for summary judgment, expert report, deposition
- L36. 2019 **Ropes & Gray LLP**
Case: SIPCO, LLC v. Emerson Electric Co.
In the Matter of Certain Wireless Mesh Networking Products and Related Components Thereof, ITC Investigation No. 337-TA-1131
Matter: Patent infringement, links in wireless networks and remote monitoring
Project: Source code review, declaration to support claim construction, invalidity expert report, rebuttal expert report regarding non-infringement and no domestic industry

- L37. 2019 **Fish & Richardson, P.C.**
Case: Maxell Ltd. v. Huawei Technologies Co. Ltd., ZTE, et al.
Eastern district of Texas, Texarkana division, Case No. 5:18-cv-0033-RWS
Matter: Patent infringement, portable computing devices
Project: Declaration regarding claim construction
- L38. 2019 **Ropes & Gray LLP**
Case: Emerson Electric Co. v. SIPCO, LLC
IPR2019-00548, IPR2019-00549
Matter: *Inter Partes* Review, routing in wireless networks
Project: Two declarations to support two IPR petitions
- L39. 2018 **Mishcon de Reya New York LLP**
Case: ChanBond, LLC v. Cox Communications, Inc.
District of Delaware, Case No. 1:15-cv-00849-RGA
Matter: Patent infringement, wideband signal distribution system
Project: Validity expert report, deposition, sur-reply expert report, second sur-reply expert report, second deposition, jury trial with settlement mid-trial
- L40. 2018 **Fish & Richardson, P.C.**
Case: In re: Qualcomm Antitrust Litigation (Client: Apple)
Southern district of California, Case No. 3:17-cv-00108-GPC-MDD
Matter: Qualcomm antitrust litigation
Project: Two expert rebuttal reports, deposition
- L41. 2018 **Susman Godfrey LLP**
Case: In re: Qualcomm Antitrust Litigation (Client: Class Action)
Northern district of California, Case No. 5:17-md-02773-LHK
Matter: Qualcomm antitrust litigation
Project: Expert declaration on standard essential patents, expert report on deemed essential patents, rebuttal expert report, deposition
- L42. 2018 **284 Partners**
Case: Federal Trade Commission v. Qualcomm Incorporated
Northern district of California, Case No. 5:17-cv-00220
Matter: Qualcomm antitrust litigation
Project: Expert report on standard essential patents, expert rebuttal report, deposition
- L43. 2018 **Vorys, Sater, Seymour and Pease LLP**
Case: Route1 Inc. v. Airwatch LLC
District of Delaware, Case No. 17-331-RGA
Matter: Patent infringement, remote access

- Project: Source code review, declaration regarding claim construction, infringement expert report, validity expert report, reply expert report, deposition, three declarations regarding re-exam
- L44. 2018 **Sidley Austin LLP**
Case: Samsung Electronics Co., Ltd v. Huawei Technologies Co., Ltd.
IPR2017-01471, IPR2017-01474, IPR2017-01475
Matter: *Inter Partes* Review, 4G/LTE
Project: Three declarations to support three Patent Owner responses, supplemental declaration, deposition
- L45. 2018 **Fitzpatrick Cella Harper & Scinto**
Case: IPC Systems, Inc. v. Cloud9 Technologies, LLC
District of Delaware, Case No. 16-cv-443-GMS
Matter: Patent infringement, telephone stations and trading turrets
Project: Source code review, declaration regarding claim construction, supplemental declaration regarding claim construction
- L46. 2018 **Haynes and Boone, LLP**
Case: LG Electronics Inc., et al. v. Wi-LAN Inc., et al.
IPR2018-00673, IPR2018-00674, IPR2018-00704, IPR2018-00705, IPR2018-00709, IPR2018-00710
Matter: *Inter Partes* Review, bandwidth allocation
Project: Six declarations to support six IPR petitions, two depositions, two reply declarations
- L47. 2018 **Pillsbury Winthrop Shaw Pittman LLP**
Case: Cellular Communications Equipment v. ZTE, HTC Corporation, et al.
Eastern district of Texas, Case No. 6:16-cv-475-RWS
Matter: Patent infringement, LTE, power control, emergency notification
Project: Invalidity expert report, deposition
- L48. 2018 **Finnegan Henderson Farabow Garrett & Dunner LLP**
Case: FanDuel, Inc. DraftKings, Inc., and Bwin.Party Digital Entertainment PLC. v. CG Technology Development, LLC
IPR2017-00902, IPR2017-01333, IPR2017-01491, IPR2017-01532
Matter: *Inter Partes* Review, location-based gaming
Project: Four declarations to support four Patent Owner responses, two supplemental declarations, four depositions
- L49. 2018 **Calfee, Halter & Griswold LLP**
Case: Hytera Communications Corp. Ltd. v. Motorola Solutions, Inc.
Northern district of Ohio, Case No. 1:17-cv-01794-DNC
Matter: Patent infringement, two-way radios

- Project: Source code review, declaration regarding claim construction, rebuttal declaration regarding claim construction, deposition, infringement expert report, validity expert report, two-day deposition
- L50. 2017 **Covington & Burling LLP**
Case: Sharp Corporation, et al. v. Hisense Co., Ltd., et al.
In the Matter of Certain Wi-Fi Enabled Electronic Devices and Components Thereof, ITC Investigation No. 337-TA-1072
Matter: Patent infringement, Wi-Fi, OFDMA
Project: Declaration regarding claim construction
- L51. 2017 **Vorys, Sater, Seymour and Pease LLP**
Case: Airwatch LLC and VMWare Inc. v. Route1 Inc.
IPR2017-02145
Matter: *Inter Partes* Review, remote access
Project: Declaration to support Patent Owner response
- L52. 2017 **Simpson Thacher & Bartlett LLP**
Case: XR Communications, LLC. v. Ubiquiti Networks, Inc.
Central district of California, Case No. 2:17-cv-02968-AG(JCGx)
Matter: Patent infringement, Wi-Fi and adaptive antennas
Project: Declaration regarding claim construction, deposition
- L53. 2017 **Covington & Burling LLP**
Case: Huawei Device USA Inc. v. Hitachi Maxell, Ltd.
IPR2018-00209, IPR2018-00210
Matter: *Inter Partes* Review, base station selection, GPS/Cellular location
Project: Two declarations to support two IPR petitions
- L54. 2017 **Calfee, Halter & Griswold LLP**
Case: Hytera Communications Corp. Ltd. v. Motorola Solutions, Inc.
IPR2018-00128, IPR2017-02183
Matter: *Inter Partes* Review, two-way radios
Project: Declaration to support IPR petition, deposition, two supplemental declarations, two depositions
- L55. 2017 **Finnegan Henderson Farabow Garrett & Dunner LLP**
Case: Hytera Communications Corp. Ltd. v. Motorola Solutions, Inc.
IPR2017-02179, IPR2017-02183
Matter: *Inter Partes* Review, two-way radios
Project: Two declarations to support two IPR petitions, deposition

- L56. 2017 **Mayer Brown LLP**
Case: Silver Spring Networks, Inc. v. Sunrise Technologies, Inc.
Silver Spring Networks, Inc. v. Weatherproof Wireless, LLC
IPR2017-To Be Assigned, IPR2017-To Be Assigned
Matter: *Inter Partes* Review, power meter
Project: Two declarations to support two IPR petitions
- L57. 2017 **Covington & Burling LLP**
Case: Hitachi Maxell, Ltd. v. Huawei Device USA Inc. et al.
Eastern district of Texas, Texarkana division, Case No. 5:16-cv-00178-RWS
Matter: Patent infringement, 3G/4G
Project: Source code review, declaration regarding claim construction, invalidity expert report, non-infringement expert report, non-infringing alternatives expert report, two depositions
- L58. 2017 **Finnegan Henderson Farabow Garrett & Dunner LLP**
Case: LG Electronics, Inc. et al. v. BLU Products, Inc. and CT Miami, LLC
In the Matter of Certain LTE Wireless Communication Devices and Components Thereof, ITC Investigation No. 337-TA-1051
Matter: Patent infringement, 4G/LTE
Project: Declaration regarding claim construction, second declaration regarding claim construction
- L59. 2017 **Sidley Austin LLP**
Case: Huawei Technologies Co., Ltd. v. Samsung Electronics Co., Ltd.
IPR2017-01979, IPR2017-01980, IPR2017-01986
Matter: *Inter Partes* Review, 4G/LTE
Project: Three declarations to support three IPR petitions, deposition
- L60. 2017 **Finnegan Henderson Farabow Garrett & Dunner LLP**
Case: Motorola Solutions, Inc. v. Hytera Communications Corp. Ltd. et al.
In the Matter of Certain Two-way Radio Equipment Systems, Related Software and Components Thereof, ITC Investigation No. 337-TA-1053
Matter: Patent infringement, two-way radio
Project: Source code review, declaration regarding claim construction, invalidity expert report, non-infringement expert report, deposition, ITC hearing testimony
- L61. 2017 **Haynes and Boone, LLP**
Case: Rackspace US, Inc. v. Realtime Data LLC
IPR2017-01691
Matter: *Inter Partes* Review, data compression
Project: Declaration to support IPR petition

- L62. 2017 **Pillsbury Winthrop Shaw Pittman LLP**
Case: ZTE (USA), HTC Corporation, et al. v. Cellular Communications Equipment
IPR2017-01508, IPR2017-01509
Matter: *Inter Partes* Review, LTE, power control, emergency notification
Project: Two declarations to support two IPR petitions, two depositions
- L63. 2017 **Alston & Bird LLP; Womble Carlyle Sandridge & Rice LLP**
Case: Iron, Inc. and Duke Energy Corp. v. Smart Meter Technologies
IPR2017-01199
Matter: *Inter Partes* Review, power meter
Project: Declaration to support IPR petition, deposition
- L64. 2017 **Haynes and Boone, LLP**
Case: Ericsson Inc. v. Regents of the University of Minnesota
IPR2017-01186, IPR2017-01200, IPR2017-01213
Matter: *Inter Partes* Review, OFDM and MIMO
Project: Three declarations to support three IPR petitions
- L65. 2017 **Quinn Emanuel Urquhart & Sullivan, LLP**
Case: GENBAND US, LLC v. Metaswitch Networks Ltd, et al.
Eastern district of Texas, Marshall division, Case No. 2:16-cv-582-JRG-RSP
Matter: Patent infringement, Internet protocols and VoIP
Project: Expert report regarding essentiality
- L66. 2017 **Mayer Brown LLP**
Case: Uniloc USA, Inc. et al. v. Avaya Inc., ShoreTel, Inc., et al.
Eastern district of Texas, Tyler division, Case Nos. 6:15-cv-1168-JRG
Matter: Patent infringement, instant messaging and conference calling
Project: Source code review, non-infringement consulting
- L67. 2017 **Fish & Richardson P.C.**
Case: Nokia Solutions and Networks US LLC, et al. v. Huawei Technologies Co. Ltd., et al.
Eastern district of Texas, Marshall division, Case Nos. 2:16-cv-753-JRG-RSP, 2:16-cv-754
Matter: Patent infringement, 4G/LTE
Project: Claim construction, two declarations
- L68. 2017 **Rothwell Figg Ernst & Manbeck, PC; Pepper Hamilton LLP**
Case: Samsung Electronics, et al. v. Rembrandt Wireless Technologies, LP
IPR2015-00555
Matter: *Ex Parte* Reexamination, Bluetooth
Project: Two declarations to support two Patent Owner responses, supplemental declaration to support Patent Owner reply

