UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

PALO ALTO NETWORKS, INC.,
Petitioner,

v.

CENTRIPETAL NETWORKS, LLC,
Patent Owner.

Case IPR2023-01353
Patent No. 10,567,343

PETITION FOR *INTER PARTES* REVIEW

TABLE OF CONTENTS

I. INTRODUCTION 1

II. MANDATORY NOTICES 4

A. Real Party-In-Interest 4

B. Related Matters 4

C. Lead and Back-up Counsel, and Service Information 5

III. PAYMENT OF FEES 5

IV. BACKGROUND 5

A. Overview of the ’343 Patent 5

B. Prosecution History 10

C. Priority Date of the Challenged Claims 12

D. Patent Family History 12

V. OVERVIEW OF CHALLENGE AND RELIEF REQUESTED 16

A. Grounds for Standing (§ 42.104(a)) 16

D. The Board Should Not Exercise Discretion to Deny Institution 23

2. The Board Should Not Exercise 325(d) Discretion 24

VI. SOURCEFIRE IN VIEW OF EMERGING THREATS RENDERS CLAIMS 1-20 OBVIOUS 26

A. Overview of Sourcefire 26

B. Overview of Emerging Threats 34

D. Independent Claims 1, 8 and 15 45

E. Dependent Claims 58

2. Claims 3, 10, and 17 59

3. Claims 4, 11 and 18 61

(A) [4pre]/[11pre]/[18pre].....61

(B) [4a]/[11a]/[18a].....61

(C) [4b]/[11b]/[18b] 62

(D) [4c]/[11c]/[18c].....62

(E) [4d]/[11d]/[18d] 63

4. Claims 5 and 12.....63

(A) [5pre]/[12pre].....63

(B) [5a]/[12a] 63

(C) [5b]/[12b].....65

5. Claims 6, 13 and 19 65

6. Claims 7, 14 and 20 67

VII. SECONDARY CONSIDERATIONS 69

VIII. CONCLUSION.....71

TABLE OF AUTHORITIES

Page(s)

Cases

Advanced Bionics, LLC v. Med-El Elektromedizinische Gerate GMBH, 2020 Pat. App. LEXIS 13184 (PTAB February 13, 2020) 15

Alphatec Holdings, Inc. et al. v. Nuvasive, Inc., 2020 Pat.App. LEXIS 12487 (PTAB July 8, 2020) 3

Amazon.com, Inc., v. M2M Solutions LLC, IPR2019-01205, Paper 43 (PTAB Jan. 25, 2021) 45

Becton, Dickinson and Co. v. B. Braun Melsungen AG, IPR2017-01586, Paper 8 (PTAB Dec. 15, 2017) 15

Bruckelmyer v. Ground Heaters, Inc., 445 F.3d 1374 (Fed. Cir. 2006).....18

Centripetal Networks, Inc. v. Keysight Technologies, Inc., Civ. No. 2-22-cv-00002 (E.D. Va., filed Jan. 1, 2022) 4

Centripetal Networks, Inc. v. Palo Alto Networks, Inc., Case No. 2:21-cv-00137-RCY-RJK (E.D. Va., filed 3/12/2021)..... 4, 24

MaxLinear, Inc. v. CF CRESPE LLC, 880 F.3d 1373 (Fed. Cir. 2018).....43

Nestlé USA, Inc. v. Steuben Foods, Inc., 884 F.3d 1350 (Fed. Cir. 2018) 18, 43

Nidec Motor Corp. v. Zhongshan Broad Ocean Motor Co., 868 F.3d 1013 (Fed. Cir. 2017) 20

Ohio Willow Wood Co. v. Alps South, LLC, 735 F.3d 1333 (Fed. Cir. 2013).....44

Pohl v. Mh Sub I, LLC, 332 F.R.D. 713 (N.D. Fla. 2019).....19

Soverain Software LLC v. Victoria's Secret Direct Brand Mgmt., LLC, 778 F.3d 1311 (Fed. Cir. 2015)..... 43, 44

SRI Int'l, Inc. v. Internet Sec. Sys., Inc., 511 F.3d 1186 (Fed. Cir. 2008).....18

United Access Techs., LLC v. CenturyTel Broadband Services LLC, 778 F.3d 1327 (Fed. Cir. 2015) 44

Valve Corp. v. Ironburg Inventions Ltd., 8 F.4th 1364 (Fed. Cir. 2021) 18

Veeam Software Corp. v. Symantec Corp., IPR2013-00142, Paper 11 (Aug. 7, 2013).....18

Vivid Techs., Inc. v. Am. Sci. & Eng'g, Inc., 200 F.3d 795 (Fed. Cir. 1999) 20

Webpower, Inc. v. WAG Acquisition, LLC, IPR2016-01239, Paper 21 (PTAB July 8, 2020) 45

LIST OF EXHIBITS

Exhibit No.	Description
EX-1001	U.S. Patent No. 10,567,343 ("the '343 Patent")
EX-1002	File History of the '343 Patent ("File History")
EX-1003	Declaration of Dr. Stuart Staniford in support of Petition for *Inter Partes* Review of U.S. Patent No. 10,567,343
EX-1004	Sourcefire 3D System User Guide Version 4.10 ("Sourcefire"), *available* at https://www.scribd.com/document/96860221/Sourcefire-3D-System-User-Guide-v4-10
EX-1005	IPR2018-01760 CS-1005: Declaration of John Leone
EX-1006	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
EX-1007	Barry M. Leiner et al., "A Brief History of the Internet," Internet Society, 1997, available at: https://www.internetsociety.org/wp-content/uploads/2017/09/ISOC-History-of-theInternet_1997.pdf
EX-1008	U.S. Pat. No. 7,032,031 ("Jungck")
EX-1009	E. H. Spafford, "The Internet Worm Program: An Analysis," *Purdue Technical Report CSD-TR-823*, 1988, available at: https://spaf.cerias.purdue.edu/tech-reps/823.pdf
EX-1010	Eichin, M.W. and Rochlis, J.A., "With Microscope and Tweezers: The Worm from MIT's Perspective," *Communications of the ACM*, Volume 32 Number 6, June 1989; *available at* https://www.cs.jhu.edu/~fabian/courses/CS600.624/Rochlis_morris.pdf
EX-1011	T. Eisenberg *et al.*, "The Cornell Commission: On Morris and the Worm," *Communications of the ACM*, Volume 32 Number 6, June 1989; *available* at https://www.cs.cornell.edu/courses/cs1110/2009sp/assignments/a1/p706-eisenberg.pdf

EX-1012	J. C. Mogul, "Simple and Flexible Datagram Access Controls for Unix-based Gateways," Digital Western Research Laboratory Technical Report No. 89/4, 1989; *available* at https://www.hpl.hp.com/techreports/Compaq-DEC/WRL-89-4.pdf
EX-1013	*Intentionally Left Blank*
EX-1014	*Intentionally Left Blank*
EX-1015	*Intentionally Left Blank*
EX-1016	Heberlein, L. T.L, *et al.*, "A Network Security Monitor," *Proceedings, 1990 IEEE Computer Society Symposium on Security and Privacy*, 1990, *available at* https://ieeexplore.ieee.org/document/63859.
EX-1017	IPR2018-01760 CS-1017: "NetRanger Network Security Management System User Guide". v1.3.1 *Wheelgroup Corporation*, 1997.
EX-1018	Roesch, M., "Snort: Lightweight Intrusion Detection for Networks," *Lisa* (Vol. 99, No. 1, pp. 229-238), 1999, *available* at https://www.usenix.org/legacy/event/lisa99/full_papers/roesch/roesch.pdf
EX-1019	U.S. Pat. No. 8,370,936 ("Zuk")
EX-1020	Emerging Threats, available at https://web.archive.org/web/20101202025325/http://rules.emergingthreats.net/open/snort-2.8.4/emerging-all.rules (Dec. 2, 2010).
EX-1021	Authorized Classification and Control Markings Register, Vol. 1, Ed. 2 (Version 1.2), May 12, 2008, available at http://sgp.fas.org/othergov/intel/register.pdf
EX-1022	"Transmission Control Protocol," *IETF RFC 793*. 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.
EX-1023	"Internet Protocol," *IETF RFC 791*, J. Postel, ed., September 1981, archived at Internet Archive Wayback Machine on February 4, 2007 at

	https://web.archive.org/web/20070204151303/http://tools.ietf.org/html/rfc791.
EX-1024	"File Transfer Protocol," *IETF RFC 765*, J. Postel, ed., June 1980, archived at Internet Archive Wayback Machine on February 6, 2007 at https://web.archive.org/web/20070206005843/https://tools.ietf.org/html/rfc765.
EX-1025	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.
EX-1026	I. Ristic, "SSL/TLS and PKI History," April 2018, available at https://www.feistyduck.com/ssl-tls-and-pki-history/.
EX-1027	Service Name and Transport Protocol Port Number Registry, Internet Assigned Numbers Authority (IANA), available at https://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xhtml
EX-1028	Understanding /etc/password File Format, available at https://www.cyberciti.biz/faq/understanding-etcpasswd-file-format/
EX-1029	ASCII Table, available at https://www.asciitable.com/
EX-1030	Affidavit of Nathaniel E Frank-White, Internet Archive, Archive.org (Dec. 12, 2022)
EX-1031	*Intentionally Left Blank*
EX-1032	K. Ingham & S. Forrest, "A History and Survey of Network Firewalls," *University of New Mexico Computer Science Department Technical Report 2002-37*, 2002, *available* at https://www.cs.unm.edu/~treport/tr/02-12/firewall.pdf.
EX-1033	Sourcefire Support Website Webpage, archived at Internet Archive Wayback Machine on April 30, 2011, *available* at https://web.archive.org/web/20110430194135/https:/support.sourcefire.com/ ("Sourcefire Support Webpage").
EX-1034	Crow, J., Bennet, B.; Structure of Social Security Numbers, available at https://www.eff.org/pages/structure-social-security-numbers
EX-1035	*Curriculum Vitae* of Stuart Staniford

EX-1036	Webster's New World Computer Dictionary, Tenth Edition (2003)
EX-1037	Microsoft Computer Dictionary, Fifth Edition (2002)
EX-1038	T. Dierks & E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.1," IETF RFC 4346, 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
EX-1039	*Intentionally Left Blank*
EX-1040	Minoli, Daniel, et al., Minoli-Cordovana's Authoritative Computer & Network Security Dictionary (2006)
EX-1041	Dictionary Of Computing & Communications, McGraw-Hill (2003)
EX-1042	IPR2018-01436 CS-1003: Declaration of Dr. Stuart Staniford in Support of Petition for Inter Partes Review of US Patent No. 9,124,552.
EX-1043	U.S. Patent No. 9,124,552
EX-1044	IPR2018-01437 CS-1003: Declaration of Dr. Stuart Staniford in Support of Petition for Inter Partes Review of US Patent No. 9,160,713
EX-1045	IPR2018-01760 CS-1038, Declaration of Christopher Butler
EX-1046	IPR2018-01760 CS-1042, Declaration of Jacob H. Baugher III
EX-1047	U.S. Patent No. 8,356,357
EX-1048	U.S. Patent No. 8,141,127
EX-1049	Declaration of John M. Hendricks In Support IPR Petition for U.S. Pat. 10,193,917
EX-1050	M. Edwards, "Bleeding Edge of Snort," Oct. 10, 2004, available at https://www.itprotoday.com/windows-78/bleeding-edge-snort
EX-1051	The Aggregation Point for Snort Signatures and Research, Bleedingsnort.com, Wayback Machine, Internet Archive, available at https://web.archive.org/web/20041230024004/http:/www.bleedingsnort.com/
EX-1052	*Intentionally Left Blank*

EX-1053	Emerging Threats Rule Documentation Wiki, available at ttps://doc.emergingthreats.net/
EX-1054	Proofpoint Emerging Threats Rules, available at https://rules.emergingthreats.net/
EX-1055	*Intentionally Left Blank*
EX-1056	*Intentionally Left Blank*
EX-1057	*Intentionally Left Blank*
EX-1058	U.S. Patent No. 9,160,713
EX-1059	*Intentionally Left Blank*
EX-1060	*Intentionally Left Blank*
EX-1061	*Intentionally Left Blank*
EX-1062	IPR2018-01559 CS-1003: Declaration of Dr. Stuart Staniford in Support of Petition for Inter Partes Review of US Patent No. 9,686,193
EX-1063	U.S. Patent No. 9,686,193
EX-1064	*Intentionally Left Blank*
EX-1065	*Intentionally Left Blank*
EX-1066	*Intentionally Left Blank*
EX-1067	Cisco Systems, Inc., v. Centripetal Networks, Inc., IPR2018-01436 at Paper 40 (Final Written Decision) (U.S. Pat. 9,124,552)
EX-1068	Cisco Systems, Inc., v. Centripetal Networks, Inc., IPR2018-01437 at Paper 40 (Final Written Decision) (U.S. Pat. 9,160,713)
EX-1069	Centripetal Networks, Inc. v. Cisco Sys., 2020-1635 (Fed. Cir. Mar. 10, 2021), available via https://cafc.uscourts.gov/opinions-orders/20-1635.opinion.3-10-2021_1745854.pdf (decision on appeal from ‘552 and ‘713 FWDs)
EX-1070	Cisco Systems, Inc., v. Centripetal Networks, Inc., IPR2018-01559 at Paper 7 (Decision Denying Institution of Inter Partes Review) (U.S. Pat. 9,686,193)
EX-1071	*Intentionally Left Blank*

EX-1072	*Intentionally Left Blank*
EX-1073	"Hypertext Transfer Protocol -- HTTP/1.1" IETF RFC 2616. R. Fielding, et al., June 1999, archived at Internet Archive Wayback Machine on January 9, 2005 at https://web.archive.org/web/20050119021312/https://www.ietf.org/rfc/rfc2616.txt.
EX-1074	HTTP-Requests, available at https://www.tutorialspoint.com/http/http_requests.htm
EX-1075	HTTP – Responses, available at https://www.tutorialspoint.com/http/http_responses.htm
EX-1076	IPR2021-01155 CS-1003: Declaration of Dr. Robert Akl in Support of Petition for Inter Partes Review of US Patent No. 10,567,343
EX-1077	Palo Alto Networks, Inc., v. Centripetal Networks, Inc., IPR2021-01155 at Paper 12 (Decision Denying Institution of Inter Partes Review) (U.S. Pat. 10,567,343)
EX-1078	Emerging Threats FAQ, available at https://doc.emergingthreats.net/bin/view/Main/EmergingFAQ
EX-1079	*Intentionally Left Blank*
EX-1080	Snort rules 2.4 ftp.rules, available via https://web.archive.org/web/20060112054053/http://www.snort.org/pub-bin/downloads.cgi/Download/vrt_pr/snortrules-pr-2.4.tar.gz
EX-1081	Declaration of Dr. Sean Moore in Support of Centripetal's Responses to Non-Final Office Action in Re-examination of '193 patent.
EX-1082	Centripetal Networks Press Release, "Centripetal Networks Inc. Adds Emerging Threats IQRisk® Rep List and IQRisk Query for Their RuleGate® Appliance". Downloaded from https://www.centripetal.ai/press-releases/centripetal-emerging-threats-rulegate/
EX-1083	Cisco Systems, Inc., v. Centripetal Networks, Inc., IPR2018-01559, Patent Owner's Preliminary Response. (U.S. Pat. 9,686,193).
EX-1084	Palo Alto Networks, Inc., v. Centripetal Networks, Inc., IPR2021-01155, Patent Owner's Preliminary Response. (U.S. Pat. 10,567,343).
EX-1085	Ex-Parte Re-examination History for Patent 9,686,193.