- L69. 2016 **Sidley Austin LLP**
Case: Huawei Technologies Co., et al. v. Samsung Electronics Co, et al. and Samsung Research America v. Hisilicon Technologies Co, LTD
Northern district of California, San Francisco division, Case No. 3:16-cv-2787-WHO
Matter: Patent infringement, 3G/4G/LTE
Project: Source code review, declaration regarding claim construction, declaration opposing summary judgement, infringement expert report, invalidity expert report, non-infringement expert report, validity expert report, two depositions
- L70. 2016 **Bragalone Conroy PC**
Case: Securus Technologies, Inc. v. Global Tel*Link Corporation
CBM2017-00034
Matter: Covered Business Method Review, call monitoring and recording
Project: Declaration to support CBM petition, deposition
- L71. 2016 **Braxton, Hilton & Perrone PLLC**
Case: Biosonix, LLC. v. Hydrowave, LLC et al.
Eastern district of Texas, Case No. 2:16-cv-139-RC
Matter: Patent infringement, underwater transceivers
Project: Claim construction, Markman hearing testimony
- L72. 2016 **Gray Reed & McGraw**
Case: Optis Cellular Technology, LLC and PanOptis Patent Management, LLC. v. Blackberry Corporation, et al.
Eastern district of Texas, Marshall division, Case No. 2:16-cv-59-JRG-RSP, Case No. 2:16-cv-61-JRG-RSP, Case No. 2:16-cv-62-JRG-RSP
Matter: Patent infringement, LTE
Project: Claim construction, three declarations regarding claim construction, deposition
- L73. 2016 **Ropes & Gray LLP; Davidson Berquist Jackson & Gowdey**
Case: SIPCO, LLC et al v. Emerson Electric Co. et al
Eastern district of Texas, Tyler division, Case No. 6:15-cv-907
Emerson Electric Co. et al v. SIPCO, LLC et al.
Northern district of Georgia, Atlanta division, Case No. 1:15-cv-00319-AT
Matter: Patent infringement, links in wireless networks and remote monitoring
Project: Source code review, invalidity consulting

- L74. 2016 **EIP US LLP**
Case: GENBAND US, LLC et al. v. Metaswitch Networks Ltd
IPR2015-01456, IPR2015-01457
Matter: *Inter Partes* Review, media gateways
Project: Two declarations to support Patent Owner responses, two depositions
- L75. 2016 **Haynes and Boone, LLP**
Case: Cox Communications, Inc. v. AT&T Intellectual Property I, II, LP
IPR2015-01187, IPR2015-01227, IPR2015-01273, IPR2015-01536
Matter: *Inter Partes* Review, cable networks
Project: Four declarations to support Patent Owner responses, four depositions
- L76. 2016 **Mayer Brown LLP**
Case: Odyssey Wireless v. Motorola Mobility LLC
Eastern district of North Carolina, Western division, Case No. 5:14-cv-491-D
Southern district of California, Case No. 3:15-cv-01741-H-RBB
Matter: Patent infringement, LTE
Project: Source code review, non-infringement consulting
- L77. 2016 **Cooley LLP; Finnegan LLP**
Case: Saint Lawrence Comm. LLC v. Motorola Mobility LLC, ZTE (USA) Inc., et al.
Eastern district of Texas, Marshall division, Case No. 2:15-cv-000351-JRG, Case No. 2:15-cv-000349-JRG
Matter: Patent infringement, speech compression, coding and decoding
Project: Invalidity expert report, expert report regarding AMR-WB standard, expert report regarding Opus and Silk, supplemental expert report regarding invalidity, two-day depositions, jury trial testimony for Motorola
- L78. 2015 **Sidley Austin LLP**
Case: Evolved Wireless, LLC v. Microsoft Corp, et al.
District of Delaware, Case No. 15-cv-546
Matter: Patent infringement, LTE
Project: Prior art and invalidity consulting
- L79. 2015 **McKool Smith**
Case: Optis Wireless Technology, LLC and PanOptis Patent Management, LLC. v. ZTE Corporation and ZTE (USA) Inc.
Eastern district of Texas, Marshall division, Case No. 2:15-cv-300-JRG-RSP
Matter: Patent infringement, cellular messages and multimedia attachments
Project: Source code review, claim construction, declaration

- L80. 2015 **Fish & Richardson, P.C.**
Case: Saint Lawrence Comm. LLC v. LG Elec., Inc. et al.
Eastern district of Texas, Marshall division, Case No. 2:14-cv-1055-JRG
Matter: Patent infringement, speech compression, coding and decoding
Project: Invalidity expert report
- L81. 2015 **Finnegan Henderson Farabow Garrett & Dunner LLP**
Case: LG Electronics, Inc. v. Cellular Communications Equipment LLC
IPR2016-00178
Matter: *Inter Partes* Review, LTE
Project: Declaration to support IPR petition
- L82. 2015 **McKool Smith**
Case: AT&T, et al. v. Cox Communication, Inc., et al.
District of Delaware, Case No. 14-1106-GMS
Matter: Patent infringement, cable networks
Project: Claim construction, declaration
- L83. 2015 **McKool Smith**
Case: Ericsson Inc., et al. v. TCL Communication, et al.
Eastern district of Texas, Marshall division, Case No. 2:15-cv-00011-RSP
Matter: Patent infringement, wireless devices and systems
Project: Source code review, claim construction, declaration, infringement expert report, validity expert report, two-day depositions
- L84. 2015 **Foley & Lardner LLP**
Case: Kyocera Communications, Inc. v. Cellular Communications Equipment LLC
IPR2015-01559, IPR2015-01564
Matter: *Inter Partes* Review, LTE, power control, emergency notification
Project: Two declarations to support two IPR petitions
- L85. 2015 **Fish & Richardson, P.C.**
Case: Fairfield Industries Inc. v. Wireless Seismic, Inc.
Southern district of Texas, Case No. 4:14-cv-02972-KPE
Matter: Patent infringement, wireless sensor networks
Project: Non-infringement expert report
- L86. 2015 **Quinn Emanuel Urquhart & Sullivan, LLP**
Case: GENBAND US, LLC v. Metaswitch Networks Ltd, et al.
Eastern district of Texas, Marshall division, Case No. 2:14-cv-33-JRG-RSP
Matter: Patent infringement, Internet protocols and VoIP

- Project: Expert report regarding essentiality, non-infringement expert report, rebuttal expert report regarding non-practice, supplemental rebuttal expert report, three-day depositions, jury trial testimony
- L87. 2015 **Duane Morris LLP; Foley & Lardner LLP**
Case: Mobile Telecommunications Technologies, LLC v. Leap Wireless International, Cricket Communications, Inc.
Eastern district of Texas, Marshall division, Case No. 2:13-cv-00885-RSP
Matter: Patent infringement, OFDM and MIMO
Project: Non-infringement expert report, deposition
- L88. 2015 **Hogan Lovells US LLP; Kenyon & Kenyon LLP**
Case: One-E-Way v. Beats Electronics, LLC, Sony Corporation, et al.
In the Matter of Certain Wireless Headsets, ITC Investigation No. 337-TA-943
Matter: Patent infringement, wireless communication
Project: Claim construction, declaration
- L89. 2015 **McKool Smith**
Case: Solocron Media, LLC v. AT&T Inc., et al.
Eastern district of Texas, Marshall division, Case No. 2:13-cv-1059-JRG
Matter: Patent infringement, ringtone download
Project: Claim construction, invalidity expert report
- L90. 2015 **EIP US LLP**
Case: Good Technology Software, Inc. v. Mobile Iron, Inc.
IPR2015-00833, IPR2015-00836, IPR2015-01090
Matter: *Inter Partes* Review, software management in wireless devices
Project: Three declarations to support three IPR petitions
- L91. 2015 **McKool Smith**
Case: AirWatch LLC v. Good Technology Corp
Northern district of Georgia, Case No. 1:14-cv-02281-SCJ
Matter: Patent infringement, software management in wireless devices
Project: Claim construction, declaration
- L92. 2015 **Simpson Thacher & Bartlett LLP**
Case: IXI Mobile (R&D) Ltd. et al. v. Apple Inc.
Southern district of New York, Case No. 14-cv-7594-RJS
Matter: Patent infringement, PDA and Bluetooth
Project: Invalidity consulting

- L93. 2014 **Bragalone Conroy PC**
Case: Global Tel*Link Corporation v. Securus Technologies, Inc.
IPR2014-00785, IPR2014-00810, IPR2014-00824, IPR2014-00825,
IPR2014-01278, IPR2014-01282, IPR2014-01283
Matter: *Inter Partes* Review, VoIP call monitoring and recording, allocating
telecommunication resources and information systems
Project: Seven declarations to support seven Patent Owner responses, five
depositions
- L94. 2014 **Orrick, Herrington & Sutcliffe LLP**
Case: Shopkick, Inc. v. Novitaz, Inc.
IPR2015-00277, IPR2015-00278
Matter: *Inter Partes* Review, wireless customer service management
Project: Two declarations to support two IPR petitions
- L95. 2014 **Paul Hastings LLP**
Case: Cellular Communications Equipment LLC v. AT&T, et al.
Eastern district of Texas, Tyler division, Case No. 6:13-cv-507-LED
(Lead Case for Consolidation)
Matter: Patent infringement, 3G cellular communication
Project: Claim construction, declaration
- L96. 2014 **Baker Botts LLP**
Case: Orlando Communications LLC v. AT&T, et al.
M.D. Florida, Case No. 6:14-cv-01021
Matter: Patent infringement, 3G/4G cellular communication
Project: Non-infringement and claim construction consulting
- L97. 2014 **EIP US LLP**
Case: Good Technology Software, Inc. v. AirWatch, LLC
IPR2015-00248, IPR2015-00875
Matter: *Inter Partes* Review, software management in wireless devices
Project: Two declarations to support two IPR petitions
- L98. 2014 **Bragalone Conroy PC**
Case: Securus Technologies, Inc. v. Global Tel*Link Corporation
IPR2015-00153, IPR2015-00155, IPR2015-00156
Matter: *Inter Partes* Review, VoIP call monitoring and recording
Project: Three declarations to support three IPR petitions, two depositions
- L99. 2014 **Andrews Kurth LLP**
Case: Sony Mobile Communications (USA) v. Adaptix Inc.
IPR2014-01524, IPR2014-01525
Matter: *Inter Partes* Review, subcarrier selection in LTE
Project: Two declarations to support two IPR petitions, deposition

- L100. 2014 **Steptoe & Johnson LLP; Baker & McKenzie LLP**
Case: VTech Communications, Inc. and Uniden America Corporations v. Spherix Incorporated
IPR2014-01432
Matter: *Inter Partes* Review, IP telephony
Project: Declaration to support IPR petition, deposition, reply declaration, deposition
- L101. 2014 **Steptoe & Johnson LLP; Baker & McKenzie LLP**
Case: Spherix Inc. v. VTech Telecommunications Ltd., et al.
Spherix Inc. v. Uniden Corp, et al.
Northern district of Texas, Dallas Division, Case No. 3:13-cv-3494 and 3:13-cv-3496
Matter: Patent infringement, IP telephony
Project: Claim construction, declaration, deposition
- L102. 2014 **McKool Smith**
Case: Good Technology Corp. v. MobileIron, Inc.
Northern district of California, Case No. 5:12-cv-05826-PSG
Matter: Patent infringement, software management in wireless devices
Project: Claim construction, three declarations, claim invalidity expert report, non-infringement expert report, deposition, jury trial testimony
- L103. 2014 **Lee & Hayes**
Case: Broadcom Corp. v. Ericsson, Inc.
IPR2013-00601, IPR2013-00602, and IPR2013-00636
Matter: *Inter Partes* Review, ARQ protocols
Project: Three declarations to support Patent Owner responses, two declarations to support Patent Owner Motion to Amend, deposition, two reply declarations
- L104. 2014 **Sidley Austin LLP**
Case: Adaptix, Inc. v. Huawei Technologies Co., et al.
Eastern district of Texas, Case No. 6:13-cv-00438, 439, 440 and 441
Matter: Patent infringement, subcarrier selection in LTE
Project: Source code review, non-infringement consulting
- L105. 2014 **Finnegan Henderson Farabow Garrett & Dunner LLP**
Case: Cell and Network Selection LLC v. Huawei Technologies Co., et al.
Eastern district of Texas, Case No. 6:13-cv-00404-LED-JDL
Matter: Patent infringement, base station selection in LTE
Project: Non-infringement consulting
- L106. 2014 **Feinberg Day Alberti & Thompson LLP**
Case: DSS Technology Management, Inc. v. Apple Inc.
Eastern district of Texas, Tyler division, Case No. 6:13-cv-00919-JDL

- Matter: Patent infringement, PDA and Bluetooth
Project: Claim construction and invalidity consulting
- L107. 2014 **Sheppard Mullin Richter & Hampton LLP**
Case: Digcom Inc. v. ZTE (USA), Inc.
District of Nevada, Case No. 3:13-cv-00178-RCJ-WGC
Matter: Patent infringement, cellular communication
Project: Claim construction consulting
- L108. 2014 **Lott & Fischer**
Case: Zenith Electronics, LLC, et al. v. Craig Electronics, Inc.
Southern district of Florida, Case No. 9:13-cv-80567-DMM/DLB
Matter: Patent infringement, HDTV transmission and reception
Project: Opening expert report regarding nonessentiality
- L109. 2013 **McKool Smith**
Case: Zenith Electronics, LLC, et al. v. Curtis International Ltd.
Southern district of Florida, Case No. 9:13-cv-80568-DMM/DLB
Matter: Patent infringement, HDTV transmission and reception
Project: Claim construction, declaration, deposition
- L110. 2013 **Gibson Dunn**
Case: Straight Path IP Group v. Sharp Corp. and Sharp Electronics Corp.
In the Matter of Certain Point-to-Point Network Communication
Devices and Products Containing Same, ITC Investigation No. 337-
TA-892
Matter: Patent infringement, point-to-point network communication
Project: Non-infringement consulting
- L111. 2013 **Kilpatrick Townsend & Stockton LLP; Cooley LLP**
Case: Monec Holding AG v. Motorola Mobility LLC, HTC, et al.
District of Delaware, Case No. 1:11-cv-798-LPS-SRF
Matter: Patent infringement, displaying books on tablets
Project: Non-infringement expert report for Motorola, non-infringement expert
report for HTC, deposition
- L112. 2013 **Gartman Law Group**
Case: Lone Star WiFi LLC v. Legacy Stonebriar Hotel, Ltd; et al.
Eastern district Of Texas, Tyler, Case No. 6:12-cv-957
Matter: Patent infringement, levels of access in Wi-Fi networks
Project: Claim validity consulting
- L113. 2013 **White & Case, LLP**
Case: Nokia Corp and Nokia, Inc. v. HTC Corp and HTC America, Inc.

- In the Matter of Certain Portable Electronic Communication Devices,
Including Mobile Phones and Components Thereof, ITC Investigation
No. 337-TA-885
- Matter: Patent infringement, App download and installation
Project: Non-infringement consulting
- L114. 2013 **Heim, Payne & Chorush, LLP**
Case: Rembrandt Wireless v. Samsung Electronics Co., et al.
Eastern district of Texas, Marshall, Case No. 2:13-cv-213-JRG-RSP
Matter: Patent infringement, Bluetooth
Project: Expert report regarding validity, deposition, jury trial
- L115. 2013 **Baker Hostetler; Davis Polk & Wardwell LLP**
Case: Comcast v. Sprint; and Nextel Inc.
Eastern district of Pennsylvania, Case No. 2:12-cv-00859-JD
Matter: Patent infringement, SMS/MMS in Cellular Networks
Project: Infringement expert report, validity expert report, reply expert report,
declaration, two-day depositions, jury trial testimony
- L116. 2013 **McKool Smith**
Case: Samsung Electronics America v. Ericsson Inc.
In the Matter of Certain Wireless Communications Equipment and
Articles Therein, ITC Investigation No. 337-TA-866
Matter: Patent infringement, LTE uplink and downlink
Project: Source code review, prior art research, claim construction, claim
invalidity expert report, non-infringement expert report, ITC hearing
testimony
- L117. 2012 **DLA Piper US LLP**
Case: CSR Technology Inc. v. Freescale Semiconductor, Inc.
USDC-San Francisco, Case No. 3:12-cv-02619-RS
Matter: Patent infringement, radio transceivers
Project: Claim construction, declaration
- L118. 2012 **Fish & Richardson P.C.**
Case: GPNE Corp. v. Apple, Inc.; et al.
USDC-ND California, Case No. 5:12-cv-02885-LHK
Matter: Patent infringement, resource allocation in wireless networks
Project: Prior art research consulting
- L119. 2012 **Polsinelli Shughart PC**
Case: Single Touch Interactive, Inc. v. Zoove Corporation
Northern district of California, Case No. 3:12-cv-00831-JSC
Matter: Patent infringement, abbreviated dialing, information delivery
Project: Claim construction, Markman hearing testimony, two declarations

- L120. 2012 **K & L Gates**
Case: EON Corp. IP Holdings, LLC v. Novatel Wireless, Inc.; et al.
DC-Tyler, Texas, Case No. 6:11-cv-00015-LED-JDL
Matter: Patent infringement, wireless modem and 3G services
Project: Non-infringement expert report, deposition
- L121. 2012 **Simpson Thacher & Bartlett LLP**
Case: CSR Technology, Inc. v. Bandspeed, Inc.
Western district of Texas, Case No. 1:12-cv-297-LY
Matter: Patent infringement, packet identification in 2.4 GHz and 5 GHz
Project: Source code review, Markman hearing testimony, infringement expert report
- L122. 2012 **Sheppard Mullin Richter & Hampton LLP**
Case: Wi-LAN v. HTC America, Inc., et al.
Eastern district of Texas, Case No. 6:10-cv-521-LED
Matter: Patent infringement, CDMA, Orthogonal Codes
Project: Source code review, non-infringement expert report, deposition, jury trial testimony
- L123. 2012 **Dechert LLP**
Case: Hitachi v. TPV and Vizio, Inc.; and Vizio v. Hitachi, LTD.
Eastern district of Texas, Case No. 2:10-cv-260
Matter: Patent infringement, HD television transmission and reception
Project: Prior art research, claim invalidity consulting
- L124. 2012 **Fish & Richardson P.C.; Covington & Burling; Alston & Bird; Brinks Hofer Gilson & Lione**
Case: InterDigital Commc'n, LLC v. Huawei Tech. Co. LTD; LG Electronics, Inc.; Nokia, Inc.; and ZTE (USA) Inc.
Certain Wireless Devices With 3G Capabilities and Components Thereof, ITC Investigation No. 337-TA-800
Matter: Patent infringement, channel coding in UMTS, HSDPA
Project: Non-infringement consulting
- L125. 2012 **Fish & Richardson P.C.; Covington & Burling; Alston & Bird; Brinks Hofer Gilson & Lione**
Case: InterDigital Commc'n, LLC v. Huawei Tech. Co. LTD; LG Electronics, Inc.; Nokia, Inc.; and ZTE (USA) Inc.
District of Delaware, Case No. 1:11-cv-00654-UNA
Matter: Patent infringement, channel coding in UMTS, HSDPA
Project: Non-infringement consulting