EX-1086	Palo Alto Networks, Inc., v. Centripetal Networks, Inc., IPR2021-01155, Paper 16: Decision Denying Petitioner's Request for Rehearing.
EX-1087	Cisco Systems, Inc., v. Centripetal Networks, Inc., IPR2018-01436, Patent Owner's Preliminary Response. (U.S. Pat. 9,124,552).
EX-1088	Gartner, August 2011. Magic Quadrant for Content-Aware Data Loss Prevention. Obtained by SS from https://www.etruserve.com.tw/gtb/GTB%20wp/rp-gartner-magic-quadrant-dlp-2011.pdf
EX-1089	US Patent No. 8,565,108.
EX-1090	US Patent No. 7,467,202.
EX-1091	*Intentionally Left Blank*
EX-1092	*Intentionally Left Blank*
EX-1093	*Intentionally Left Blank*
EX-1094	S. Turner & C. Polk, "Prohibiting Secure Sockets Layer (SSL) Version 2.0," IETF RFC 6176, 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.
EX-1095	M. Chatel, "Classical vs. Transparent IP Proxies," *IETF RFC 1919*, March 1996, available at https://web.archive.org/web/20070630145311/https://tools.ietf.org/html//rfc1919
EX-1096	E. Rescorla, "HTTP over TLS", *IETF RFC 2818*, 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.
EX-1097	J. Moser, "The First Few Milliseconds of an HTTPS Connection", *Moserware*, 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

EX-1098	Apple Quicktime RTSP URL buffer overflow. IBM advisory for CVE-2007-0015, 2007, available at https://exchange.xforce.ibmcloud.com/vulnerabilities/31203
EX-1099	Real Time Streaming Protocol (RTSP). RFC 2326, April 1998. Available at https://www.rfc-editor.org/rfc/rfc2326.html

LISTING OF CLAIMS

CLAIM 1:

[1pre]	A method comprising:
[1a]	receiving, by a computing system comprising memory and at least one processor, a plurality of packets, wherein the plurality of packets comprises a first portion of packets and a second portion of packets;
[1b]	determining, 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;
[1c]	responsive to a determination 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, applying, 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;
[1d]	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; and
[1e]	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, 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.

CLAIM 2:

[2]	The method of claim 1, wherein: 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.

CLAIM 3:

[3]	The method of claim 1, wherein: 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.

CLAIM 4:

[4pre]	The method of claim 1, wherein
[4a]	the application header field value identifies one or more hypertext transfer protocol (HTTP) methods,
[4b]	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:
[4c]	allowing each packet of the third portion of packets to continue toward its respective destination; and
[4d]	blocking each packet of the fourth portion of packets from continuing toward its respective destination.

CLAIM 5:

[5pre]	The method of claim 1, the method further comprising:
[5a]	allowing, based on a determination that the application header field value indicates a request for specified resource data, each packet of the second portion of packets indicating a request for specified resource data to continue toward its respective destination; and

[5b]	blocking, 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 specified resource from continuing toward its respective destination.

CLAIM 6:

[6]	The method of claim 1, further comprising dropping 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.

CLAIM 7:

[7]	The method of claim 1, further comprising dropping 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.

CLAIM 8:

[8pre]	An apparatus comprising: at least one processor; and memory storing instructions that, when executed by the at least one processor, cause the apparatus to:
[8a]	receive a plurality of packets;
[8b]	determine, based on a packet header field value, whether the plurality of packets comprises data corresponding to first criterion specified by one or more packet-filtering rules;
[8c]	responsive to a determination that a packet header field value of a first portion of packets comprises data corresponding to the first criterion specified by at least one matching packet-filtering rule, apply, to each packet in the first portion of packets, one or more operators specified by the at least one matching packet-filtering rule;

[8d]	determine, based on an application header field value, 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
[8e]	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, apply, to each packet in the second portion of packets, 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.

CLAIM 9:

[9]	The apparatus of claim 8, wherein 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.

CLAIM 10:

[10]	The apparatus of claim 8, wherein: 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.

CLAIM 11:

[11pre]	The apparatus of claim 8, wherein
[11a]	the application header field value identifies one or more hypertext transfer protocol (HTTP) methods,

[11b]	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, and wherein the instructions, when executed by the at least one processor cause the apparatus to:
[11c]	allow each packet of the third portion of packets to continue toward its respective destination; and
[11d]	block each packet of the fourth portion of packets from continuing toward its respective destination.

CLAIM 12:

[12pre]	The apparatus of claim 8, wherein the instructions, when executed by the one or more computing devices, cause the apparatus to:
[12a]	allow, based on a determination that the application header field value indicates a request for data from a specified resource, each packet of the second portion of packets indicating a request for data from a specified resource to continue toward its respective destination; and
[12b]	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 a specified from continuing toward its respective destination.

CLAIM 13:

[13]	The apparatus of claim 8, wherein the instructions, when executed by the one or more computing devices, cause the apparatus to 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.

CLAIM 14:

[14]	The apparatus of claim 8, wherein the instructions, when executed by the one or more computing devices, cause the apparatus to 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.

CLAIM 15:

[15pre]	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:
[15a]	receive a plurality of packets;
[15b]	determine, based on a packet header field value, whether the plurality of packets comprises data corresponding to first criterion specified by one or more packet-filtering rules;
[15c]	responsive to a determination that a packet header field value of a first portion of packets comprises data corresponding to the first criterion specified by at least one matching packet-filtering rule, apply, to each packet in the first portion of packets, one or more operators specified by the at least one matching packet-filtering rule;
[15d]	determine, based on an application header field value, 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

[15e]	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, apply, 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.

CLAIM 16:

[16]	The one or more non-transitory computer-readable media of claim 15, wherein: 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.

CLAIM 17:

[17]	The one or more non-transitory computer-readable media of claim 15, wherein: 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.

CLAIM 18:

[18pre]	The one or more non-transitory computer-readable media of claim 15, wherein
[18a]	the application header field value identifies one or more hypertext transfer protocol (HTTP) methods,

[18b]	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, and wherein the instructions, when executed by the one or more computing devices, cause the one or more computing devices to:
[18c]	allow each packet of the third portion of packets to continue toward its respective destination; and
[18d]	block each packet of the fourth portion of packets from continuing toward its respective destination.

CLAIM 19:

[19]	The one or more non-transitory computer-readable media of claim 15, further comprising dropping packets associated with a particular type of data transfer responsive to a determination that the packets associated with the particular type of data transfer comprises data corresponding to a particular transport layer security (TLS) version value.

CLAIM 20:

[20]	The one or more non-transitory computer-readable media of claim 15, further comprising dropping packets associated with a particular type of data transfer responsive to a determination that the packets associated with the particular type of data transfer comprises data corresponding to a particular real-time transport protocol value.

I. INTRODUCTION

Petitioner Palo Alto Networks, Inc. ("Petitioner") requests *inter partes* review of claims 1-20 ("Challenged Claims") of U.S. Patent 10,567,343 ("'343 Patent") (EX-1001) in accordance with §§ 311-319 and §§ 42.100 *et seq*.[1]

The '343 patent concerns "filtering network data transfers." EX-1001 Abstract. The specification discusses preventing "exfiltration," which involves extracting information without authorization from a network using "popular network data transfer protocols" to disguise the exfiltration "as normal network behavior." EX-1001 1:28-36. The patent says prior systems "interpret[ed] exfiltrations as trusted operations." EX-1001 1:36-38. The patent describes receiving packets and determining whether their header field values correspond to a packet-filtering rule. EX-1001 Abstract. If so, an operator determines whether the filtered packets have application header field values corresponding to criteria specified by the operator, and, if so, applies a packet transformation function to prevent an exfiltration. EX-1001 Abstract, 2:40-47. During prosecution, Patent Owner amended the claims to recite a packet transformation function configured "to prevent an exfiltration operation" and argued that this feature distinguished the claims over the prior art. EX-1077 15. The Examiner allowed the claims based on that amendment and argument. *Id.*

[1] Section cites are to 35 U.S.C. or 37 C.F.R. as context indicates.

Petitioner then sought IPR of the '343 patent, contending that most claims were obvious in view of Sourcefire (EX-1004), a user guide for a rule-based network-threat detection and prevention system, and that some dependent claims were obvious in view of Sourcefire in view of another reference. EX-1077 9. The Board denied institution of that IPR under § 325(d), finding that Sourcefire was considered during prosecution of the '343 patent and "the Examiner allowed the claims because of the exfiltration operation," and that Petitioner advanced substantially the same arguments that were already considered. EX-1077 11-19.

This petition presents new prior art that discloses using Sourcefire to detect and prevent exfiltrations. Sourcefire implemented the industry-standard Snort rules-based detection engine, and allowed users to import various publicly available Snort rules. EX-1003 ¶171; EX-1004 1857-1859. Snort began as an open-source project in the late 1990s and allowed users to write two-stage rules for detecting and blocking packets meeting various criteria. EX-1003 ¶¶111-112, 121. By 2010, the Emerging Threats project was a leading source of such Snort rules. EX-1003 ¶¶202-206. A publication of Emerging Threats rules from 2010 (EX-1020, "Emerging Threats") shows Snort rules for preventing exfiltrations as claimed in the '343 patent. A person of ordinary skill in the art ("POSA") would have been motivated to use the Emerging Threats rules with Sourcefire, as Sourcefire expressly teaches

loading and using independent rules. EX-1004 1857-1859, 435-438; EX-1003 ¶¶162-164, 207-208.

The Board found Sourcefire to be prior art to U.S. Patent 9,124,552 ("'552 patent"), the grandparent of the '343 patent, and that it rendered the '552 claims obvious. EX-1067 23-31, 66-67. The Federal Circuit affirmed, rendering those findings binding. EX-1069 17-21 (affirming unpatentability in view of Sourcefire); *Alphatec Holdings, Inc. et al. v. Nuvasive, Inc.*, 2020 Pat.App. LEXIS 12487, *25 (PTAB July 8, 2020) ("as to a different petitioner, when the prior decisions finding claims unpatentable have 'subsequently been affirmed by [the Federal Circuit] those prior decisions, having been affirmed by [the Federal Circuit], are binding in this proceeding, as a matter of collateral estoppel.'") (citation omitted). Emerging Threats, which was not considered by the Examiner or in the prior '343 IPR, discloses the additional limitations in the '343 claims.

Sourcefire and Emerging Threats show that it was known to use Sourcefire for detecting and preventing exfiltrations long before the '343 patent. As detailed below, Sourcefire in view of Emerging Threats renders all Challenged Claims obvious.

II. MANDATORY NOTICES

A. Real Party-In-Interest

Petitioner identifies Palo Alto Networks, Inc. as the real party-in-interest. No other party had access to or control over the present Petition, and no other party funded or participated in preparation of the present Petition.

B. Related Matters

The ’343 Patent is currently the subject of IPR2023-00446, which was filed on 1/6/2023 and instituted 8/7/2023. The grounds presented herein are identical to the grounds presented in IPR2023-00446 (“Keysight IPR”), and Petitioner files concurrently with this Petition a Motion for Joinder.

Petitioner further identifies that the ’343 Patent is currently asserted in *Centripetal Networks, Inc. v. Palo Alto Networks, Inc.*, Case No. 2:21-cv-00137-RCY-RJK (E.D. Va., filed Mar. 12, 2021) (the “Parallel Litigation”) and *Centripetal Networks, Inc. v. Keysight Technologies, Inc.*, Civ. No. 2-22-cv-00002 (E.D. Va., filed Jan. 1, 2022).

C. Lead and Back-up Counsel, and Service Information

Lead Counsel	Backup Counsel
James L. Davis, Jr. Reg. No. 57,325 ROPES & GRAY LLP 1900 University Avenue, 6th Floor East Palo Alto, CA 94303-2284 P: 650-617-4794 / F: 617-235-9492 james.l.davis@ropesgray.com Customer No. 28120 **Mailing address for all PTAB correspondence:** ROPES & GRAY LLP IPRM—Floor 43 Prudential Tower 800 Boylston Street Boston, Massachusetts 02199-3600	Andrew Radsch (*pro hac vice* forthcoming) ROPES & GRAY LLP 1900 University Ave., 6th Floor East Palo Alto, CA 94303-2284 Phone: 650-617-4000 Fax: 617-235-9492 andrew.radsch@ropesgray.com

Petitioner consents to electronic service of documents to the email addresses of the counsel identified above.