- L126. 2011 **O'Melveny & Myers LLP**
Case: MobileMedia Ideas, LLC v. Apple, Inc.
District of Delaware, Case No. 1:10-cv-00258-SLR-MPT
Matter: Patent infringement, voice control, call rejection in mobile phones
Project: Source code review, prior art research, declaration, claim invalidity expert report, non-infringement expert report, deposition, jury trial testimony
- L127. 2011 **Wilmer Cutler Pickering Hale and Dorr**
Case: Apple, Inc. v. Samsung Electronics Co.
Northern district of California, Case No. 5:11-cv-01846-LHK
Matter: Patent infringement, channel coding in CDMA, E-AGCH, TFCI
Project: Prior art research, claim construction consulting
- L128. 2011 **Weil, Gotshal & Manges LLP**
Case: Vizio, Inc. v. Renesas Electronics America, Inc.
ITC Investigation No. 337-TA-789
Matter: Patent infringement, HD television transmission and reception
Project: Claim invalidity consulting
- L129. 2011 **Shapiro Cohen**
Case: TenXc Wireless Inc. v. Andrew LLC
TenXc Wireless Inc. v. Mobi Antenna Technologies Ltd.
Matter: Patent infringement, antenna design, sectorized cellular network
Project: Claim validity consulting
- L130. 2010 **Fish & Richardson P.C.**
Case: Vizio, Inc., v. LG Electronics, Inc.
ITC Investigation No. 337-TA-733
Matter: Patent infringement, HD television transmission and reception
Project: Claim charts, claim construction expert report, deposition
- L131. 2010 **Fish & Richardson P.C.**
Case: Vizio, Inc., v. LG Electronics, Inc.
District of Maryland, Case No. 1:09-cv-1481-BEL
Matter: Patent infringement, HD television transmission and reception
Project: Claim charts, claim construction expert report, deposition
- L132. 2008 **Kaye Scholer LLP**
Case: eBay Inc. v. IDT.
Western district of Arkansas, Case No. 4:08-cv-4015-HFB
Matter: Patent infringement, long distance communication using Internet
Project: Prior art research, claim construction consulting

- L133. 2008 **Simpson Thacher & Bartlett LLP**
Case: Commil USA, LLC v. Cisco Systems, Inc.
 Eastern district of Texas, Case No. 2:07-cv-00341-DF-CE
Matter: Patent infringement, two-level wireless protocol
Project: Prior art research
- L134. 2006 **Woodfill and Pressler**
Case: Charles Russell v. Interinsurance Exchange of the Auto Club
 Harris County, Texas, Case No. 2005-19706
Matter: House fire and insurance claim
Project: Determining user location using cellular phone records, expert report,
 deposition, jury trial testimony

Consulting History

- From: 1/2013 **Heim, Payne & Chorush, LLP**
To: 3/2013 Houston, TX
Duties: Analyze patents on wireless technologies.
- From: 4/2007 **Collin County Sheriff's Office**
To: 5/2007 McKinney, TX
Duties: Analyzed cellular record data and determined user location in a double-homicide investigation.
- From: 4/2004 **Allegiant Integrated Solutions**
To: 5/2004 Fort Worth, TX
Duties: Designed and developed an integrated set of tools for fast deployment of wireless networks. The tools optimize the placement of Access Points and determine their respective channel allocations to minimize interference and maximize capacity.
- From: 3/2002 **Input/Output Incorporated**
To: 4/2002 New Orleans, LA
Duties: Designed and implemented an algorithm in MATLAB for optimizing the frequency selection process used by sonar for scanning the bottom of the ocean.
- From: 6/1998 **Teleware Corporation**
To: 7/1998 Seoul, South Korea
Duties: Designed and developed a software package for analyzing the capacity in a CDMA network to maximize the number of subscribers.

Employment History

- From: 1/2015 **University of North Texas**
To: Present Denton, TX

Position: *Associate Chair of Graduate Studies Department of Computer Science and Engineering*
In charge of all administrative duties related to the Master's and Ph.D. programs in the department.

From: 5/2008 **University of North Texas**
To: Present Denton, TX
Position: *Tenured Associate Professor Department of Computer Science and Engineering*
Conducting research on cellular networks and wireless sensor networks. Teaching wireless communication courses. Advising graduate and undergraduate students.

From: 9/2002 **University of North Texas**
To: 5/2008 Denton, TX
Position: *Assistant Professor Department of Computer Science and Engineering*
Conducting research on WCDMA/UMTS wireless networks. Teaching wireless communication and computer architecture courses. Advising graduate and undergraduate students.

From: 1/2002 **University of New Orleans**
To: 8/2002 New Orleans, LA
Position: *Assistant Professor Department of Electrical Engineering*
Designed and taught two new courses "Computer Systems Design I and II". Developed a Computer Engineering Curriculum with strong hardware-design emphasis. Formed a wireless research group. Advised graduate and undergraduate students.

From: 10/2000 **Comspace Corporation**
To: 12/2001 Coppell, TX
Position: *Senior Systems Engineer*
Designed, coded (in Matlab), and simulated Viterbi decoding, Turbo coding, trellis coded modulation (TCM), and Reed-Muller codes. Optimized soft decision parameters and interleavers for additive white Gaussian and Rayleigh faded channels. Extended the control and trunking of push-to-talk Logic Trunked Radio (LTR) to include one-to-one and one-to-many voice and data messaging.

From: 8/1996 **MinMax Corporation**
To: 8/2000 Saint Louis, MO
Position: *Research Associate*
Designed software packages that provide the tools to flexibly allocate capacity in a CDMA network and maximize the number of subscribers. Analyzed and simulated different audio compression schemes. Validated, simulated (logical and timing), and developed the hardware architecture for an ATM switch capable of channel group switching.

From: 8/1994 **Washington University**
To: 8/2000 Saint Louis, MO
Position: *Research and Teaching Assistant*
Taught, consulted, and graded Circuit Analysis at the undergraduate level and Network Design at the graduate level.

Publications

Conference Proceedings

- C1. U.K. Dey, **R. Akl**, R. Chataut, M. Robaei, "Modified PHY Layer for High Performance V2X Communication using 5G NR," *IEEE UEMCON 11th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference*, October 2020, paper no. 1570681534, 6 pgs.
- C2. R. Chataut, **R. Akl**, "An Efficient and Fair Scheduling for Downlink 5G Massive MIMO Systems," *IEEE Texas Symposium on Wireless and Microwave Circuits and Systems*, June 2020, paper no. TSWMCS2020-39, 8 pgs.
- C3. R. Chataut, **R. Akl**, "Efficient and Low-Complexity Iterative Detectors for 5G Massive MIMO Systems," *IEEE DCOSS 2020 International Conference on Distributed Computing in Sensor Systems*, May 2020, pp. 442-449, 8 pgs.
- C4. U.K. Dey, **R. Akl**, R. Chataut, "High Throughput Vehicular Communication Using Spatial Multiplexing MIMO," *IEEE CCWC 2020 The 10th Annual Computing and Communication Workshop and Conference*, January 2020, paper no. 1570613408, 6 pgs.
- C5. R. Chataut, **R. Akl**, M. Robaei, "Accelerated and Preconditioned Refinement of Gauss-Seidel Method for Uplink Signal Detection in 5G Massive MIMO Systems," *IEEE CCWC 2020 The 10th Annual Computing and Communication Workshop and Conference*, January 2020, paper no. 1570605343, 7 pgs.
- C6. M. Robaei, **R. Akl**, "Examining Spatial Consistency for Millimeter-Wave Massive MIMO Channel Estimation in 5G-NR," *IEEE ICCE 2020 The 38th International Conference on Consumer Electronics*, January 2020, paper no. 1570596880, 6 pages.
- C7. R. Chataut, **R. Akl**, "Channel Gain Based User Scheduling for 5G Massive MIMO Systems," *IEEE HONET-ICT 2019 The 16th International Conference on Smart Cities: Improving Quality of Life Using ICT & IoT and AI*, October 2019, paper no. 1570565594, 5 pgs.
- C8. M. Robaei, **R. Akl**, "Time-Variant Broadband mmWave Channel Estimation Based on Compressed Sensing," *IEEE UEMCON 2019 The 10th Annual*

Ubiquitous Computing, Electronics & Mobile Communication Conference, October 2019, paper no. 1570577430, 7 pages.

- C9. R. Chataut, **R. Akl**, U. Dey, “Least Square Regressor Selection Based Detection for Uplink 5G Massive MIMO Systems,” *IEEE WAMICON 2019 The 20th Annual IEEE Wireless and Microwave Technology Conference*, April 2019, paper no. 1570524727, 6 pgs.
- C10. R. Chataut, **R. Akl**, “Huber Fitting Based ADMM Detection for Uplink 5G Massive MIMO Systems,” *IEEE UEMCON 2018 The 9th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference*, November 2018, paper no. 1570492416, 5 pgs.
- C11. R. Chataut, **R. Akl**, “Efficient and Low Complex Uplink Detection for 5G Massive MIMO Systems,” *IEEE WAMICON 2018 The 19th Annual Wireless and Microwave Technology Conference*, April 2018, paper no. 1570431593, 6 pgs.
- C12. R. Chataut, **R. Akl**, “Optimal Pilot Reuse Factor Based on User Environments in 5G Massive MIMO,” *IEEE CCWC 2018 The 8th Annual Computing and Communication Workshop Conference*, January 2018, paper no. 1570413394, 6 pgs.
- C13. S. Alotaibi, **R. Akl**, “Radio Resource Management in LTE Femtocell Networks,” *IEEE NCA '17 International Symposium on Network Computing and Applications*, November 2017, paper no. 117, 5 pgs.
- C14. U. Sawant, **R. Akl**, “Subcarrier Allocation in LTE Network Deployment with Mobility,” *IEEE UEMCON 2017 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference*, October 2017, paper no. 1570349184, 8 pgs.
- C15. S. Alotaibi, **R. Akl**, “Packet Scheduling Bandwidth Type-Based Mechanism for LTE,” *IEEE UEMCON 2017 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference*, October 2017, paper no. 1570394639, 6 pgs.
- C16. S. Alotaibi, **R. Akl**, “Dynamic Fractional Frequency Reuse (FFR) Scheme for Two-Tier Network in LTE,” *IEEE UEMCON 2017 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference*, October 2017, paper no. 1570394969, 6 pgs.
- C17. U. Sawant, **R. Akl**, “Evaluation of Adaptive and Non Adaptive LTE Fractional Frequency Reuse Mechanisms,” *IEEE WOCC 2017 The 26th Annual Wireless and Optical Communications Conference*, April 2017, paper no. 1570341174, 6 pgs.
- C18. S. Alotaibi, **R. Akl**, “Range-Based Scheme for Adjusting Transmission Power for Femtocells in Co-Channel Deployment,” *IEEE WTS 2017 The 16th Annual*

- Wireless Telecommunications Symposium*, April 2017, paper no. 1570334744, 5 pgs.
- C19. U. Sawant, **R. Akl**, "A Novel Metric to Study the Performance of Sectorized Fractional Frequency Reuse Techniques in LTE," *IEEE WTS 2017 The 16th Annual Wireless Telecommunications Symposium*, April 2017, paper no. 1570338498, 7 pgs.
- C20. S. Alotaibi, **R. Akl**, "Dynamic Frequency Partitioning Scheme for LTE HetNet Networks Using Fractional Frequency Reuse," *IEEE WCNC '17 Wireless Communications and Networking Conference*, March 2017, paper no. 1570332420, 5 pgs., demo and poster.
- C21. U. Sawant, **R. Akl**, "Performance Evaluation of Network Productivity for LTE Heterogenous Networks with Reward-Penalty Weights Assessment," *IEEE CCWC 2017 The 7th Annual Computing and Communication Workshop Conference*, January 2017, paper no. 1570328396, 6 pgs.
- C22. S. Alotaibi, **R. Akl**, "Self-Adjustment Downlink Transmission Power for Femtocells in Co-Channel Deployment in Heterogeneous Networks," *IEEE CCWC 2017 The 7th Annual Computing and Communication Workshop Conference*, January 2017, paper no. 1570326815, 6 pgs.
- C23. U. Sawant, **R. Akl**, "Performance Evaluation of Sectorized Fractional Frequency Reuse Techniques Using Novel Metric," *IEEE ISCC 2016 The Twenty-First IEEE Symposium on Computers and Communications*, June 2016, paper no. 1570275270, 7 pgs.
- C24. R. Tidwell, S. Akumalla, S. Karlaputi, **R. Akl**, K. Kavi, and D. Struble, "Evaluating the Feasibility of EMG and Bend Sensors for Classifying Hand Gestures," *1st International Conference on Multimedia and Human Computer Interaction*, July 2013, paper no. 63, 8 pgs.
- C25. **R. Akl**, K. Pasupathy, and M. Haidar, "Anchor Nodes Placement for Effective Passive Localization," *2011 IEEE International Conference on Selected Topics in Mobile and Wireless Networks (iCOST)*, October 2011, paper no. 1569490799, pp. 127 - 132.
- C26. **R. Akl**, P. Kadiyala, and M. Haidar, "Non-Uniform Grid-Based Routing in Sensor Networks," *9th IEEE Malaysia International Conference on Communications*, December 2009, paper no. 1569243649, pp. 536 - 540.
- C27. M. Haidar, M. Al-Rizzo, Y. Chan, **R. Akl**, M. Bouharras, "Throughput Validation of an Advanced Channel Assignment Algorithm in IEEE 802.11 WLAN," *ICCSN 2009 – International Conference on Communication Software and Networks*, February 2009, paper no. P385, pp. 801 - 806.

- C28. **R. Akl** and D. Keathly, "Robocamp: Encouraging Young Women to Embrace STEM," 4th Annual TETC Best Practices Conference, February 2009, 13 pgs.
- C29. M. Haidar, R. Ghimire, M. Al-Rizzo, **R. Akl**, Y. Chan, "Channel Assignment in an IEEE 802.11 WLAN Based on Signal-to-interference Ratio," *IEEE CCECE – Canadian Conference on Electrical and Computer Engineering: Communications and Networking*, May 2008, paper no. 1569092894, pp. 1169 - 1174.
- C30. H. Al-Rizzo, M. Haidar, **R. Akl**, and Y. Chan, "Enhanced Channel Assignment and Load Distribution in IEEE 802.11 WLANs," *IEEE International Conference on Signal Processing and Communication*, November 2007, paper no. 1569042132, pp. 768 - 771.
- C31. **R. Akl** and Y. Saravanos, "Hybrid Energy-Aware Synchronization Algorithm in Wireless Sensor Networks," *18th Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, September 2007, paper no 692, 5 pgs.
- C32. M. Haidar, **R. Akl**, and H. Al-Rizzo, "Channel Assignment and Load Distribution in a Power-Managed WLAN," *18th Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, September 2007, paper no. 463, 5 pgs.
- C33. D. Keathly and **R. Akl**, "Attracting and Retaining Women in Computer Science and Engineering: Evaluating the Results," *Proceedings of American Society for Engineering Education: ASEE Annual Conference*, June 2007, paper no. AC 2007-1229, 10 pgs.
- C34. M. Haidar, **R. Akl**, H. Al-Rizzo, Y. Chan, R. Adada, "Optimal Load Distribution in Large Scale WLAN Networks Utilizing a Power Management Algorithm," *Proceedings of IEEE Sarnoff Symposium*, May 2007, 5 pgs.
- C35. R. Dantu, P. Kolan, **R. Akl**, and K. Loper, "Classification of Attributes and Behavior in Risk Management Using Bayesian Networks," *Proceedings of IEEE Intelligence and Security Informatics Conference*, May 2007, pp. 71-74.
- C36. **R. Akl** and A. Arepally, "Dynamic Channel Assignment in IEEE 802.11 Networks," *Proceedings of IEEE Portable 2007: International Conference on Portable Information Devices*, March 2007, pp 309-313.
- C37. **R. Akl** and U. Sawant, "Grid-based Coordinated Routing in Wireless Sensor Networks," *Proceedings of IEEE CCNC 2007: Consumer Communications and Networking Conference*, January 2007, pp. 860-864.

- C38. **R. Akl** and A. Arepally, "Simulation of Throughput in UMTS Networks with Different Spreading Factors," *Proceedings of IEEE VTC Fall 2006: Vehicular Technology Conference*, September 2006, pp. C1-5.
- C39. A. Alhabsi, H. Al-Rizzo, and **R. Akl**, "Parity Assisted Decision Making for QAM Modulation," *International Conference on Mobile Computing and Wireless Communications*, September 2006, paper no. 1568988776, 5 pgs.
- C40. **R. Akl** and R. Garlick, "Retention and Recruitment of Women in Computer Engineering," *ICEE 2006: International Conference on Engineering Education*, July 2006, paper no. 3318, 5 pgs.
- C41. R. Garlick and **R. Akl**, "Intra-Class Competitive Assignments in CS2: A One-Year Study," *ICEE 2006: International Conference on Engineering Education*, July 2006, paper no. 3325, 5 pgs.
- C42. **R. Akl**, D. Tummala, and X. Li, "Indoor Propagation Modeling at 2.4 GHz for IEEE 802.11 Networks," *WNET 2006: Wireless Networks and Emerging Technologies*, July 2006, paper no. 510-014, 6 pgs.
- C43. P. Chen, K. Kavi, and **R. Akl**, "Performance Enhancement by Eliminating Redundant Function Execution," *Proceedings of IEEE: 39th Annual Simulation Symposium*, April 2006, pp. 143-150.
- C44. **R. Akl** and S. Nguyen, "Capacity Allocation in Multi-cell UMTS Networks for Different Spreading Factors with Perfect and Imperfect Power Control," *Proceedings of IEEE CCNC 2006: Consumer Communications and Networking Conference*, January 2006, vol. 2, pp. 928-932.
- C45. W. Li, K. Kavi, and **R. Akl**, "An Efficient Non-Preemptive Real-Time Scheduling," *18th International Conference on Parallel and Distributed Computing Systems*, Las Vegas, NV, September 2005, pp. 154-160.
- C46. S. Nguyen and **R. Akl**, "Approximating User Distributions in WCDMA Networks Using 2-D Gaussian," *CCCC20T 05: International Conference on Computing, Communications, and Control Technologies*, July 2005, 5 pgs.
- C47. **R. Akl** and S. Park, "Optimal Access Point Selection and Traffic Allocation in IEEE 802.11 Networks," *Proceedings of 9th World Multiconference on Systemics, Cybernetics and Informatics (WMSCI 2005): Communication and Network Systems, Technologies and Applications*, July 2005, vol. 8, pp. 75-79.
- C48. **R. Akl**, M. Naraghi-Pour, M. Hegde, "Throughput Optimization in Multi-Cell CDMA Networks," *IEEE WCNC 2005 - Wireless Communications, and Networking Conference*, March 2005, vol. 3, pp. 1292-1297.

- C49. **R. Akl**, "Subscriber Maximization in CDMA Cellular Networks," *Proceedings of CCCT 04: International Conference on Computing, Communications, and Control Technologies*, August 2004, vol. 3, pp. 234-239.
- C50. **R. Akl** and A. Parvez, "Global versus Local Call Admission Control in CDMA Cellular Networks," *Proceedings of CITSA 04: Communications, Information and Control Systems, Technologies and Applications*, July 2004, vol. 2, pp. 283-288.
- C51. **R. Akl** and A. Parvez, "Impact of Interference Model on Capacity in CDMA Cellular Networks," *Proceedings of SCI 04: Communication and Network Systems, Technologies and Applications*, July 2004, vol. 3, pp. 404-408. Selected as **best paper** of those presented in the session: Tele-Communication Systems, Technologies and Application II.
- C52. **R.G. Akl**, M.V. Hegde, M. Naraghi-Pour, P.S. Min, "Call Admission Control Scheme for Arbitrary Traffic Distribution in CDMA Cellular Systems," *IEEE Wireless Communications and Networking Conference*, September 2000, vol. 1, pp. 465-470.
- C53. **R.G. Akl**, M.V. Hegde, M. Naraghi-Pour, P.S. Min, "Cell Placement in a CDMA Network," *IEEE Wireless Communications and Networking Conference*, September 1999, vol. 2, pp. 903-907.
- C54. **R.G. Akl**, M.V. Hegde, P.S. Min, "Effects of Call Arrival Rate and Mobility on Network Throughput in Multi-Cell CDMA," *IEEE International Conference on Communications*, June 1999, vol. 3, pp. 1763-1767.
- C55. **R.G. Akl**, M.V. Hegde, M. Naraghi-Pour, P.S. Min, "Flexible Allocation of Capacity in Multi-Cell CDMA Networks," *IEEE Vehicular Technology Conference*, May 1999, vol. 2, pp. 1643-1647.