III. PAYMENT OF FEES

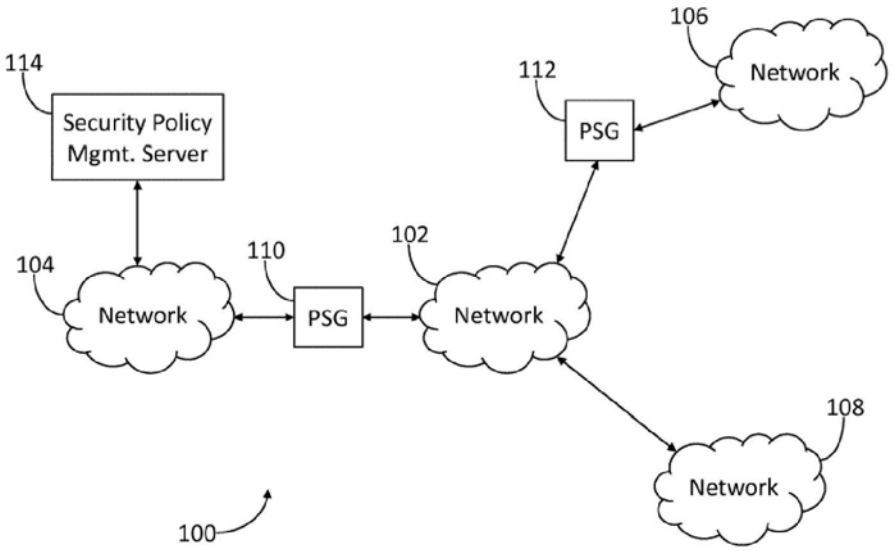
The undersigned authorizes the Office to charge the fee required by § 42.15(a) and any additional fees that might be due to Deposit Account No. 18-1945, under Order No. 115271-0003-679.

IV. BACKGROUND

A. Overview of the ’343 Patent

The ‘343 patent is directed to “filtering network data transfers” to protect data from being stolen by “exfiltrations.” EX-1001 Abstract, 1:28-31. According to the

patent, exfiltrations are often facilitated using popular data transfer protocols, such as the Hypertext Transfer Protocol (HTTP), and such attacks may appear to be "normal network behavior" and interpreted by firewalls as "trusted operations." EX-1001 1:28-51; EX-1003 ¶35.



The '343 patent describes a two-stage filtering process for stopping such attacks implemented at a Packet Security Gateway (PSG). EX-1001 8:39-52; EX-1003 ¶36. Figure 1 (above) shows a network environment 100, including PSGs 10 and 112 placed at boundaries of networks they protect (e.g., from exfiltrations) and a security policy management server 114. EX-1001 4:15-41, 3:62-4:14, 5:29-49; EX-1003 ¶¶37-38. As the PSG receives packets, it applies packet-filtering rules, which may implement "operators." EX-1001 3:62-4:14, 8:39- 52.

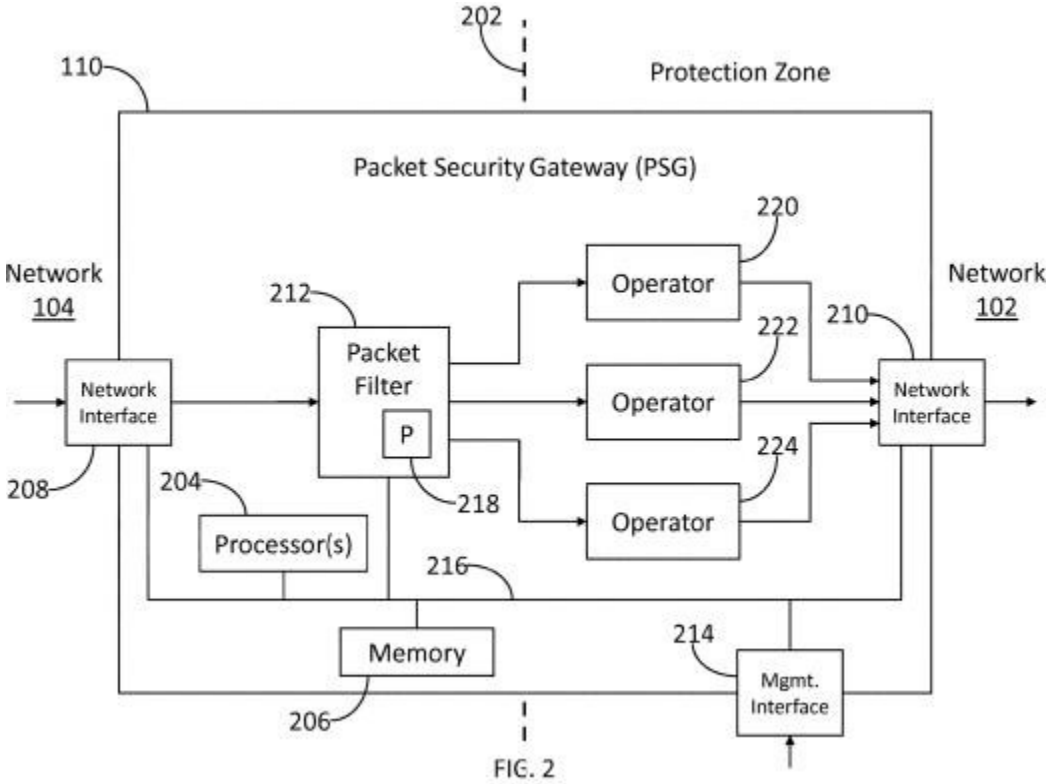


FIG. 2

In the first stage, the "5-tuple" of packet header fields (source and destination IPs and ports and transport protocol) are filtered. EX-1001 5:20-49. In the second stage, the operators may filter application packet header field values. EX-1001 5:40-49; 8:39-52; EX-1003 ¶39.

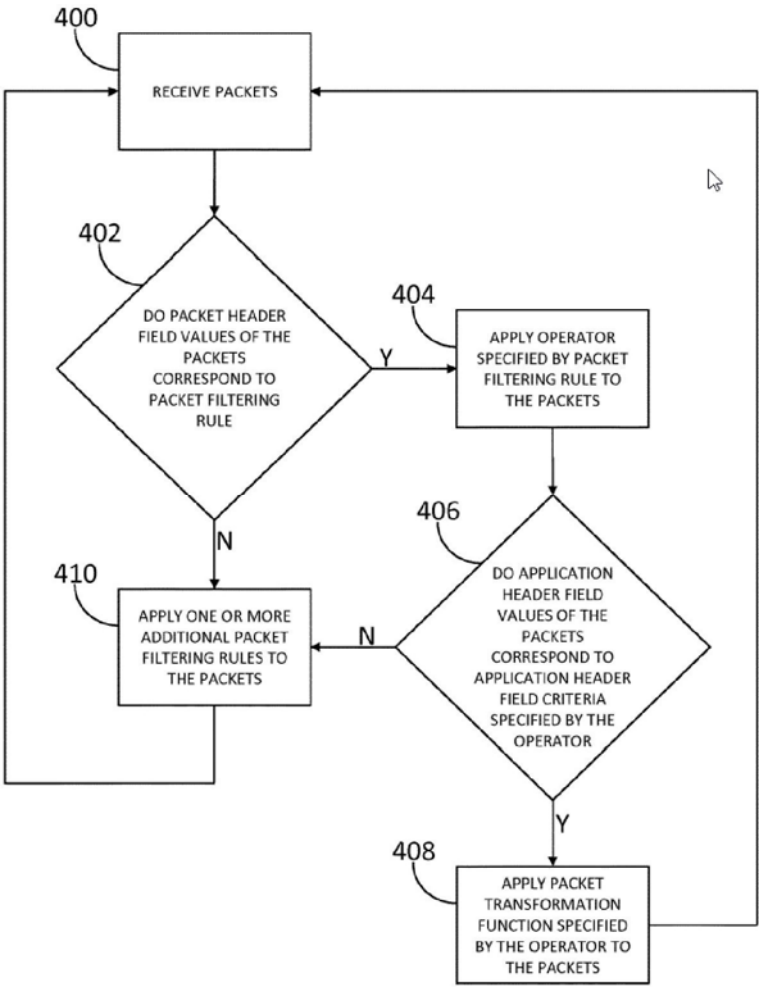


Figure 4 (above) illustrates the two-stage filtering process. Packets are received (step 400) and the PSG determines whether packet header field values (e.g., source or destination IP and/or port) correspond to a packet-filtering rule (step 402). EX-1001 9:18-30; EX-1003 ¶¶40-41. If the received packets' header field values correspond to a rule, the PSG applies an operator specified by the rule (step 404) and the PSG determines whether an application header field value of the received packets corresponds to application header field criteria specified by the operator (step 406). EX-1001 9:30-42; EX-1003 ¶¶42-43. Upon determining that received packets have

an application header field value that corresponds, the PSG applies a "packet transformation function," such as blocking or allowing the packets (step 408). EX-1001 7:41-47, 9:47-61; EX-1003 ¶44.

218

Five-tuple

Rule #	IP Protocol	Source IP Address	Source Port	Destination IP Address	Destination Port	Operator
1 (302)	TCP	140.210.*	*	140.212.*	22	ALLOW
2 (304)	TCP	140.210.*	*	140.212.*	25	ALLOW
3 (306)	TCP	140.210.*	*	140.212.*	110	ALLOW
4 (308)	TCP	140.210.*	*	140.212.*	143	ALLOW
5 (310)	TCP	140.210.*	*	140.212.*	443	REQUIRE-TLS-1.1-1.2
6 (312)	TCP	140.210.*	*	214.*	80	HTTP-EXFIL
7 (314)	*	*	*	*	*	BLOCK

Figure 3 (above) illustrates sample rules, each of which may specify criteria and one or more operators that may be applied to packets matching specified criteria. EX-1003 ¶¶46-47. For example, rule 1 specifies that IP packets containing TCP packets from an IP 140.210.* (IP addresses that begin with 140.210), having any source port, destined for an IP 140.212.*, and destined for port 22 (typically used for Secure Shell (SSH) for logins) have an ALLOW operator. Rules 2-4 similarly apply ALLOW or BLOCK operators based on specified 5-tuple criteria. EX-1001 5:63-6:24; EX-1003 ¶48.

Rules 5 and 6 have both 5-tuple criteria (first stage) and application-header-field-value criteria (second stage). EX-1001 6:25-7:23, 7:41-47, 8:11-38, 7:4-23;

EX-1003 ¶¶49-53. For example, rule 6 specifies that a PSG applies an HTTP-EXFIL operator to IP packets containing TCP packets from the same sources and ports as rule 1 but destined for port 80 (associated with HTTP) on an IP address 214.* (or 214.0.0.0/8). EX-1001 Fig. 3, 7:46-26. The HTTP-EXFIL operator allows HTTP packets containing a GET method, but blocks HTTP packets containing other HTTP methods (e.g., PUT, POST, CONNECT, etc.). "Because attackers may often use HTTP PUT or POST methods to exfiltrate sensitive data, operators such as HTTP-EXFIL of rule 6 may be used to stop such exfiltrations." EX-1001 7:4-23, 41-8:6; EX-1003 ¶¶50, 52-53, 55.

Rule 7 applies a BLOCK operator to any packets that do not match the first-stage criteria for rules 1-6. EX-1001 6:34-39; EX-1003 ¶51.

The '343 patent claims recite methods (claims 1-7), apparatuses (claims 8-14), and computer-readable media (claims 15-20) for implementing the two-stage filtering process to prevent exfiltrations. EX-1001 11:28-15:7; EX-1003 ¶54.

B. Prosecution History

The '343 patent issued from application 15/614,956, which is a continuation of application 14/625,486 (now Patent 9,686,193 ("'193 patent," EX-1063)), which is a continuation of application 13/795,822 (now the '552 patent), filed March 12, 2013. EX-1001 code (63).

During prosecution, the Examiner rejected the then-pending claims as obvious. EX-1002 290-305; EX-1003 ¶¶65-66. In response, on March 27, 2019, Applicant amended claim 1 to differentiate between "a first criterion" specified to examine a packet header and a "second criterion" specified to examine an "application header field value," to recite a "matching" packet-filtering rule, and to recite a packet transformation function configured to "an prevent exfiltration operation" (*sic*), among other changes. EX-1002 271-276; EX-1003 ¶67. The Examiner responded by rejecting the claims as obvious in view of a modified set of prior art. EX-1002 99-115; EX-1003 ¶68.

On November 4, 2019, Applicant amended the claims to recite at least one packet transformation function configured to "prevent an exfiltration operation, where the at least one packet transformation function indicates whether each packet in the second portion of packets is allowed to continue toward its destination." EX-1002 77-82; EX-1003 ¶69. Applicant also 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 claimed. EX-1002 87.

Based on this claim amendment and argument, Examiner entered a Notice of Allowance. EX-1002 39-40. In the reasons for allowance, Examiner specifically noted that, in view of these claim amendments and arguments, the obviousness and other rejections had been "withdrawn," in turn allowing the claims because of the exfiltration operation. EX-1002 40; EX-1003 ¶70. Hence, "the Examiner considered preventing an exfiltration operation pertinent to allowance of the claims." EX-1077 at 15.

The Examiner did not consider Emerging Threats during prosecution. EX-1001 code (56). Nor did the Examiner consider Sourcefire in view of any other publication of Snort rules for preventing exfiltrations.

C. Priority Date of the Challenged Claims

The '343 Patent claims priority to an application filed on March 12, 2013, which Petitioner assumes is the priority date for purposes of this petition. EX-1001 code (63); EX-1003 ¶¶56-57.[2]

D. Patent Family History

As noted above, the '343 patent is a continuation of the '193 patent, which is a continuation of the '552 patent. EX-1001 code (63). U.S. Patent 9,160,713 ("'713 patent," EX-1058) is another continuation of the '552 patent.

[2] Petitioner reserves the right to challenge the priority claim in other proceedings.

Cisco brought an IPR challenging the '552 patent claims as obvious in view of Sourcefire. EX-1067 2. In that proceeding, Patent Owner argued that the petition should be denied because of failure to construe the term "operator." EX-1067 14; EX-1003 ¶146. The Board rejected that argument and found all the claims of the '552 Patent obvious in view of Sourcefire. EX-1067 15-17, 66-67. In doing so, the Board agreed with Cisco that "Sourcefire discloses an 'operator' that specifies (1) the keyword and argument that indicates 'application-header-field-value criteria,' e.g., TLS version 1.0, and (2) a 'packet transformation function,' e.g., blocking packets that match the criteria." EX-1067 50. The Federal Circuit affirmed. EX-1069 17 (affirming that Sourcefire qualifies as a printed publication), 20 ("The language of the claims and of Sourcefire provide substantial evidence for the Board's finding that Sourcefire teaches the operator in the '552 patent's claims."). The Board also found all claims of the '713 patent unpatentable over Sourcefire, which was also affirmed. EX-1068 29-30; EX-1069 17-21; EX-1003 ¶¶58-59.

Cisco also petitioned for IPR of the related '193 patent. The Board denied institution of that IPR because the Petitioner failed to explain sufficiently the meaning of the term "operator" as used in those claims, which resulted in Cisco failing to establish a reasonable likelihood of prevailing. EX-1070 5-6, 8-11; EX-1003 ¶¶60-61.

After being denied IPR institution, Cisco sought reexamination of the '193 patent in view of Sourcefire. EX-1085 616-627; EX-1003 ¶62. A non-final Office Action initially rejected all claims as obvious over Sourcefire in view of the knowledge of a POSA. EX-1085 157-179. The Patent Owner then argued that the "operator" as claimed in the '913 patent was distinct from the operator in the '552 patent, and that the purpose of the '913 patent was to block exfiltrations while Sourcefire supposedly does not provide a motivation to a POSA to write rules targeting exfiltrations. EX-1085 11-13; EX-1003 ¶63. The Office then issued a reexamination certificate confirming the patentability of all claims of the '193 patent over Sourcefire. EX-1085 1. The Examiner did not consider whether the claims would have been obvious in view of Sourcefire in combination with published Snort rules, compatible with Sourcefire, that targeted exfiltrations, such as those in Emerging Threats. EX-1003 ¶¶63-64.

E. Procedural History of '343 Patent

Petitioner previously petitioned for IPR of the '343 patent, challenging claims 1-6, 8-13, and 15-19 as being unpatentable in view of Sourcefire; and claims 7, 14, and 20 in view of Sourcefire and published application 2009/0103524 ("Mantripragada"). EX-1077 1. In its preliminary response, Patent Owner argued that the Patent Office had considered Sourcefire already and determined that the '343 patent, like the '193 patent, was non-obvious over Sourcefire. EX-1084 at 25-49;

EX-1003 ¶¶76-79. Patent Owner specifically argued that the '343 patent was not obvious over Sourcefire because, according to Patent Owner, Sourcefire does not disclose a technique for preventing exfiltrations and a POSA would not be motivated to write such rules. EX-1084 40-49; EX-1003 ¶79.

Under *Advanced Bionics, LLC v. Med-El Elektromedizinische Gerate GMBH*, 2020 Pat. App. LEXIS 13184, (PTAB February 13, 2020). (precedential) ("*Advanced Bionics*"), the Board determined that Sourcefire was considered during prosecution of the '343 patent and Mantripragada was cumulative of a considered reference. EX-1077 14-15. The Board further found under *Becton, Dickinson and Co. v. B. Braun Melsungen AG*, IPR2017-01586, Paper 8 (PTAB Dec. 15, 2017) (precedential in relevant part) ("*Becton, Dickinson*") that Patent Owner's argument that the packet transformation function is configured "to prevent an exfiltration operation," which resulted in allowance of the '343 patent, overlapped with the argument that Patent Owner made against the petition. The Board determined that the petition failed to show that a two-stage filtering process found obvious in the '552 IPR was a basis for the Examiner's allowance of the '343 patent. The Board thus found that the same or substantially the same prior art, and the same or substantially the same arguments, previously were presented to the Office. EX-1077 15-16.

Under *Advanced Bionics* part two, the Board determined that Petitioner had not identified sufficiently any material error on the part of the Examiner. The Board also determined that the Examiner allowed the claims because of the exfiltration operation, and not the "two-stage filtering process," as argued by Petitioner. EX-1077 at 16-18; EX-1003 ¶80. Accordingly, the Board exercised discretion not to institute under § 325(d). *See* § V.D.2 regarding § 325(d) and the instant Petition.

F. Person of Ordinary Skill In The Art

Dr. Staniford's declaration submitted with this petition, the '343 Patent, and the art cited herein demonstrate that a person of ordinary skill in the art ("POSA") would have had a bachelor's degree in computer science, computer engineering, or an equivalent, and four years of industry experience. EX-1003 ¶23. Additionally, a POSA would have had a working knowledge of packet-switched networking, firewalls, security policies, communication protocols and layers, and the use of customized rules to address cyber-attacks. *Id.* ¶25. Additional graduate education could substitute for professional experience, and vice versa. *Id.* ¶¶23-27. Dr. Staniford is at least a POSA, and has employed and supervised POSAs. *Id.* ¶27; EX-1035.

V. OVERVIEW OF CHALLENGE AND RELIEF REQUESTED

A. Grounds for Standing (§ 42.104(a))

Petitioner certifies that the '343 patent is available for *inter partes* review and that Petitioner is not barred or estopped from requesting an *inter partes* review

on the grounds identified herein. Although Petitioner was served more than a year ago with a complaint alleging infringement of this patent, the one-year statutory bar under 35 U.S.C. § 315(b) expressly does not apply here because Petitioner accompanies this Petition with a timely Motion for Joinder to join IPR2023-00446, which was instituted on 8/7/2023.

B. Identification of Challenge (§§ 42.104(b)(1) & (2))

Petitioner asserts that the Challenged Claims are unpatentable on the following grounds:

Ground	Claim(s)	35 U.S.C. §	References
1	1-20	103	Sourcefire[3], Emerging Threats[4]

Petitioner requests IPR of claims 1-20 of the '343 patent as obvious under § 103 in view of Sourcefire and Emerging Threats. This petition is supported by the Declaration of Dr. Stuart Staniford. EX-1003 ¶¶1-301.

The Board determined in the '552 IPR that "Sourcefire qualifies as a prior art publication under § 102(b)." EX-1067 at 31. The Federal Circuit affirmed, rendering those findings binding. EX-1069 at 17 ("we agree with the Board that Sourcefire was publicly accessible and therefore qualifies as a printed publication"). Sourcefire

[3] Sourcefire 3D System User Guide Version 4.10 ("Sourcefire"), EX-1004.

[4] Emerging Threats ("Emerging Threats"), EX-1020.

therefore qualifies as prior art to the '343 patent, which claims priority to the '552 patent. *Nestlé USA, Inc. v. Steuben Foods, Inc.*, 884 F.3d 1350, 1351 (Fed. Cir. 2018) (collateral estoppel protects a party from having to litigate issues that have been fully and fairly tried and resolved against a party-opponent). Petitioner submits herewith evidence submitted in IPR2018-01760 to establish Sourcefire's qualification as prior art. EX-1003 ¶¶9, 168-169; EX-1005 ¶¶18-19; EX-1033; EX-1045; EX-1046 ¶9.

Emerging Threats also qualifies as a "printed publication" under § 102(b) as it has been "disseminated or otherwise made available to the extent that persons interested and ordinarily skilled in the subject matter or art exercising reasonable diligence, can locate it." *SRI Int'l, Inc. v. Internet Sec. Sys., Inc.*, 511 F.3d 1186, 1194 (Fed. Cir. 2008) (quoting *Bruckelmyer v. Ground Heaters, Inc.*, 445 F.3d 1374, 1378 (Fed. Cir. 2006)); EX-1003 ¶9. The Internet Archive's Wayback Machine shows the public accessibility of Emerging Threats on December 2, 2010 by preserving Emerging Threats as it was "made available to the public" at that date. *Valve Corp. v. Ironburg Inventions Ltd.*, 8 F.4th 1364, 1374 (Fed. Cir. 2021) (recognizing that "[E]xaminers use 'The Wayback [M]achine" to determine when a "reference was first made available to the public"), *Veeam Software Corp. v. Symantec Corp.*, IPR2013-00142, Paper 11, *9-10 (Aug. 7, 2013) (considering webpages from the Wayback Machine even without declaration from Archive.org

when granting institution); EX-1049 ¶20; EX-1030 1 (affidavit of a person with personal knowledge attesting that the Wayback Machine crawler operates to create an copy of a website as it existed at a particular point in time); *Pohl v. Mh Sub I, LLC*, 332 F.R.D. 713, 717-718 (N.D. Fla. 2019) (discussing evidence sufficient to support a finding that WayBack machine evidence is authentic).

As a Wayback Machine reference, Emerging Threats' "accuracy cannot reasonably be questioned." *Valve Corp.*, 8 F.4th at 1374; *see also Pohl*, 332 F.R.D. at 716 ("follow[ing] the lead of the overwhelming number of courts that have decided the issue and tak[ing] judicial notice of the contents of WayBack Machine evidence because they 'can be accurately and readily determined from sources whose accuracy cannot reasonably be questioned.'")(citation omitted). Dr. Staniford also explained that, based on his experience, the Emerging Threats rules appear to be true copies of rules that existed in 2010. EX-1003 ¶¶204-206. Emerging Threats was a well-known independent clearinghouse of Snort rules that were available to a POSA. EX-1003 ¶¶202-203, 206; EX-1018; EX-1050, EX-1051; EX-1082 (Patent Owner's 2014 press release referring to Emerging Threats as "a world-leading provider of commercial and open source threat intelligence" and the "de facto standard in network-based malware threat detection").

C. Claim Construction (37 C.F.R. § 42.104(b)(3))

The Board construes claims in IPRs using the same claim construction standard as in civil litigation. § 42.100(b). Construction, however, is only required when necessary to resolve a controversy. *Nidec Motor Corp. v. Zhongshan Broad Ocean Motor Co.*, 868 F.3d 1013, 1017 (Fed. Cir. 2017) (*citing Vivid Techs., Inc. v. Am. Sci. & Eng'g, Inc.*, 200 F.3d 795, 803 (Fed. Cir. 1999) ("only those terms need be construed that are in controversy, and only to the extent necessary to resolve the controversy"). Petitioner submits that no express claim construction is required for this proceeding and claims should be given their plain and ordinary meaning.[5]

The term "operator" requires discussion, however, given its history in prior challenges to related patents. The Federal Circuit affirmed the Board's finding that Sourcefire disclosed applying the operator claimed in the two-stage packet filtering of the related '552 patent. EX-1067 17, 50-52; EX-1069 21; EX-1003 ¶¶146-148. However, the Board denied institution of the '193 IPR finding that the petition lacked a sufficiently clear explanation of the term "operator," and in the '193 *ex parte* reexamination Patent Owner argued that the '193 claimed operator was

[5] Petitioner does not concede the scope or construction of any claim term or indefiniteness argument in any other proceeding.

different than the '552 claimed operator. EX-1070 6-12; EX-1085 127-128; EX-1003 ¶¶149-150.

In computer science generally, an "operator" is a function that takes some inputs and produces an output. EX-1003 ¶140 (citing dictionary definitions EX-1036, EX-1037, EX-1040, EX-1041). Computer languages also allow "operator overloading," in which a programmer can create custom definitions of operators over their own data types, and such operators can do anything that can be expressed in the computer language. EX-1003 ¶141. The use of "operator" in the '343 patent is consistent with this general use: the operator is a function that may take packets as inputs and ultimately produces a decision as to what to do with the packets. EX-1003 ¶¶142-143. As shown in figure 3 (below), an operator may perform a function (a "packet transformation function" such as ALLOW or BLOCK) if a first criteria specified by a rule is met (packet-header-field-value information corresponds to five-tuple criteria). Some operators (e.g., HTTP-EXFIL) may also specify application header field criteria and apply a packet transformation function if the application header field values of packets match the application header field criteria.

218

Five-tuple

Rule #	IP Protocol	Source IP Address	Source Port	Destination IP Address	Destination Port	Operator
1 (302)	TCP	140.210.*	*	140.212.*	22	ALLOW
2 (304)	TCP	140.210.*	*	140.212.*	25	ALLOW
3 (306)	TCP	140.210.*	*	140.212.*	110	ALLOW
4 (308)	TCP	140.210.*	*	140.212.*	143	ALLOW
5 (310)	TCP	140.210.*	*	140.212.*	443	REQUIRE-TLS-1.1-1.2
6 (312)	TCP	140.210.*	*	214.*	80	HTTP-EXFIL
7 (314)	*	*	*	*	*	BLOCK

The '343 patent provides examples of operators in pseudocode, including the HTTP-EXFIL operator as shown below. EX-1001 7:41-8:6; EX-1003 ¶¶143-144.

```
Operator HTTP-EXFIL(ip-packet):
Inspect app-pkt(ip-packet) match GET return ALLOW;
Inspect app-pkt(ip-packet) match POST return BLOCK;
Inspect app-pkt(ip-packet) match PUT return BLOCK;
Inspect app-pkt(ip-packet) match DELETE return
  BLOCK;
Inspect app-pkt(ip-packet) match CONNECT return
  BLOCK;
Return BLOCK;
End Operator HTTP-EXFIL;
```

EX-1001 7:53-62. This example shows an operator that performs an ALLOW packet-transformation function on packets if their application header field values contain a GET HTTP method, but a BLOCK packet-transformation function if their HTTP application header field values contain a POST, PUT, DELETE, or CONNECT HTTP method. EX-1001 7:4-13. This also shows that the '343 patent's operators themselves may call other functions, since they are calling packet

transformation functions, and that the system's source code implements the operator. EX-1003 ¶¶144, 151. The operator functionality is triggered when the first-stage criteria is met (e.g., the 5-tuple criteria shown in figure 3 above). EX-1003 ¶¶152-153. The operator may then perform a packet-transformation function (such as ALLOW or DROP as shown in rules 1-4), or may inspect information in the packets to determine application header field values and perform a packet-transformation function according to those values (such as for HTTP-EXFIL shown in rule 5, ALLOW packets containing GET HTTP methods, but BLOCK packets containing POST HTTP methods). EX-1003 ¶154.

As shown below, while they do not use the same language, Sourcefire discloses such operators that include criteria and perform a packet transformation function, and Emerging Threats discloses such operators that perform the same exfiltration protection disclosed and claimed in the '343 patent. EX-1003 ¶150.

D. The Board Should Not Exercise Discretion to Deny Institution

1. Stipulation

Petitioner stipulates that if the Board institutes *inter partes* review, Petitioner will not pursue in the Parallel Litigation any ground that was raised in this IPR petition. This effectively accelerates the estoppel that would apply after a final written decision issues in this proceeding. Moreover, the parties in the Parallel Litigation have jointly requested a stay of the Parallel Litigation with respect to the

’343 patent. *See Centripetal Networks, Inc. v. Palo Alto Networks, Inc.*, Case No. 2:21-cv-00137 (E.D. Va.), Dkt. No. 399 (Joint Motion for Stay of Asserted U.S. Patent Nos. 10,567,343 and 10,785,266 Pending Resolution of *Inter Partes* Review) and Dkt. No. 400 (Memorandum In Support) (filed Aug. 24, 2023). Accordingly, the Parallel Litigation does not provide a basis for discretionary denial.