Journal Publications

- J1. R. Chataut, **R. Akl**, "Massive MIMO Systems for 5G and Beyond Networks – Overview, Recent Trends, Challenges, and Future Research Direction," *Sensors* 2020, 20(10), 2753, May 2020.
- J2. R. Chataut, **R. Akl**, "Massive MIMO Systems for 5G," *Encyclopedia* 2020, doi:10.32545, 2020, (ISSN 2309-3366).
- J3. S. Alotaibi, **R. Akl**, "Range-Based Scheme for Adjusting Transmission Power of Femtocell in Co-Channel Deployment," *International Journal of Interdisciplinary Telecommunications and Networking*, IJITN Vol. 10, No. 4, pgs. 14-24, 2018.
- J4. U. Sawant, **R. Akl**, "Adaptive and Non Adaptive LTE Fractional Frequency Reuse Mechanisms Mobility Performance," *Advances in Science, Technology and*

- J5. M. Haidar, H.M. Al-Rizzo, **R. Akl**, and Z. Elbazzal, "The Effect of an Enhanced Channel Assignment Algorithm in an IEEE 802.11 WLAN," *World Scientific and Engineering Academy and Society Transactions on Communications*, WSEAS, Vol. 8, Issue 12, December 2009.
- J6. **R. Akl**, P. Kadiyala, and M. Haidar, "Non-Uniform Grid-Based Coordinated Routing in Wireless Sensor Networks," *Journal of Sensors*, article ID 491349, volume 2009, 11 pages.
- J7. M. Haidar, M. Al-Rizzo, Y. Chan, **R. Akl**, "User-Based Channel Assignment Algorithm in a Load-Balanced IEEE 802.11 WLAN," *International Journal of Interdisciplinary Telecommunications & Networking (IJITN)*, April-June 2009, 1(2), pp. 66-81.
- J8. **R. Akl**, D. Keathly, and R. Garlick, "Strategies for Retention and Recruitment of Women and Minorities in Computer Science and Engineering," *iNEER Special Volume: Innovations 2007- World Innovations in Engineering Education and Research*, 9 pgs., 2007.
- J9. R. Garlick and **R. Akl**, "Motivating and Retaining CS2 Students with a Competitive Game Programming Project," *iNEER Special Volume: Innovations 2007- World Innovations in Engineering Education and Research*, 9 pgs., 2007.
- J10. **R. Akl** and S. Nguyen, "UMTS Capacity and Throughput Maximization for Different Spreading Factors," *Journal of Networks*, July 2006, vol. 1, issue 3, pp. 40-49. ISSN: 1796-2056
- J11. W. Li, K. Kavi, and **R. Akl**, "A Non-preemptive Scheduling Algorithm for Soft Real-time Systems," *Journal of Computer and Electrical Engineering*, 2006, vol. 32, 18 pgs. ISSN: 0045-7906
- J12. **R. Akl**, A. Parvez, and S. Nguyen, "Effects of Interference on Capacity in Multi-Cell CDMA Networks," *Journal of Systemics, Cybernetics and Informatics*, 2006, vol. 3, no. 1, p825612, 7 pgs. ISSN: 1690-4524
- J13. **R.G. Akl**, M. Hegde and M. Naraghi-Pour, "Mobility-based CAC Algorithm for Arbitrary Traffic Distribution in CDMA Cellular Systems," *IEEE Transactions on Vehicular Technology*, March 2005, vol. 54, no. 2, pp. 639-651.
- J14. **R.G. Akl**, M.V. Hegde, M. Naraghi-Pour, P.S. Min, "Multi-Cell CDMA Network Design," *IEEE Transactions on Vehicular Technology*, May 2001, vol. 50, no. 3, pp. 711-722.

Technical Papers

- T1. J. Williams, **R. Akl**, et al, "Flight Control Subsystem," *The Eagle Feather*, Special Section: Undergraduate Research Initiative in Engineering, University of North Texas, Vol. 7, 2010.
- T2. **R.G. Akl**, M.V. Hegde, A. Chandra, P.S. Min, "CDMA Capacity Allocation and Planning," Technical Document, Washington University Department of Electrical Engineering WUEE-98, April 1998.

Book Chapters

- B1. R. Akl, Y. Saravanos, and M. Haidar, "Chapter 18: Hybrid Approach for Energy-Aware Synchronization in Sensor Networks," *Sustainable Wireless Sensor Networks*, December 2010, pgs. 413-429, ISBN: 978-953-307-297-5.
- B2. K. Kavi, **R. Akl** and A. Hurson, "Real-Time Systems: An Introduction and the State-of-the-Art," *Encyclopedia of Computer Science and Engineering*, John Wiley & Sons, Volume 4, January 2009, pgs. 2369-2377.
- B3. **R. Akl** and K. Kavi, "Chapter 12: Modeling and Analysis using Computational Tools," *Introduction to Queuing Theory: Modeling and Analysis*, Birkhauser Boston, December 2008, pgs. 295-320.

Technical Presentations

- P1. "Bio-Com Project," Raytheon, Richardson TX, May 2012, (invited).
- P2. "Bio-Com Project," Net-Centric Software and Systems I/UCRC Meeting, Denton TX, December 2011, (invited).
- P3. "Student Outreach Report: Robocamp," College of Engineering Advisory Board Meeting, Denton TX, May 2011, (invited).
- P4. "Robocamp: Encouraging Young Women to Embrace STEM," 4th Annual TETC Best Practices Conference, Austin TX, February 2009, (invited).
- P5. "Self-Configuring Wireless MEMS Network (demo)," Southern Methodist University, Dallas TX, January 2008, (invited).
- P6. "Energy-aware Routing and Hybrid Synchronization in Sensor Networks," *Southern Methodist University*, Dallas TX, September 2007, (invited).
- P7. "Retention and Recruitment of Women in Computer Engineering," *ICEE 2006: International Conference on Engineering Education*, Puerto Rico, July 2006, (refereed).

- P8. "Capacity Allocation in Multi-cell UMTS Networks for Different Spreading Factors with Perfect and Imperfect Power Control," *IEEE CCNC 2006: Consumer Communications and Networking Conference*, Las Vegas, NV, January 2006, (refereed).
- P9. "Research, Teaching, and Outreach," CSE Advisory Council Meeting, *UNT Research Park*, Denton, TX, December 2005, (invited).
- P10. "Wi-Fi and WCDMA Network Design," *University of Arkansas*, Little Rock, AR, April 2005, (invited).
- P11. "Wi-Fi and WCDMA Network Design," *Southern Methodist University*, Dallas, TX, March 2005, (invited).
- P12. "Current Research in Wireless at UNT," *Nortel Networks*, Richardson, TX, October 2004, (invited).
- P13. "Subscriber Maximization in CDMA Cellular Networks," *International Conference on Computing, Communications, and Control Technologies*, Austin, TX, August 2004, (refereed).
- P14. "Global versus Local Call Admission Control in CDMA Cellular Networks," *International Conference on Cybernetics and Information Technologies, Systems and Applications*, Orlando, FL, July 2004, (refereed).
- P15. "Impact of Interference Model on Capacity in CDMA Cellular Networks," *8th World Multi-Conference on Systemics, Cybernetics, and Informatics*, Orlando, FL, July 2004, (refereed).
- P16. "CDMA Network Design," *IEEE Communications Society – New Orleans Chapter*, New Orleans, LA, May 2002, (invited).
- P17. "Cell Design to Maximize Capacity in CDMA Networks," *Louisiana State University*, Baton Rouge, LA, April 2002, (invited).
- P18. "Call Admission Control Scheme for Arbitrary Traffic Distribution in CDMA Cellular Systems," *IEEE Wireless Communications and Networking Conference*, Chicago, IL, September 2000, (refereed).
- P19. "Cell Placement in a CDMA Network," *IEEE Wireless Communications and Networking Conference*, September 1999, (refereed).
- P20. "Effects of Call Arrival Rate and Mobility on Network Throughput in Multi-Cell CDMA," *IEEE International Conference on Communications*, June 1999, (refereed).

- P21. "Flexible Allocation of Capacity in Multi-Cell CDMA Networks," *IEEE Vehicular Technology Conference*, May 1999, (refereed).
- P22. "CCAP: A Strategic Tool for Managing Capacity of CDMA Networks," Teleware Co. Ltd., Seoul, South Korea, 1998, (invited).

Courses Developed

- CSCE 5933: LTE Physical Layer Using MATLAB.
Research issues in the design of LTE physical layer and simulate using MATLAB. Topics include modulation and coding, OFDM, channel modeling, MIMO, and link adaption.
- CSCE 6590: Advanced Topics in Wireless Communications & Networks: 4G/LTE.
Research issues in the design of next generation wireless networks: cellular systems, medium access techniques, signaling, mobility management, control and management for mobile networks, wireless data networks, Internet mobility, quality-of-service for multimedia applications, caching for wireless web access, and ad hoc networks.
- CSCE 5933: Fundamentals of VoIP.
Fundamentals of VoIP, with emphasis on network infrastructure implementation and security. Topics include IP protocol suite, SS7, speech-coding techniques, quality of service, session initiation protocol, and security issues.
- CSCE 5540: Introduction to Sensor Networks.
Topics include: design implications of energy (hardware and software), and otherwise resource-constrained nodes; network self-configuration; services such as routing under network dynamics, localization, time-synchronization and calibration; distributed data management, in-network aggregation and collaborative signal processing, programming tools and language support.
- CSCE 5510. Wireless Communication.
- Point-to-point signal transmission through a wireless channel, channel capacity, channel encoding, and multi-user transmissions. First, second, and third generation cellular systems, and mobility management.
- CSCE 3510. Introduction to Wireless Communication.
Fundamentals of wireless communications and networking, with emphasis on first, second, and third generation cellular systems. Topics include point-to-point signal transmission through a wireless channel, cellular capacity, multi-user transmissions, and mobility management.
- CSCE 3020. Communications Systems.

Introduction to the concepts of transmission of information via communication channels. Amplitude and angle modulation for the transmission of continuous-time signals. Analog-to-digital conversion and pulse code modulation. Transmission of digital data. Introduction to random signals and noise and their effects on communication. Optimum detection systems in the presence of noise.

- ENEE 3583. Computer Systems Design I (UNO).
The design process of digital computer systems is studied from the instruction set level, system architecture level, and digital logic level. Topics include machine organization, register transfer notation, processor design, memory design, and input/output considerations. Includes semester project.
- ENEE 3584. Computer Systems Design II (UNO).
The design and evaluation of contemporary computer systems are analyzed to compare the performance of different architectures. Topics include performance metrics, computer arithmetic, pipelining, memory hierarchies, and multiprocessor systems.
- ENEE 3514. Computer Architecture Laboratory (UNO).
Selected experiments examining programmable logic, VHDL and logic synthesis, and including a final design project, to accompany and complement the lecture course ENEE 3584. Three hours of laboratory.