2. The Board Should Not Exercise 325(d) Discretion

Where, as here, a petitioner files a “copycat” petition in conjunction with a timely motion for joinder and agrees to assume a “passive understudy role,” the *General Plastic* factors are “effectively neutraliz[ed].” *Apple Inc. v. Uniloc 2017 LLC*, IPR2018-00580, Paper 13 at 10-11 (Aug. 21, 2018). While the present copycat Petition is Petitioner’s second petition challenging the ’343 patent, this case is distinguishable from *Apple Inc. v. Uniloc 2017 LLC*, IPR2020-00854, Paper 9 at 4 (Oct. 28, 2020) (precedential) (“Uniloc II”). Unlike the petitioner in *Uniloc II*, who “failed to set forth facts or offer an explanation concerning its knowledge, at the time it filed the first petition, of the prior art asserted in the [copycat petition],” *id*. at 9, here, at the time of filing of the first petition (IPR2021-01154, Paper 1, filed 7/6/2021), counsel for Petitioner in all of the actions involving Patent Owner did not identify or consider the prior art reference (Emerging Threats) used in the Keysight IPR and in the sole ground of the present Petition (EX-1020) that specifically teaches the exfiltration prevention operation that the Examiner considered pertinent to

allowance of the claims (EX-1077 15), despite extensive prior art search (Factor 2). *See Google LLC v. Express Mobile, Inc.*, IPR2022-00789, Paper 15 at 7-8 (Oct. 5, 2022) (finding Petitioner's representation that it did not earlier know of the prior art sufficient and distinguishable from *Uniloc II*). Petitioner did not become aware of Emerging Threats until it reviewed the Keysight IPR petition, which was filed on 1/6/2023. Accordingly, there was no delay between when Petitioner learned of Emerging Threats and when it filed this second petition, which could not have been filed before the Keysight IPR Institution Decision (Factors 4, 5). Regarding Factor 3, "there is no reason to conclude that Petitioner used the filings or decision in the [first IPR it filed] to obtain an unfair advantage in this Petition, which was essentially prepared by someone else." *Id.* at 8. Finally, the Board should "decline to speculate on whether [Keysight] will settle with Patent Owner," especially given that "the Board has already allocated resources to complete the [Keysight] IPR [Proceeding]." *Advanced Micro Devices, Inc. v. Monterey Research, LLC*, IPR2021-00776, Paper 13 at 17 (Oct. 13, 2021) (factor 6 does not weigh in favor of denying institution). Accordingly, unlike *Uniloc II*, but like *Google*, the *General Plastic* factors favor institution and joinder here.

VI. SOURCEFIRE IN VIEW OF EMERGING THREATS RENDERS CLAIMS 1-20 OBVIOUS

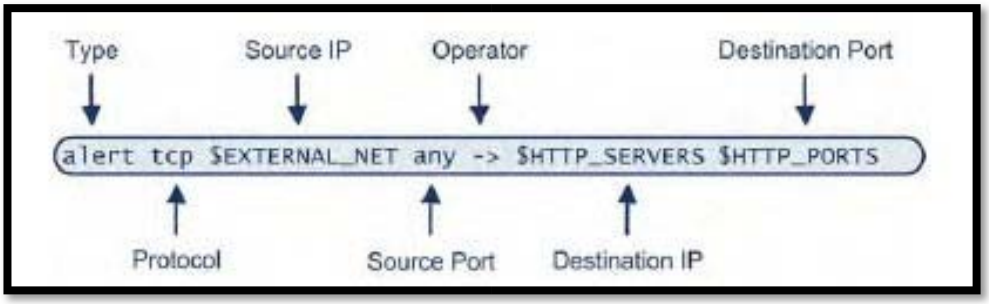
A. Overview of Sourcefire

Sourcefire is a user guide for the Sourcefire 3D System, which was a commercially successful Intrusion Protection System (IPS). EX-1003 ¶167; *see also* EX-1003 ¶¶101-116 (background of network security systems and network intrusion detection and protection systems). Sourcefire describes identifying changing assets and vulnerabilities on a network and defending the network in real-time. EX-1004 32; EX-1003 ¶170. Sourcefire describes packet-filtering devices referred to as "3D Sensors" that could be deployed in a network and could include an IPS, which provided "best-in-class intrusion detection and prevention by harnessing the power of the industry-standard Snort® rules-based detection engine." EX-1004 33-34; EX-1003 ¶171. 3D Sensors with IPS examined the packets that traversed the network and reported and/or took actions based on detected activity using a rules engine (as described below), which implemented rules to look for the broad range of malicious behaviors that attackers had developed. EX-1004 32-35, 254-256; EX-1003 ¶172. The rules engine implemented "intrusion" rules to determine whether the headers and/or payloads of received packets triggered one or more of such rules. EX-1004 256-260, 513, 428-430, 760-770, 2084, 2089; EX-1003 ¶¶173-174. While Sourcefire uses the term "intrusion rules," the rules served a number of purposes, including detecting and preventing exfiltration, as discussed below. EX-1003 ¶175.

The 3D Sensor with IPS could be deployed inline to allow or drop packets that the 3D Sensor detected as being harmful. EX-1004 262-265, 276-277, 435-438, 761; EX-1003 ¶¶162, 173.

Sourcefire's "intrusion rules" are a set of "keywords and arguments on a 3D Sensor" that detects attempts to "exploit vulnerabilities on [a] network by analyzing network traffic to check if it matches the criteria in the rule." EX-1004 761. A rule consists of two logical sections: a rule header and rule options. EX-1004 358, 488-490, 761-766; EX-1003 ¶175.

The rule header consists of parameters including 5-tuple criteria values (protocol, source and destination IP addresses, and source and destination ports), the rule's action or type (e.g., alert and allow, drop, or ignore and allow), and direction indicating the flow of traffic. EX-1003 ¶175. For example, the following figure and table illustrate the parts of a rule header and how they could be used by IPS:



EX-1004 764; EX-1003 ¶177. Each header component corresponds to a "value" as noted in the below chart:

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.

EX-1004 762-764; EX-1003 ¶177.

The rule options part consists of event messages, keywords and their parameters and arguments, patterns that a packet's payload must match to trigger the rule (which included application header field values for some rules, as discussed below), and specifications of which parts of the packet the rules engine should inspect. EX-1004 358, 488-490, 761-766; EX-1003 ¶175. An example intrusion rule in Sourcefire illustrates these rule parts:

Rule Header

```
alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS $HTTP_PORTS
```

Rule Keywords and Arguments

```
(msg:"WEB-IIS newdsn.exe access";
flow:to_server,established; uricontent:"/scripts/
tools/newdsn.exe"; nocase; metadata:service http;
reference:bugtraq,1818; reference:cve,1999-0191;
reference:nessus,10360; classtype:web-application-
activity; sid:1024; rev:10; )
```

EX-1004 762-763; EX-1003 ¶176. This rule alerted the user based on TCP packets from any external network source IP address and port, destination IP addresses and ports matching known values for HTTP servers, and including a URI (/scripts/tools/newdsn.exe) used in a popular attack which matched the corresponding content keyword. *Id.* The rule also identified its message, unique identifying number ("Snort ID" or SID:1024) and the various sources (identified by the "reference" text) to external reports about the exploit the rule detected. *Id.*

Sourcefire's rules could be applied packet-by-packet and the type of actions that could be triggered by the rules include *pass* (and ignore), *alert* (and pass) or *drop* (block). EX-1004 256-260, 358-359, 762, 764-769; EX-1003 ¶¶178, 190-191, 200.

The Sourcefire IPS was designed such that packets would trigger a rule when data within a packet satisfied one or more specified criteria that corresponded to packet header field values (e.g., source or destination IP address, source or

destination port, etc.). EX-1004 762-769; EX-1003 ¶¶179, 193-198. The IPS may apply the action specified by the rule (*e.g.* pass, drop, alert (and pass)) to the triggering packet upon the packet triggering the rule. EX-1003 ¶178. Based on keywords in the rules, Sourcefire filtered packets as they arrived at application layer preprocessors and at the rules engine based on various application layer data in such packets. EX-1004 334, 373, 428-429, 435-439, 491, 538-542, 546-560, 597-601, 697-701, 765, 785-786, 827-828, 874, 2032-2043, 2092; EX-1003 ¶180. Like the operators in the '343 claims, Sourcefire describes that rules could be configured to look for application header field values and apply actions based on whether packets match those values. EX-1004 598, 697-701; EX-1003 ¶¶181-185.

As an example, Sourcefire discloses that HTTP content keywords could be used to search for matched application layer content in packets, such as known HTTP methods "GET, PUT, CONNECT, and so on" and allow or drop packets based on the results. EX-1004 785-786; EX-1003 ¶183. Thus, when application layer information matched the specified criteria (e.g., a GET method), the IPS applied the "packet transformation function" specified by the rule (e.g., allow or drop). EX-1004 435-439, 295-297, 765; EX-1003 ¶183.

Sourcefire also explains that users may import rules from local rule files, which are ASCII text files from a local machine. EX-1004 1857-1859; EX-1003

¶163. Specifically, Sourcefire discloses the following interface for importing a rule file:

2. Click **Import Rules**.

   The Import SEU page appears.

One-Time SEU/Rules Import

Source	○ SEU or text rule file to upload and install [ ] Browse... ○ Download new SEU from the support site
Policy Reapply	☐ Reapply intrusion policies after the SEU import completes

Import

Recurring SEU Imports

The scheduled SEU import feature is not enabled

Enable Recurring SEU Imports	☐

Save Cancel

TIP! You can also click **Import SEU** on the SEU page, which you access by selecting **Policy & Response > IPS > SEU**.

3. Select **SEU or text rule file to upload and install** and click **Browse** to navigate to the rule file. Note that all rules uploaded in this manner are saved in the local rule category.
4. Click **Import**.

   The rule file is imported. Make sure you enable the appropriate rules in your intrusion policies. The rules are not activated until the next time you apply the affected policies.

IMPORTANT! 3D Sensors do **not** use the new rule set for inspection until after you apply their intrusion policies. See Applying an Intrusion Policy on page 347 for procedures.

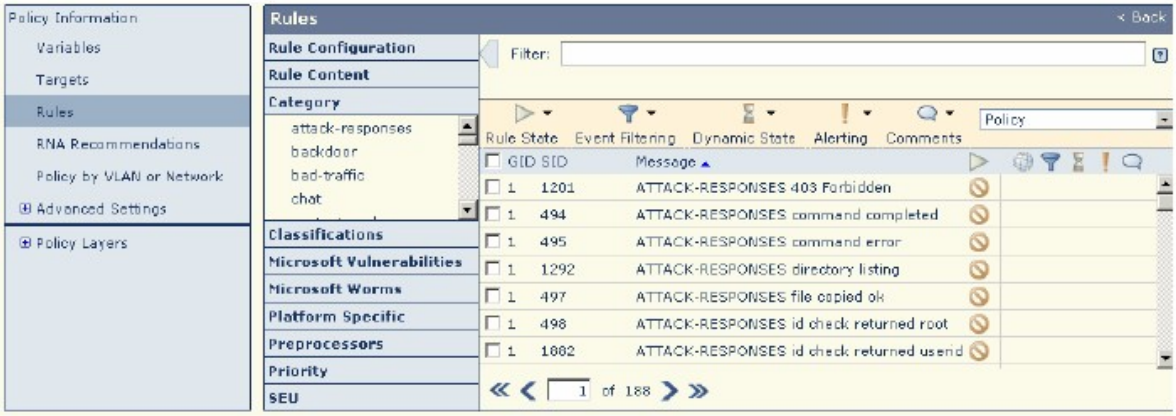
EX-1004 1859. When rules are imported, the Sourcefire IPS sets the rules to a disabled rule state, and requires a user to set the rule state before they can be used:

- IPS always sets local rules that you import to the disabled rule state; you must manually set the state of local rules before you can use them in your intrusion policy. See Setting Rule States on page 435 for more information.

EX-1004 1858; EX-1003 ¶163. As shown above, Sourcefire expressly explains that once imported, a user would need to select whether rules should provide alerts, or

block and provide alerts, and references the "Setting Rule States" portion of Sourcefire.

In "Setting Rule States," Sourcefire teaches how a user of a Sourcefire IPS can change rules to alert (i.e., generate events) or drop (i.e., drop packets that trigger the rules). EX-1004 435-438; EX-1003 ¶162. The Sourcefire Rules page, shown below, lists the rules and enables a user to select one or more rules and then select the "Rule State" button to select whether the rule or rules are set to generate events, drop and generate events, or are disabled.



EX-1004 437. The following excerpt explains how to change the rule state:

5. Select the rule or rules where you want to set the rule state. You have the following options:
   - To select a specific rule, select the check box next to the rule.
   - To select all the rules in the current list, select the check box at the top of the column.
6. You have the following options:
   - To generate events when traffic matches the selected rules, select **Rule State > Generate Events**.
   - To generate events and drop the traffic in inline deployments when traffic matches the selected rules, select **Rule State > Drop and Generate Events**.

     Note that in passive deployments, including when an inline interface is in tap mode, setting the rule state to **Drop and Generate Events** causes rules to generate events without dropping packets.
   - To not inspect traffic matching the selected rules, select **Rule State > Disable**.

IMPORTANT! Sourcefire **strongly** recommends that you **do not** enable all the intrusion rules in an intrusion policy. The performance of your sensor is likely to degrade if all the rules are enabled. Instead, tune your rule set to match your network environment as closely as possible.

EX-1004 438. As illustrated above, a Sourcefire user would typically tune their rule set to match the specific network environment, including selecting rules to activate and selecting whether to set each rule to block or alert. EX-1003 ¶¶162, 268.

Sourcefire therefore shows that it was customary for users to import custom rules, and then to select whether the rules would be set to alert or to block traffic. EX-1003 ¶163. As explained below, Emerging Threats was a well-known source of custom rules, including exfiltration rules, designed to be used with Snort-based intrusion detection and prevention systems, like Sourcefire. EX-1003 ¶¶164-166. A POSA would have understood the obvious desirable effect of setting Emerging

Threats rules imported into Sourcefire to drop traffic associated with detected exfiltration operations. EX-1003 ¶289.