Courses Taught

Spring 2021

- CSCE 2610.2: Computer Organization (no evaluation yet)
- CSCE 3020.1: Communication Systems (no evaluation yet)
- CSCE 6950.743: Dissertation (no evaluation done)

Fall 2020

- CSCE 3010.1: Signals and Systems (4.1 / 5.0)
- CSCE 6940.743: 5G MIMO Systems (no evaluation done)
- CSCE 6950.743: Dissertation (no evaluation done)

Spring 2020

- CSCE 6940.743: 5G MIMO Systems (no evaluation done)
- CSCE 6950.743: Dissertation (no evaluation done)

Fall 2019

- CSCE 5933.3: LTE Physical Layer Using MATLAB (4.3 / 5.0)
- CSCE 6940.743: 5G MIMO Systems (no evaluation done)
- CSCE 6950.743: Dissertation (no evaluation done)

Spring 2019

- CSCE 6940.743: 5G MIMO Systems (no evaluation done)
- CSCE 6940.743: Software Defined Radios (no evaluation done)
- CSCE 6950.743: Dissertation (no evaluation done)

Fall 2018

- CSCE 5933.3: LTE Physical Layer Using MATLAB (4.8 / 5.0)
- CSCE 6940.743: 5G MIMO Systems (no evaluation done)
- CSCE 6950.743: Dissertation (no evaluation done)

Spring 2018

- CSCE 6940.743: 5G MIMO Systems (no evaluation done)
- CSCE 6940.743: Jitter-buffer Management and Interference in VoIP (no evaluation done)
- CSCE 6950.743: Dissertation (no evaluation done)

Fall 2017

- CSCE 5933.3: LTE Physical Layer Using MATLAB (4.9 / 5.0)
- CSCE 6940.743: 5G MIMO Systems (no evaluation done)
- CSCE 6940.743: VoLTE and VoWiFi (no evaluation done)
- CSCE 6950.743: Dissertation (no evaluation done)

Spring 2017

- CSCE 6950.743: Dissertation (no evaluation done)

Fall 2016

- CSCE 5933.3: LTE Physical Layer Using MATLAB (4.7 / 5.0)
- CSCE 6950.743: Dissertation (no evaluation done)

Spring 2016

- CSCE 5950.743: Thesis (no evaluation done)
- CSCE 6950.743: Dissertation (no evaluation done)

Fall 2015

- CSCE 3010.1: Signals and Systems (5.7 / 7.0)
- CSCE 5950.743: Thesis (no evaluation done)
- CSCE 6950.743: Dissertation (no evaluation done)

Spring 2015

- CSCE 5934.743: Directed Study (no evaluation done)
- CSCE 5950.743: Thesis (no evaluation done)
- CSCE 6950.743: Dissertation (no evaluation done)

Fall 2014

- CSCE 3010.1: Signals and Systems (3.32 / 4.00)
- CSCE 5950.743: Thesis (no evaluation done)
- CSCE 6950.743: Dissertation (no evaluation done)
- CSCE 6590.1: Advanced Topics in Wireless Communications & Networks: 4G/LTE (3.79 / 4.00)

Spring 2014

- CSCE 3510.1: Intro to Wireless Communication (808 – Highly Effective)
- CSCE 5510.1: Wireless Communications (808 – Highly Effective)
- CSCE 5950.743: Thesis (no evaluation done)
- CSCE 6950.743: Dissertation (no evaluation done)

Fall 2013

- CSCE 5950.743: Thesis (no evaluation done)
- CSCE 6950.743: Dissertation (no evaluation done)

- CSCE 6590.1: Advanced Topics in Wireless Communications & Networks: 4G/LTE (804 – Highly Effective)
- Spring 2013
- CSCE 4890.743: Directed Study (no evaluation done)
 - CSCE 5950.743: Thesis (no evaluation done)
 - CSCE 6940.743: Individual Research (no evaluation done)
 - CSCE 6950.743: Dissertation (no evaluation done)
- Fall 2012
- CSCE 3010.1: Signals and Systems (793 – Highly Effective)
 - CSCE 5540.1: Intro to Sensor Networks (814 – Highly Effective)
 - CSCE 5950.743: Thesis (no evaluation done)
 - CSCE 6950.743: Dissertation (no evaluation done)
- Spring 2012
- CSCE 3020.1: Communication Systems (809 – Highly Effective)
 - CSCE 3510.1: Intro to Wireless Communication (811 – Highly Effective)
 - CSCE 5510.1: Wireless Communications (817 – Highly Effective)
 - EENG 3810.1: Communication Systems (801 – Highly Effective)
- Fall 2011
- CSCE 3010.1: Signals and Systems (793 – Highly Effective)
 - CSCE 5540.1: Intro to Sensor Networks (824 – Highly Effective)
- Spring 2011
- CSCE 3020.1: Communication Systems (820 – Highly Effective)
 - CSCE 3510.1: Intro to Wireless Communication (812 – Highly Effective)
 - CSCE 5510.1: Wireless Communications (812 – Highly Effective)
 - EENG 3810.1: Communication Systems (826 – Highly Effective)
- Fall 2010
- CSCE 3010.1: Signals and Systems (857 – Highly Effective)
 - CSCE 5540.1: Intro to Sensor Networks (831 – Highly Effective)
- Spring 2010
- CSCE 3020.1: Communication Systems (792 – Highly Effective)
 - CSCE 3510.1: Intro to Wireless Communication (793 – Highly Effective)
 - CSCE 5510.1: Wireless Communications (834 – Highly Effective)
 - EENG 3810.1: Communication Systems (854 – Highly Effective)
- Fall 2009
- CSCE 3010.1: Signals and Systems (4.40 / 5.00)
 - CSCE 5540.1: Intro to Sensor Networks (4.70 / 5.00)
 - EENG 2620.1: Signals and Systems (4.40 / 5.00)
- Spring 2009
- CSCE 3020.1: Communication Systems (4.87 / 5.00)
 - CSCE 3510.1: Intro to Wireless Communication (4.65 / 5.00)
 - CSCE 5510.1: Wireless Communications (4.79 / 5.00)
- Fall 2008
- CSCE 3010.1: Signals and Systems (4.91 / 5.00)
 - CSCE 5540.2: Intro to Sensor Networks (4.10 / 5.00)

- EENG 2620.3: Signals and Systems (4.91 / 5.00)
- Spring 2008
- CSCE 3020.1: Communication Systems (4.68 / 5.00)
 - CSCE 3510.1: Intro to Wireless Communication (3.96 / 5.00)
 - CSCE 5510.1: Wireless Communications (4.75 / 5.00)
- Fall 2007
- CSCE 3010.1: Signals and Systems (4.57 / 5.00)
 - CSCE 5540.2: Intro to Sensor Networks (4.01 / 5.00)
- Summer 2007
- CSCE 3020.1: Fund. of Communication Theory (no evaluation done)
 - EENG 3810.1: Communication Systems (no evaluation done)
- Spring 2007
- CSCE 5510.2: Wireless Communications (4.75 / 5.00)
 - CSCE 5933.6: Fundamentals of VoIP (4.70 / 5.00)
- Fall 2006
- CSCE 3010.1: Signals and Systems (4.58 / 5.00)
 - CSCE 5540.1: Intro to Sensor Networks (4.70 / 5.00)
 - EENG 2620.1: Signals and Systems (4.58 / 5.00)
- Summer 2006
- CSCE 3020.1: Fund. of Communication Theory (no evaluation done)
 - CSCE 3510.21: Intro to Wireless Communications (no evaluation done)
 - CSCE 5510.21: Intro to Wireless Communications (no evaluation done)
 - EENG 3810.1: Communication Systems (no evaluation done)
- Spring 2006
- CSCE 2610.2: Computer Organization (3.69 / 5.00)
 - CSCE 3010.1: Signals and Systems (4.41 / 5.00)
 - EENG 2620.1: Signals and Systems (4.41 / 5.00)
- Fall 2005
- CSCE 3510.1: Intro to Wireless Communications (4.52 / 5.00)
 - CSCE 5510.1: Wireless Communications (4.46 / 5.00)
 - CSCE 5933.6: Intro to Sensor Networks (4.60 / 5.00)
- Summer 2005
- CSCE 3010.21: Signals and Systems (no evaluation done)
 - CSCE 3510.21: Intro to Wireless Communications (no evaluation done)
- Spring 2005
- CSCE 3510.02: Intro to Wireless Communications (4.46 / 5.00)
 - CSCI 3100.02: Computer Organization (4.14 / 5.00)
- Fall 2004
- CSCE 3510.01: Intro to Wireless Communications (4.15 / 5.00)
 - CSCI 4510.01: Machine Structures (4.55 / 5.00)
 - CSCI 5330.02: Intro to Wireless Communications (4.05 / 5.00)
- Summer 2004
- CSCI 4330.22: Intro to Wireless Communications (no evaluation done)
 - CSCI 4330.23: Intro to Wireless Communications (no evaluation done)

- CSCI 5330.22: Intro to Wireless Communications (no evaluation done)

Spring 2004

- CSCI 3100: Computer Organization (4.64 / 5.00)
- CSCI 4330: Intro to Wireless Communications (4.22 / 5.00)

Fall 2003

- CSCI 4510: Machine Structures (4.49 / 5.00)
- CSCI 5330: Intro to Wireless Communications (4.83 / 5.00)

Summer 2003

- CSCI 3100: Computer Organization (no evaluation done)

Spring 2003

- CSCI 3100: Computer Organization (3.84 / 5.00)

Fall 2002

- CSCI 4510: Machine Structures (4.38 / 5.00)

Funded Proposals

- R1. "I/UCRC Industrial Membership - Ashum Corp," 2020. Krishna Kavi (PI), Robert Akl (co-PI), **\$52,000.**
- R2. "I/UCRC Industrial Membership - Ashum Corp," 2019. Krishna Kavi (PI), Robert Akl (co-PI), **\$60,900.**
- R3. "I/UCRC Industrial Membership - Ashum Corp," 2018. Krishna Kavi (PI), Robert Akl (co-PI), **\$57,700.**
- R4. "Robotics and App Design Summer Camp" under Texas Higher Education Coordinating Board: Engineering Summer Program. Requested amount is \$11,727. Submitted 5/5/17. Robert Akl (PI), **\$11,727.**
- R5. "I/UCRC Industrial Membership - Ashum Corp," 2017. Krishna Kavi (PI), Robert Akl (co-PI), **\$50,000.**
- R6. "UNT GenCyber Summer Program: Inspiring the Next Generation of Cyber Stars in North Texas," National Security Agency (NSA). Requested amount is \$85,000. Submitted 11/4/2016. Robert Akl (co-PI), **\$85,000.**
- R7. "App Design Summer Camp" under Texas Higher Education Coordinating Board: Engineering Summer Program. Requested amount is \$12,900. Submitted 5/6/16. Robert Akl (PI), **\$12,900.**
- R8. "I/UCRC Industrial Membership - Ashum Corp," 2016. Krishna Kavi (PI), Robert Akl (co-PI), **\$65,000.**
- R9. "Robotics, Game and App Programming Summer Camps" under Texas Workforce Commission: Summer Merit Program. Requested amount is \$63,000. Submitted 11/16/15. Robert Akl (PI), **\$63,000.**