B. Overview of Emerging Threats

The core intrusion detection and prevention engine in Sourcefire, Snort, began as an open source project in the late 1990s. EX-1004 33; EX-1018 1-5; EX-1003 ¶202. The rulesets were open-source and accepted community contributions. EX-1018 5-6; EX-1003 ¶202. Rules were made available by independent rule providers, such as the Bleeding Snort project that launched in 2004 as a clearinghouse for Snort rules to detect a variety of problems. EX-1050 1-2; EX-1051; EX-1003 ¶202. The project was renamed Emerging Threats and continues today. EX-1053; EX-1054; EX-1078; EX-1030; EX-1003 ¶¶202-206.

As Dr. Staniford explains, he used Snort rules extensively before 2010 and was aware of the Bleeding Snort project, which later became Emerging Threats. EX-1003 ¶¶121, 164, 206. Patent Owner's own 2014 press release describes Emerging Threats, founded in 2003, as a world-leading provider of threat intelligence rulesets:

About Emerging Threats

Emerging Threats is a world-leading provider of commercial and open source threat intelligence. Founded in 2003 as a cyber security research community, Emerging Threats has become the de facto standard in network-based malware threat detection. The company's ETOpen Ruleset, ETPro® Ruleset, and IQRisk® Suite of threat intelligence are platform agnostic for easy integration with Suricata, SNORT®, and other network intrusion protection and detection systems. With ETPro Ruleset, organizations can achieve the highest standards of malicious threat detection with world-class support and research for extended vulnerability coverage. ETPro Ruleset is ideal for enterprises, government agencies, financial institutions, SMBs, higher education, and service providers.

For more information, please visit http://www.emergingthreats.net

EX-1082. Patent Owner licensed certain Emerging Threats rulesets and described Emerging Threats as the "de facto standard in network-based malware threat detection." EX-1082.[6] Therefore, a POSA would have known about the rules published in Emerging Threats by 2010, and it qualifies as a printed publication under § 102(b).

Emerging Threats (EX-1020) is a text file of rules designed to be imported into and used with Snort systems, like Sourcefire. As explained in the prior section, Sourcefire instructs how to import text files of rules into the product, which provides a user-interface for the section and import of such files. *See* § IV.A; EX-1004 1857-1859; EX-1003 ¶207. Sourcefire sets the rules to be disabled upon import, and instructs users to set the rule state (e.g., alert or drop) to tune the ruleset to match the

[6] Patent Owner did not disclose any Emerging Threats rules during prosecution of the '343 patent. EX-1001 code 56 (References Cited); EX-1003 ¶¶209-210.

network environment. *See* § IV.A; EX-1004 435-438; EX-1003 ¶207. Patent Owner's press release touted that the Emerging Threat rulesets are "platform agnostic" for easy integration with SNORT® and other network intrusion protection and detection systems. EX-1082. It would therefore have been obvious for a POSA to use the functionality provided by Sourcefire to import rules included in Emerging Threats. EX-1003 ¶206. A POSA would expect success importing and activating Emerging Threats rules within Sourcefire, since Sourcefire was designed for importation and activation of such rules, and the Emerging Threats rules were designed for use with Snort-based systems, like Sourcefire. EX-1003 ¶¶208, 232-233.

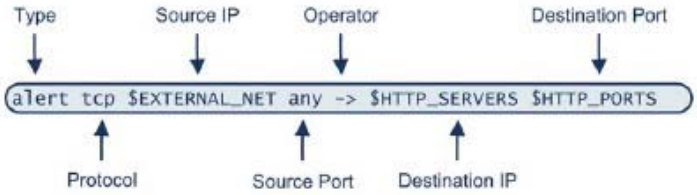
Emerging Threats further contains rules targeted at detecting exfiltration attempts. EX-1003 ¶211; *see also* EX-1003 ¶¶117-127 (discussing history of network security devices detecting and preventing data exfiltration). An example of exfiltration would be malware attempting to exfiltrate corporate trade secrets. EX-1081 4; EX-1003 ¶212. An Emerging Threats rule designed to detect such an exfiltration is SID:2002526:[7]

```
#US Top Secret Not Releasable to Foreign Nationals
#
#alert tcp $HOME_NET any -> $EXTERNAL_NET
$HTTP_PORTS (msg:"ET POLICY HTTP - US Top Secret
NOFORN"; flow:to_server,established;
```

[7] SID stands for the "Snort ID" of a rule. EX-1004 at 311.

```
pcre:"/(TOP\sSECRET|TS)//[\s\w,/-]*NOFORN[\s\w,/-
]*(?=//(25)?X[1-9])/ism"; classtype:policy-
violation;
reference:url,doc.emergingthreats.net/bin/view/Main
/2002526;
reference:url,www.emergingthreats.net/cgi-
bin/cvsweb.cgi/sigs/POLICY/POLICY_Classified_Inform
ation; sid:2002526; rev:4;)
```

EX-1020 87 (SID:2002526).[8] A POSA familiar with Snort rules would understand the purpose of this rule. EX-1004 Chapter 20 (starting at 761); EX-1003 ¶213. The following Sourcefire diagram illustrates the rule header values:



The Rule Header Values table describes each part of the rule header shown above.

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.

[8] Specific rules can be located within EX-1020 by searching for their unique SID.

EX-1004 764. These show that in the rule above, "tcp" indicates that Snort is to examine TCP traffic, the operator "->" indicates the direction of traffic to be examined, in this case external traffic such that only packets in one direction of the TCP connection were examined. EX-1003 ¶213; EX-1004 764-769. The keywords "flow:to_server,established" indicate that the direction of traffic to be inspected was from a client to a server. EX-1003 ¶213; EX-1004 820-822. The packets to be examined are from a client in "$HOME_NET any" indicating the client had an IP address on the local (protected) network, and any client port, and that the server was located somewhere in "$EXTERNAL_NET $HTTP_PORTS," that is on an external network with a server port characteristic of HTTP. EX-1003 ¶213, EX-1004 214-216, 372-376. Thus, the rule examines traffic in requests from HTTP clients (such as web browsers) on the protected network to HTTP servers on the rest of the Internet, like the HTTP exfiltration methods disclosed in the '343 patent. EX-1003 ¶214; EX-1001 1:31-40, 7:4-23; *see also* EX-1003 ¶¶88-100.

The rule then uses Snort's pcre keyword to specify a regular expression to search the HTTP request packets to search for the strings "TOP SECRET" or "TS" along with "NOFORN" separated by appropriate spaces and punctuation. EX-1003 ¶215; EX-1004 800-809. The rule thus searches for markings in HTTP content flowing toward external servers indicating that the information was classified as top secret and not for distribution to foreign nationals. EX-1021 3, 12; EX-1003 ¶215.

A POSA would understand this is a classic exfiltration rule seeking classified information in the headers or bodies of outbound HTTP requests. EX-1003 ¶216.

As another example, the Emerging Threats rule SID:2002034 was configured to detect and block unauthorized transmission of the password file from Unix/Linux systems out of an organization via a web request (HTTP). EX-1020 8; EX-1003 ¶217. The rule states "msg:"ET ATTACK_RESPONSE Possible /etc/passwd via HTTP (linux style)." A POSA would recognize that "/etc/passwd" is the text file where password information is stored on Unix and Linux systems. EX-1028 1; EX-1003 ¶217. For the first-stage criteria, the rule indicates that it examines TCP packets from a server on the local network on HTTP ports to an external server with any client port. EX-1003 ¶¶218-219; EX-1004 214-216, 372-376, 764-769, 8:20-822. For the second-stage criteria, the rule searches for the beginning of the first line of a Unix/Linux password file. EX-1003 ¶¶219-220. A POSA would understand a password file leaving a network in an HTTP response to be an exfiltration operation. EX-1003 ¶220. Dr. Staniford also discusses an example of an Emerging Threats two-stage rule for detecting and blocking exfiltration operations based on first a match with 5-tuple criteria and second a match with a regular expression searching for application layer information containing social security numbers with the familiar "-XX-XXXX" ending pattern. EX-1020 109 (SID:2001328); EX-1003 ¶¶221-224; EX-1034.

As yet another example two-stage exfiltration operation detection/prevention rule, Emerging Threats includes the following rule designed to detect/prevent transmissions of encrypted data on an usual port via SSL v3 (an insecure and very out-of-date version of SSL in 2013). EX-1003 ¶¶225-228, 263-267; *see also* EX-1003 ¶¶128-136 (discussing history of inspecting and applying packet-filtering rules to encrypted and unencrypted SSL/TLS protocol data); EX-1006; EX-1094; EX-1038 4-5, 79.

```
#Application data stream
#
#alert tcp $HOME_NET any -> $EXTERNAL_NET
1024:65535 (msg:"ET POLICY TLS/SSL Encrypted
Application Data on Unusual Port SSLv3";
flowbits:isset,BS.SSL.Established;
flow:established,to_server; content:"|17 03 00|";
depth:4; threshold:type limit, count 1, seconds
120, track by_src; classtype:unusual-client-port-
connection;
reference:url,doc.emergingthreats.net/2003021;
reference:url,www.emergingthreats.net/cgi-
bin/cvsweb.cgi/sigs/POLICY/POLICY_SSL_TLS_on_High_P
ort; sid:2003021; rev:8;)
```

EX-1020 109 (SID:2003021). The rule's message suggests its purpose: looking for transmission of data on an unusual port using an outdated encryption protocol. EX-1003 ¶¶128, 225, 264.

The first-stage criteria of the rule indicates that Snort was to inspect the contents of packets in TCP connections in the direction from the client to a server.

EX-1004 764, 766, 49, 820-822; EX-1003 ¶226. The packets to be examined are from a client in "$HOME_NET any," indicating the client had an IP address on the local (protected) network, and any client port, and to a server located somewhere in "$EXTERNAL_NET 1024:65535," that is on an external network with a server port that is not in the range used for well-known system protocols. EX-1003 ¶226; EX-1004 214-216, 372-376; EX-1027.

The second-stage criteria checks whether an ongoing SSL connection was established, and whether an application header field value indicates the transmission of application data via SSL v3.0. EX-1003 ¶¶227-228; EX-1004 851-855; EX-1038 16-17. Specifically, a portion of the operator "content:"|17 03 00|"; depth:4;" specifies criteria the operator is searching for as an application header field value for a particular SSL/TLS record header. EX-1003 ¶227. The structure of that header is as follows:

```
        record type (1 byte)
       /
      /    version (1 byte major, 1 byte minor)
     /    /
    /    /         length (2 bytes)
   /    /         /
+----+----+----+----+----+
|    |    |    |    |    |
|    |    |    |    |    | TLS Record header
+----+----+----+----+----+
```

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-1003 ¶227; EX-1038 16-17. The Snort content operator is looking for hex bytes |17| (corresponding to APPLICATION_DATA) followed by 03 and 00 (corresponding to SSL 3.0 major and minor version numbers). EX-1003 ¶227. The "depth 4" indicates that the search cannot go more than 4 bytes into the packet payload (i.e. it is looking for the header at the beginning of the record). EX-1003 ¶227; EX-1004 783. Accordingly, this rule detects, and could be tuned to block, attempts to exfiltrate data from a network using an out-of-date encryption protocol and an unusual server port. EX-1003 ¶228.

A POSA would recognize this Emerging Threats rule (SID:2003021) as similar to the REQUIRE-TLS operator of the '343 patent. EX-1001 7:41-49, 8:7-38. REQUIRE-TLS inspects packets in an HTTPS encrypted communication

session, allows packets to pass if the application packet header field value indicates that the packets are encrypted with TLS versions 1.1 or 1.2, and blocks packets if the application packet header field value indicates that the packets are encrypted with TLS version 1.0, or if no match with TLS versions 1.1 or 1.2 is found. In other words, the REQUIRE-TLS-1.1-1.2 operator in the '343 patent and SID:2003021 in Emerging Threats both detect an SSL/TLS version in application packet header field values to detect exfiltration attempts using out-of-date encryption protocols. A POSA would therefore understand both of these to be rules useful for detecting and preventing exfiltration operations. EX-1003 ¶¶229-230.

C. Collateral Estoppel Applies To Issues Decided Against Patent Owner In The '552 IPR

Collateral estoppel applies in an IPR proceeding to "protect[] a party from having to litigate issues that have been fully and fairly tried in a previous action and adversely resolved against a party-opponent." *Nestlé*, 884 F.3d at 1351 ("Collateral estoppel protects a party from having to litigate issues that have been fully and fairly tried in a previous action and adversely resolved against a party-opponent.") (citations omitted); *Soverain Software LLC v. Victoria's Secret Direct Brand Mgmt., LLC*, 778 F.3d 1311, 1319 (Fed. Cir. 2015) (where the "differences between the unadjudicated patent claims and adjudicated patent claims do not materially alter the question of invalidity, collateral estoppel applies.") (citations omitted); *MaxLinear, Inc. v. CF CRESPE LLC*, 880 F.3d 1373, 1376 (Fed. Cir. 2018) (collateral estoppel

"applies in the administrative context"). "A party seeking to apply collateral estoppel based on a prior action must show that (1) the identical issue was actually litigated; (2) the issue was actually decided in a final decision on the merits; (3) the issue was necessary to the final decision; and (4) the party being estopped was adequately represented in the prior action." *Alphatec Holdings*, 2020 Pat.App. LEXIS 12487, *28, quoting *United Access Techs., LLC v. CenturyTel Broadband Services LLC*, 778 F.3d 1327, 1331 (Fed. Cir. 2015).

Applying the identical-issue requirement to patent claim language, "[c]omplete identity of claims is not required to satisfy the identity-of-issues requirement for claim preclusion." *Soverain Software*, 778 F.3d at 1319. Rather, where the "differences between the unadjudicated patent claims and adjudicated patent claims do not materially alter the question of invalidity, collateral estoppel applies." *Id.* (citations omitted); *see also Ohio Willow Wood Co. v. Alps South, LLC*, 735 F.3d 1333, 1342 (Fed. Cir. 2013) ("Our precedent does not limit collateral estoppel to patent claims that are identical. Rather, it is the identity of the issues that were litigated that determines whether collateral estoppel should apply."). Likewise, "the mere use of different words in [] portions of the claims does not create a new issue of invalidity." *Id.* at 1343.

Accordingly, issues decided in the '552 IPR against Patent Owner and affirmed by the Federal Circuit are binding in this proceeding. *Alphatec Holdings*,

2020 Pat.App. LEXIS 12487, *24 ("[A]s to a different petitioner, when the prior decisions finding claims unpatentable have 'subsequently been affirmed by [the Federal Circuit] those prior decisions … are binding in this proceeding, as a matter of collateral estoppel.'"). Therefore, the findings that Sourcefire is prior art and that it teaches materially similar elements of the '552 patent claims are binding here. *Amazon.com, Inc., v. M2M Solutions LLC*, IPR2019-01205, Paper 43 at 45 (PTAB Jan. 25, 2021) ("Based on collateral estoppel, P.O. is precluded from re-litigating whether Kloba teaches [the subject matter] in the preamble of claim 1."); *Webpower, Inc. v. WAG Acquisition, LLC*, IPR2016-01239, Paper 21 at 27 (PTAB July 8, 2020) ("We agree with Petitioner that collateral estoppel applies to P.O.'s argument with respect to the reloading limitation."); *Alphatec Holdings*, 2020 Pat.App. LEXIS 12487, *31 ("Patent Owner is precluded from relitigating here whether [references] teach all the limitations of claim 1, as arranged in the claim, including for purposes of determining the patentability of dependent claims …").

D. Independent Claims 1, 8 and 15

1. [1pre], [8pre], [15pre]

[1pre] A method comprising: [receiving, by a computing system comprising memory and at least one processor][9],

[9] The bracketed portion of 1a is included here because its recitation of a computing system comprising memory and a processor overlaps with the other preambles.

[8pre] An apparatus comprising: at least one processor; and memory comprising instructions that, when executed by the at least one processor, cause the apparatus to:

[15pre] 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

Sourcefire discloses packet-filtering methods. EX-1003 ¶236; EX-1004 222, 226-227. Sourcefire describes 3D Sensors with IPS (*computing systems*, *apparatuses*, or *computing devices*) that receive and examine packets that traverse a network, looking for malicious activity, and report and/or take actions based on detected malicious activity. EX-1003 ¶¶171-173.

Sourcefire discloses "memory," "processors," and "computer-readable media," as found by the Board in the '552 IPR, as each 3D Sensor with IPS includes a processor and memory that executed instructions to perform tasks, such as dropping or allowing packets according to rules employed by the 3D Sensor. EX-1003 ¶¶237-238; EX-1004 222-223, 234-235, 253-254, 257, 262-264, 271, 435-439; EX-1067 ('552 FWD) 36-37.

Sourcefire discloses a "CPU" and using "available memory," which corresponded to a processor and memory that stored instructions for execution by the processor of the packet-filtering device. EX-1003 ¶238; EX-1004 2050, 2056. Additionally, each 3D Sensor with IPS included software (i.e., instructions) on a disk (i.e., a non-transitory computer-readable medium) that would cause the 3D

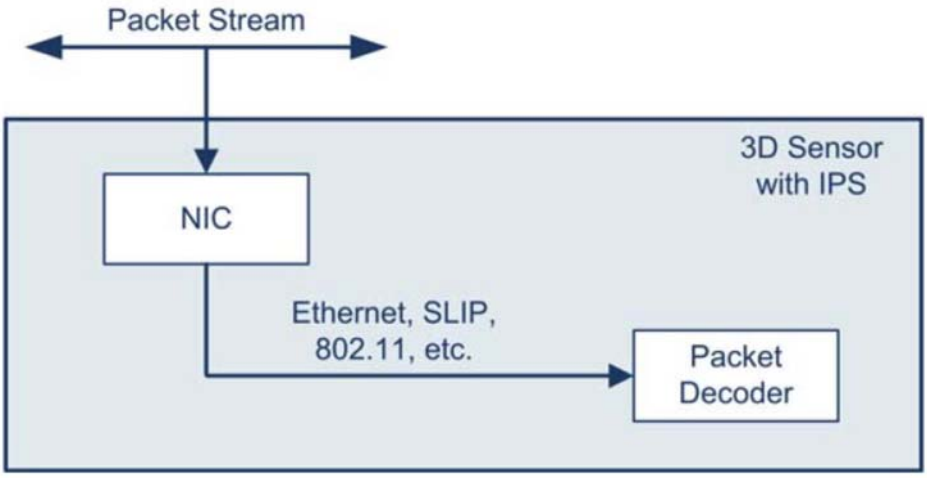
Sensor to perform the tasks set forth in the claim elements. EX-1003 ¶238; EX-1004 82.

2. [1a], [8a], [15a]

[1a] receiving, by a computer system comprising memory and at least one processor, a plurality of packets wherein the plurality of packets comprises a first portion of packets and a second portion of packets;

[8a], [15a] receive a plurality of packets;

Sourcefire discloses the 3D Sensor (a computer system comprising memory and at least one processor, as discussed above) receiving packets, subsets of which may be deemed a first portion of packets and a second portion of packets. EX-1003 ¶¶239-241; EX-1004 213-214, 225-227, 234-235, 260, 277-278, 333-335, 373-379.



As seen in the above figure, the 3D Sensor receives a "packet stream" via a communication interface. EX-1004 257. The IPS performed functions on received packets using a series of decoders, preprocessors, and a rules engine. EX-1004 34-35, 254-256. Receiving communications through "inline" interfaces allows for the

rules engine to allow or block packets. EX-1003 ¶240; EX-1004 222-223, 234-235, 253-254, 257, 262-264, 435-439.

3. [1b], [8b], [15b]

[1b] determining, 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;

[8b], [15b] 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;

The '343 Patent discloses a two-stage filtering process based on "packet header values":

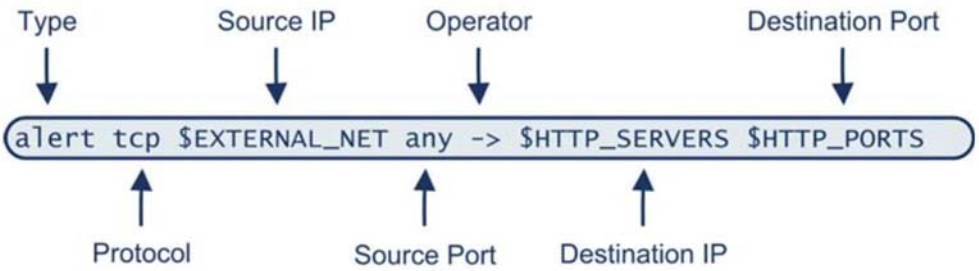
> The filtering process described herein may be viewed as having two (2) stages: A first stage in which the "5-tuple" of IP packet header field values and transport protocol (e.g., TCP, UDP, etc.) packet header field values may be filtered; and a second stage in which application packet header field values may be filtered (e.g., by applying operator logic similar to that described above).

EX-1001 8:39-45. As illustrated in Fig. 3 of the '343 Patent, examples of packet header field values include IP protocol, source IP address, source port, destination IP address, and destination port. EX-1001 5:50-64, Fig. 3.

Sourcefire discloses that intrusion rules had 5-tuple criteria associated with them. EX-1003 ¶245; EX-1004 256-259, 761-770. Such two-stage packet filtering

operation was previously found by the Board in the related '552 IPR. EX-1067 at 53 ("Sourcefire discloses determining whether to apply an operator in a two-stage packet filtering operation"); EX-1069 at 20 (affirming Board decision).

The rule criteria included the source IP addresses and port and the destination IP addresses and port. EX-1003 ¶245; EX-1004 358, 488-490, 761-762. For example, the following figure illustrates the different parts of a rule header that includes "type;" "protocol;" "source IP," etc.:



EX-1004 764. Sourcefire describes that the IP address rule criteria could be a range of IP addresses, like the "140.210.*" range shown in Figure 3 of the '343 patent. EX-1004 762-763, 767, 769, 702-704, 59-60. As in the '343 patent, the rule header in Sourcefire was evaluated first before consideration of application layer data. EX-1003 ¶246; EX-1004 764; EX-1018 4. Thus, a POSA would understand that 3D Sensors of Sourcefire received a number of packet-filtering rules that included first criteria (in the rule header) configured to cause the 3D Sensor to determine that certain packets corresponded to one or more packet-header field values specified in those criteria. EX-1003 ¶¶246-248.

4. [1c], [8c], [15c]

[1c] responsive to a determination by the computing system that a packet header field value of the first portion of the packets comprises data corresponding to the first criterion specified by at least one matching packet-filtering rule, applying, 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;

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

The language of element 1c prior to "applying" indicates that the remainder of element 1c occurs "responsive" to the determination of element 1b above. Similarly, the portion of elements 8c and 15c prior to "apply" indicate that the remainder of those elements occurs "responsive" to the determination of elements 8b/15b, except applied to a subset of the plurality of packets — the "first portion" — which could be any subset of the packets. EX-1003 ¶249. Thus, when the first criterion of 1b/8b/15b has been met, then "one or more" operators specified by the "one or more" packet-filtering rules are to be applied to every packet in the first portion of packets. As discussed above, the "operator" is some text in a rule that will invoke the functionality of the source code of the system. *See* § V.C above. It may determine application header field values in packets seen by the system, and either allow the packets or drop them according to those values. EX- 1003 ¶250.

Because the rule option keywords in Snort are not checked unless the five-tuple fields are matched (*see* element 1b), Sourcefire discloses the limitation of invoking the "operator" "responsive to the determination" that the packet header field value specified in the "first criterion" has been met. EX-1003 ¶¶251, 186-201. Thus, Sourcefire discloses or renders obvious elements 1c/8c/15c.

Further, the Board already found in the '552 IPR that Sourcefire discloses applying an operator that specifies one or more application-header-field-value criteria and a packet transformation function. EX-1067 16-17, 48-49 ("we agree with Petitioner's argument that Sourcefire discloses an 'operator' that specifies (1) the keyword and argument that indicates 'application-header-field-value criteria,' e.g., TLS version 1.0, and (2) a 'packet transformation function,' e.g., blocking packets that match the criteria."); EX-1069 20 (affirming Board's finding that Sourcfire teaches the claimed operator). *Nestlé*, 884 F.3d at 1351 ("Collateral estoppel protects a party from having to litigate issues that have been fully and fairly tried in a previous action and adversely resolved against a party-opponent.")

5. [1d], [8d], [15d]

[1d] 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; and

[8d], [15d] determine, based on an application header field value, 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 that the "rule options" portion of the rule (second criterion) could be used to detect particular application header field values. In particular, Sourcefire discloses (in a section titled "HTTP Content Options") that rule options can look for particular values of the HTTP method, the HTTP request URI, the HTTP Cookie Header, etc. Those are examples of "application header field values." EX-1003 ¶¶252-253; EX-1004 783-788; *see also* EX-1067 48-49 ("Sourcefire discloses an 'operator' that specifies (1) the keyword and argument that indicates 'application-header-field-value criteria,' …"); EX-1069 20 (affirming Board's finding).

Specifically, as in the '343 patent, Sourcefire discloses that the rule options can assess, "the request method field, which identifies the action such as GET, PUT, CONNECT, and so on to take on the resource identified in the URI." EX-1003 ¶253; EX-1004 786. The HTTP-EXFIL operator disclosed in the '343 patent likewise assesses, based on application header field values, whether HTTP methods correspond to second criteria such as "GET, POST, PUT, DELETE, and CONNECT." EX-1001 7:65-8:4; EX-1003 ¶¶186-189, 254.

Sourcefire also describes the use of SSL/TLS rule keywords to invoke the application layer SSL preprocessor and extract information about SSL or TLS

version and session state from SSL/TLS record headers in packets for an encrypted session. EX-1003 ¶256; EX-1004 825-828, 491, 597-601, 697-700. A POSA would appreciate that SSL was a widely adopted standard technology for safeguarding sensitive data being sent between systems. EX-1003 ¶¶128-132, 256. A POSA also understood that TLS was widely adopted as an updated, more secure, version of SSL. *Id.* Sourcefire discloses that the SSL preprocessor (including when invoked by the rules engine) implemented the SSL preprocessor rules and intrusion rules including SSL keywords (*e.g.*, SSL_version) for both SSL and TLS. EX-1003 ¶256; EX-1004 827-828, 491, 596-601, 700. 258. SSL versions are application header field values, as shown by the REQUIRE-TLS-1.1-1.2 operator in the '343 patent which picks the version number out of an SSL connection application header field value and makes a determination which packet transformation function to use on the packet (BLOCK or ALLOW). EX-1001 8:20-38; EX-1003 ¶¶190, 257. Indeed, in the '552 IPR the Board found that Sourcefire disclosed or rendered obvious determining whether packets have one or more application-header-field values corresponding to one or more TLS-version values, an example "second criterion." EX-1067 48-49 (Sourcefire discloses an "operator" that specifies "the keyword and argument that indicates 'application-header-field-value criteria,' e.g., TLS version 1.0"); EX-1069 17-18, 20; EX-1003 ¶258.

Not only does Sourcefire disclose the use of rule options inspecting application header field values, Emerging Threats rules published by 2010 show that the open source intrusion detection community in fact wrote and disseminated rules doing so. For example, rule SID:2003021 included in Emerging Threats, as discussed above in § IV.C, would be understood by a POSA as a classic exfiltration rule looking transmission of encrypted data out of a network on an unusual port using an out-of-date encryption protocol. EX-1020 109; EX-1003 ¶¶259-260, 264, 267. As discussed above, the operator, in part, checks that earlier rules had established that there was an ongoing SSL connection. EX-1003 ¶¶227, 260; EX-1004 851-855. The remainder of the operator notes, "content:"|17 03 00|"; depth:4," which is looking for a particular SSL/TLS record header. The Snort content operator is looking for hex bytes |17| (corresponding to APPLICATION_DATA) followed by 03 and 00 (corresponding to SSL 3.0 major and minor version numbers). The "depth 4" in the operator above indicates that the search for the content cannot go more than 4 bytes into the packet payload, which is looking for the header at the beginning of the record. EX-1003 ¶¶227, 260; EX-1004 783. A POSA would therefore recognize that this rule determines, based on an application header field value (the field value showing the SSL/TLS version), a second portion of packets based on whether the first portion of packets (the packets that matched the 5-tuple criteria) correspond to second criteria (the application header field values indicate SSL v3.0) specified by

one or more operators specified by the matching packet-filtering rule. EX-1003 ¶267. The Board's finding in the '552 IPR that Sourefire's disclosure of the ssl_version keyword in a packet-filtering rule discloses an "operator" that specifies "the keyword and argument that indicates 'application-header-field-value criteria,' e.g., TLS version 1.0" confirms that the keywords in the Emerging Threats rule to check the SSL version teach this element. EX-1067 50.

As discussed above in § VI.B, a POSA would further recognize this operator disclosed in Emerging Threats as akin to the REQUIRE-TLS operator of the '343 Patent. EX-1001 7:41-49, 8:7-38; EX-1003 ¶254. The second portion of packets would be those matching the application header field value in the operator(s). EX-1003 ¶257.