- R10. “App Design Summer Camp” under Texas Higher Education Coordinating Board: Engineering Summer Program. Requested amount is \$13,998. Submitted 5/1/15. Robert Akl (PI), **\$13,988.**
- R11. “I/UCRC Industrial Membership - Ashum Corp,” 2015. Krishna Kavi (PI), Robert Akl (co-PI), **\$40,000.**
- R12. “App Design Summer Camp” under Texas Higher Education Coordinating Board: Engineering Summer Program. Requested amount is \$12,500. Submitted 5/2/14. Robert Akl (PI), **\$12,500.**
- R13. “I/UCRC Industrial Membership - Ashum Corp,” 2014. Krishna Kavi (PI), Robert Akl (co-PI), **\$46,000.**
- R14. “I/UCRC Industrial Membership - Ashum Corp,” 2013. Krishna Kavi (PI), Robert Akl (co-PI), **\$38,500.**
- R15. “Robotics, Game and App Programming Summer Camps” under Texas Workforce Commission: Summer Merit Program. Requested amount is \$63,000. Submitted 12/14/12. Robert Akl (PI), **\$63,000.**
- R16. “Bio-Com Project,” funded by Raytheon under Net-Centric Software and Systems I/UCRC 2nd year. Requested amount is \$30,000. Submitted 5/12/12. Krishna Kavi (PI), Robert Akl (co-PI), **\$30,000.**
- R17. “Bio-Com Project,” funded by Raytheon under Net-Centric Software and Systems I/UCRC. Requested amount is \$30,000. Submitted 5/12/11. Krishna Kavi (PI), Robert Akl (co-PI), **\$30,000.**
- R18. “Game Programming for Xbox 360 Summer Camp” under Texas Higher Education Coordinating Board: Engineering Summer Program. Requested amount is \$20,000. Submitted 3/21/11. Robert Akl (PI), **\$20,000.**
- R19. “RoboCamps and Game Programming Summer Camps” under Texas Workforce Commission: Summer Merit Program. Requested amount is \$63,000. Submitted 2/17/11. Robert Akl (PI), **\$63,000.**
- R20. “Game Programming for Xbox 360 Summer Camp” under Texas Higher Education Coordinating Board: Engineering Summer Program. Requested amount is \$13,000. Submitted 2/22/10. Robert Akl (PI), **\$18,000.**
- R21. “Robotics and Game Programming Summer Camps” under Texas Workforce Commission: Summer Merit Program. Requested amount is \$63,000. Submitted 10/16/09. Robert Akl (PI), **\$63,000.**

- R22. "Micro Air Vehicle Design: A Collaborative Undergraduate Project for Electrical Engineering, Computer Engineering, and Computer Science Students," under UNT Undergraduate Research Initiative. Submitted 9/25/2009. Robert Akl (co-PI), **\$8,000.**
- R23. "Summer Merit Program" under Texas Workforce Commission. Requested amount is \$42,000. Submitted 3/20/09. Robert Akl (PI), **\$42,000.**
- R24. "Robocamp at Stewpot" under Dallas Women's Foundation. Requested amount is \$20,000. Submitted 2/23/09. Robert Akl (PI), **\$18,600.**
- R25. "Robocamp Jump Start" under Motorola Foundation Innovation Generation Grant. Requested amount is \$29,852. Submitted 2/12/09. Robert Akl (PI), **\$30,700.**
- R26. "Engineering Summer Program" under Texas Higher Education Coordinating Board. Requested amount is \$7,944. Submitted 2/13/09. Robert Akl (PI), **\$11,111.**
- R27. "Texas Youth in Technology" under Texas Workforce Commission. Requested amount is \$152,393. Submitted 11/10/08. Robert Akl (PI), **\$152,393.**
- R28. "I/UCRC Center Proposal: Net-Centric Software and Systems," under NSF-07-537: Industry/University Cooperative Research Centers. Requested amount is \$349,482. Submitted 9/26/08. Krishna Kavi (PI), Robert Akl (co-PI), **\$60,000 per year for 5 years.**
- R29. "Robocamp and Beyond" under Motorola Foundation Innovation Generation Grant. Requested amount is \$30,000. Submitted 6/20/08. Robert Akl (PI), **\$30,000.**
- R30. "Texas Youth in Technology" under Texas Workforce Commission. Requested amount is \$30,000. Submitted 2/27/08. Robert Akl (PI), **\$31,500.**
- R31. "Robocamp Program for Young Women" under RGK foundation. Requested amount is \$30,000. Submitted 11/5/07. Robert Akl (PI), **\$15,000.**
- R32. "Texas Youth in Technology" under Texas Workforce Commission. Requested amount is \$102,514. Submitted 10/22/07. Robert Akl (PI), **\$102,514.**
- R33. "Women Art Technology" under Hispanic and Global Studies Initiatives Fund. Requested amount is \$14,125. Submitted 9/30/07. Jennifer Way (PI), Robert Akl (co-PI), **\$12,785.**
- R34. "Robocamp Mobile Unit" under Motorola Foundation Innovation Generation Grant. Requested amount is \$35,000. Submitted 6/20/07. Robert Akl (PI), **\$30,000.**

- R35. "ICER: UNT Engineering Challenge Camps" under NSF 0547299. Requested amount is \$35,000. Submitted 4/27/07. Oscar Garcia (PI), Robert Akl (senior personnel), **\$32,792**.
- R36. "I/UCRC-Planning Proposal: UNT Research Site Proposal to join Embedded Systems I/UCRC," under NSF-01-116: Industry/University Cooperative Research Centers. Requested amount is \$10,000. Submitted 3/31/07. Krishna Kavi (PI), Robert Akl (co-PI), **\$10,000**.
- R37. "High-assurance NCCS: Ultra Dependability Integration Engineering," Department of Defense. Requested amount is \$20,000. Submitted 3/12/07. Krishna Kavi (PI), Robert Akl (co-PI), **\$20,000**.
- R38. "Recruiting and Retention Strategies for Computer Science at UNT" under Texas Technology Workforce Development Grant Program – 2005. Requested amount is \$163,322. Submitted 3/17/05. Robert Akl (PI), **\$125,322**.
- R39. UNT Faculty Research Grant for Fall 2003, Robert Akl (PI), \$5,000, **\$4,000**.
- R40. UNT Junior Faculty Summer Research Fellowship for Summer 2003, Robert Akl (PI), \$5,000, **\$5,000**.

Professional Associations and Achievements

Membership in Professional Organizations

- Senior Member IEEE
- Member, Federation Council of North Texas Universities
- Member, Eta Kappa Nu Electrical Engineering Honor Society
- Member, Golden Key National Honor Society
- Member, Tau Beta Pi Engineering Honor Society

Offices and Committee Assignments in Professional Organizations

- Technical Program Committee Member, IEEE Wireless Communications and Networking Conference, IEEE WCNC
- Technical Program Committee Member, International Wireless Symposium, IWS
- Technical Program Committee Member, IEEE International Conference on Computational Science, IEEE ICCS
- Technical Program Committee Member, IASTED International Conference on Wireless Communications, WC
- Technical Program Committee Member, WTS Wireless Telecommunications Symposium
- Technical Program Committee Member, Mosharaka International Conference on Computer Science and Engineering, Amman

- Invitation to serve as an NSF reviewer/panelist for Engineering Research Centers (ERC) proposals
- Technical Program Committee Member, 18th IEEE International Symposium on Personal, Indoor and Mobile Radio Communication, Greece
- International Program Committee, IASTED International Conference on Wireless and Optical Communication, Canada
- Program Committee Member, Fifth Annual Wireless Telecommunications Symposium, CA
- Technical Publications Chair, IEEE Vehicular Technology Conference, Dallas TX
- Session Chair, International Conference on Computing, Commun. and Control Tech., Austin TX
- Session Chair, International Conference on Cybernetics and Information Technologies, Orlando FL
- Session Chair, 8th World Multi Conference on Systemics, Cybernetic, and Informatics, Orlando FL

Additional Responsibilities and Activities

- Reviewer, *Wireless Communications and Mobile Computing*, 2012 – present
- Reviewer, *Journal of Sensor and Actuator Networks*, 2012 – present
- Reviewer, *IEEE Transactions on Vehicular Technology*, 2011 – present
- Reviewer, *Elsevier Journal of Computers & Electrical Engineering*, 2008 – present
- Reviewer, *IEEE Globecom*, 2007 – present
- Reviewer, *IEEE International Conference on Advanced Networks and Telecommunication Systems (ANTS)*, 2008 – present
- Reviewer, *The International Wireless Communications and Mobile Computing Conference*, 2007 – present
- Reviewer, *Journal on Wireless Communications and Networking*, 2007 – present
- Reviewer, *IEEE Transactions on Communications*, 2007 - present
- Reviewer, *International Journal of Communication Systems*, 2007 – present
- Reviewer, *IEEE Communications Magazine*, 2005 – present
- Reviewer, *Journal of Wireless Networks*, 2004 – present
- Reviewer, *IEEE Transactions on Mobile Computing*, 2004 – present
- Reviewer, *IEEE Transactions on Wireless Communications*, 2004 – present
- Reviewer, *ACM Crossroads*, 2004 – present

Honors and Awards

- Who's Who in America, 2012 Edition
- Winner of Tech Titan of the Future – University Level Award for UNT Robocamps for Girls, Metroplex Technology Business Council, 2010 with **\$15,000 cash prize**.
- IEEE Professionalism Award, Ft Worth Chapter, 2008
- UNT College of Engineering Outstanding Teacher Award, 2008
- Certificate of Appreciation: IEEE Vehicular Technology Conference, Dallas, TX,

2005

- Certificate of Appreciation: Denton County Boosting Engineering, Science and Technology (BEST) Robotics Competition, 2004
 - Summa Cum Laude Graduate, Ranked First in Undergraduate Class
 - The Computer Science Departmental Award for Academic Excellence, Washington University, 1993
 - The Dual Degree Engineering Award for Outstanding Senior, Washington University, 1993
 - The 1992 Technical Writing Competition Award, The Society for Technical Communication
-