A POSA would be motivated to combine the Emerging Threats rules with Sourcefire because Sourcefire was designed for users to load and use third-party Snort rules, and Emerging Threats published Snort rules for use with network intrusion detection and prevention systems like Sourcefire. *See* §§ VI.A, VI.B.

6. [1e], [8e], [15e]

[1e] 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, 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.

[8e] 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, apply, to each packet in the second portion of packets, 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.

[15e] 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, apply, 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.

Sourcefire discloses that rules have rule states that may be set to alert (i.e., generate events) or drop (i.e., drop packets that trigger the rule). EX-1004 435-438; EX-1003 ¶¶162, 262; *see also* § VI.A above. Thus, the Sourcefire rules specify a packet transformation function that will "indicate whether each packet in the second portion of packets is allowed to continue toward its destination" as claimed.

Further, as explained in § VI.B above, Emerging Threats contains numerous rules targeting exfiltration operations. In the above example of Emerging Threats rule SID:2003021, the rule determines whether a second portion of packets comprise data corresponding to the second criterion (for this rule, SSL v3.0) specified by operators specified by the matching packet-filtering rule (i.e., the rule that matched the first criteria). EX-1003 ¶¶263-267; *see also* element [1d] above; EX-1003 ¶¶211-229. A POSA would recognize Emerging Threats rule SID:2003021 as a rule

configured to prevent an exfiltration operation by detecting attempts to transmit traffic to unusual ports using an out-of-date and insecure encryption protocol. EX-1003 ¶¶128, 228-229, 263-264.

A POSA would understand that while Emerging Threats ships all rules set to "alert" for initial evaluation on a user network, the POSA would be motivated to change rules to "drop" — and therefore prevent an exfiltration operation — when tuning the rules to their network environment. EX-1003 ¶¶268-269. As described above, a user of a Sourcefire system would import text rules initially disabled, and would then tune their rule set to match the network environment, including selecting rules to activate and selecting whether to set each rule to block or alert. EX-1004 1857-1859, 438; EX-1003 ¶268; § VI.A (showing screens for importing rules and selecting rule states). For example, when using the Emerging Threats rules with Sourcefire, a user would enable select rules with an "alert" action to evaluate their performance in the user's network environment, and then set the rules to "drop" once the user observes that they operate as desired. EX-1003 ¶268. This would have the effect of blocking suspected exfiltration operations of this particular kind (*i.e.*, prevent an exfiltration operation). Meanwhile, packets in connections that match the same 5-tuple, but that did not show this pattern (of SSL v3 application field headers), would be allowed through. EX. 1003 ¶268. The block or allow packet

transformation function applied would indicate whether each packet in the second portion of packets is allowed to continue toward its destination. *Id.*

Thus, a POSA would understand that Sourcefire in view of Emerging Threats renders claims 1, 8, and 15 obvious.

Further, the Board found in the '552 IPR that Sourcefire discloses applying an operator that specifies one or more application-header-field-value criteria and a packet transformation function. EX-1067 16-17, 48-49; EX-1069 20 (affirming Board's finding). An Emerging Threats rule imported into Sourcefire therefore also includes an operator that specifies one or more application-header-field-criteria (i.e., second criteria) and a packet transformation function. As discussed above, a POSA would have understood that numerous of those Emerging Threats rules are configured to detect or prevent exfiltration operations, and would have found it obvious to configure Sourcefire using such rules to prevent an exfiltration operations. Setting the rule to block would indicate that each packet in the second set of packets (i.e., packets with application header field values matching the second criteria) would not be allowed to continue toward their destinations.

E. Dependent Claims

1. Claims 2, 9, and 16

[2/9/16] [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:] 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.

Sourcefire discloses that the IP protocol field is part of the 5-tuple specification in the header of each rule. "In each rule header, you must specify the protocol of the traffic the rule inspects." EX-1004 766. Allowed values for the field are ICMP, IP, TCP, and UDP. *Id.* Thus, the first portion of packets would include a packet header field value indicating a protocol type associated with a particular type of data transfer (e.g., TCP) and corresponding to the first criterion specified by the matching packet-filtering rule. EX-1003 ¶270.

All Emerging Threats rules have a protocol specified in their rule header. For example, consider the rule SID:2003021 discussed above. (EX-1020 109). The use of "tcp" after "alert" indicates that only packets with protocol type "tcp" are to be considered by this rule, which would thus be the packet-header field value of the first criterion specified by the rule, and packets that match the packet-filtering rule first criterion (i.e., the first portion of packets) include a packet header field value indicating the TCP protocol associated with the data transfer. EX-1003 ¶¶271-272. Therefore, Sourcefire and Emerging Threats renders these claims obvious.

2. Claims 3, 10, and 17

[3/10/17] [The method of claim 1, / The apparatus of claim 8, / The one or more non-transitory computer-readable media of claim 15,] wherein: 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 that the destination port is part of the 5-tuple specification in the header of each rule. EX-1004 768-769 ("Rule headers specify the source and ***destination ports*** of the traffic the rule inspects.") (emphasis added). Thus, the destination port would be the "packet header field value" of the "first criterion" in the rule. EX-1003 ¶273.

In particular, Emerging Threats demonstrates rules having a destination port specified in their rule header, even if is explicitly a wild-card. For example, consider the rule SID:2003021. EX-1020 109. As discussed above, the "1024:65535" indicates that only packets in a connection to a server with ports between 1024 and 65535 would have the remainder of the rule applied to them. EX-1003 ¶274; *see also* § VI.B above. This indicates a "particular type of data transfer" because this shows that the SSL v3.0 connection is to an unusual port for an SSL connection, since the rule is trying to detect exfiltration of encrypted data. EX-1003 ¶¶263-264, 267. This would thus be the packet-header field value corresponding to the first criterion specified by the matching packet-filtering rule. Other Emerging Threats Rules likewise specify destination port numbers associated with a particular type of data transfer, as discussed above. *See* § VI.B (e.g., SID:2002526 discloses a destination port number ($HTTP_PORTS) characteristic of an HTTP data transfer).

Thus, Sourcefire and the Emerging Threats render these claims obvious. EX-2003 ¶275.

3. Claims 4, 11 and 18

(A) [4pre]/[11pre]/[18pre]

[4pre] The method of claim 1, wherein

[11pre] The apparatus of claim 8, wherein:

[18pre] The one or more non-transitory computer-readable media of claim 15, wherein

Claim elements 4pre/11pre/18pre are discussed above in connection with claims 1, 8 and 15 above.

(B) [4a]/[11a]/[18a]

[4a/11a/18a] the application header field value identifies one or more hypertext transfer protocol (HTTP) methods

Sourcefire, in combination with Emerging Threats, renders these claims obvious. Consider the Emerging Threats rule with SID:2005319. EX-1020 45. This rule looks for a 5-tuple consistent with requests to HTTP servers ("alert tcp $HOME_NET any -> $EXTERNAL_NET $HTTP_PORTS"). EX-1003 ¶277. If this part of the rule matched, then an operator "content: "POST "; depth 5"' looked at the beginning of the request for the HTTP request method to be a POST. *Id.*, EX-1001 7:9-13 (showing POST is an HTTP method).

Additionally, it checks that the request is to a particular URL "/zuzu.php?&r=". As described above in § VI.A, a user of a Sourcefire system would import text rules and have them initially disabled, and would then tune their rule set to match the network environment, including selecting rules to activate and selecting whether to set each rule to block or alert. EX-1004 1857-1859, 438. When

using the Emerging Threats rules with Sourcefire, a user would enable select rules with an "alert" action to evaluate their performance in the user's specific network environment, and then set the rules to "drop" once the user observes that they operate as desired. EX-1003 ¶277. A user would be motivated to set the rule to drop traffic since the Emerging Threats rules are designed to identify threats, such as malware in this example attempting an HTTP POST method, which would be desirable to drop.

(C) [4b]/[11b]/[18b]

[4b/11b/18b] 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:]

Continuing with the discussion above, if the method was GET, the operator in rule SID:2005319 would not match. Any packets not matching the rule would be allowed through, because as discussed above, the rule would only block packets to this 5-tuple and URL that matched the POST HTTP method. EX-1003 ¶278

(D) [4c]/[11c]/[18c]

[4c/11c/18c] allowing each packet of the third portion of packets to continue toward its respective destination; and

See discussion regarding claim element 4b/11b/18b above. Packets in "GET" requests can be deemed the third portion of packets and would be allowed by the rule to continue toward their respective destinations. EX-1003 ¶279.

(E) [4d]/[11d]/[18d]

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

See discussion regarding claim element 4b/11b/18b above. Packets in "POST" requests can be considered to be the fourth portion of packets and would be dropped, and therefore blocked from continuing toward their respective destinations. EX-1003 ¶280.

4. Claims 5 and 12

(A) [5pre]/[12pre]

[5pre] The method of claim 1, the method further comprising:

[12pre] The apparatus of claim 8, wherein the instructions, when executed by the one or more computing devices, cause the apparatus to:

See Claims 1 and 8 above.

(B) [5a]/[12a]

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

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

Sourcefire, in combination with Emerging Threats, renders this element obvious. Emerging Threats rule SID:2010234, in the first stage, looks for connections outbound from the local network to external networks on HTTP ports (e.g. port 80). EX-1020 18; EX-1003 ¶282. Then in the operator, it checks for 'content:"POST "' within the first 5 bytes of the request. *Id.* This is examining the HTTP method, and the rule will trigger its action (i.e. packet transformation function, such as alert or drop) if the method is "POST." The operator is also checking whether the request is to a specific resource on the target server, specifically it checks that 'uricontent:"/senm.php?data="', such that the second criterion will only be matched if the request is going to the url "/senm.php?data=". A POSA would understand that this rule is an exfiltration rule because, like the example provided in the '343 patent, it is designed to detect and block HTTP POST methods. EX-1003 ¶283, EX-1001 7:53-62. A POSA would further understand that, as part of importing the Emerging Threats rules into Sourcefire and tuning their ruleset, this rule could be made a "drop" rule (instead of alert) with the desirable effect of blocking the activities of this particular Fake AV malware ("Guzz/Dload/Vobfus/ZPack"). EX-1003 ¶284. Therefore, when the "application header field value indicates a request for specified resource data," for example because the request method was a GET, the rule would not trigger and the packets would be allowed through to continue toward their respective destinations.

(C) [5b]/[12b]

[5b] blocking, 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 specified resource from continuing toward its respective destination.

[12b] 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 a specified [resource][10] from continuing toward its respective destination.

If the request being matched against the operator specified by Emerging Threats rule SID:2010234 had a request method of POST, then this would match the rule. EX-1003 ¶285. It would be obvious to a POSA that an HTTP POST request would be a "request to submit data to be processed by a specified resource." *Id.* In this case, the specified resource being the URL "/senm.php?data=". When set to block, the packets of the POST request would be blocked, and thus the conditions of Claims 5 and 12 would both be met. EX-1003 ¶286. Thus, Sourcefire and Emerging Threats render obvious these claims.

5. Claims 6, 13 and 19

[6] The method of claim 1, further comprising dropping packets associated with a particular type of data transfer responsive to a determination that the packets

[10] A word is missing from 12b. For purposes of this petition, it is presumed that the omitted word is "resource."

associated with the particular type of data transfer comprise data corresponding to a particular transport layer security (TLS) version value.

[13] The apparatus of claim 8, wherein the instructions, when executed by the one or more computing devices, cause the apparatus to 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.

[19] The one or more non-transitory computer-readable media of claim 15, further comprising dropping packets associated with a particular type of data transfer responsive to a determination that the packets associated with the particular type of data transfer comprises data corresponding to a particular transport layer security (TLS) version value.

Sourcefire, in combination with Emerging Threats, renders these claims obvious. Emerging Threats rule SID:2003021 discussed above looks for a particular SSL/TLS record header. EX-1020 109; EX-1003 ¶287; *see also* element 1d above. The structure of the header includes record type, version and length. The Snort content operator 'content:"|17 03 00|"' looks for hex bytes |17|, which corresponds to APPLICATION_DATA, followed by 03 and 00, which corresponds to SSL 3.0 major and minor version numbers. The "depth 4" in the operator indicates that the search for the content cannot go more than 4 bytes into the packet payload (i.e., looking for the header at the beginning of the record). EX-1003 ¶287; EX-1004 783.

For this rule, once the first criterion matched, the operator checked for a particular TLS/SSL version. EX-1003 ¶288; EX-1004 783. While claims 6, 13, and 19 recite data corresponding to a particular transport layer security (TLS) version, a POSA would understand that SSL 3.0 is a prior version of the SSL/TLS

protocol before TLS 1.0. EX-1038 16-17; EX-1003 ¶¶285-286. A POSA would further understand that part of importing the Emerging Threats rules into Sourcefire and tuning their packet-filtering device was deciding which rules to change to a "drop," which would have the obvious desirable effect of blocking transfers via SSLv3 on high numbered ports that might well be exfiltrations. EX-1003 ¶289; *see also* § VI.A above (describing importing rules and setting rule states in Sourcefire).

6. Claims 7, 14 and 20

[7] The method of claim 1, further comprising dropping 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.

[14] The apparatus of claim 8, wherein the instructions, when executed by the one or more computing devices, cause the apparatus to 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.

[20] The one or more non-transitory computer-readable media of claim 15, further comprising dropping packets associated with a particular type of data transfer responsive to a determination that the packets associated with the particular type of data transfer comprises data corresponding to a particular real-time transport protocol value.

Sourcefire, in combination with the Emerging Threats rules, renders these claims obvious. Emerging Threats rule SID:2003326 is directed to detecting a buffer overflow attack "WEB_CLIENT Apple Quicktime RTSP Overflow." EX-1020 211; EX-1003 ¶¶290-291. A POSA would understand that RTSP, which stands for "Real Time Streaming Protocol," is a "real-time transport protocol" as recited in

claims 7, 14 and 20. EX-1003 ¶291. RTSP is "an application-level protocol for control over the delivery of data with real-time properties. RTSP provides an extensible framework to enable controlled, on-demand delivery of real-time data, such as audio and video." EX-1099 1. The buffer overflow exists in Apple Computer's Quicktime application, and specifically concerns overflows of the buffer for handling URLs for RTSP content. "By persuading a victim to open a specially-crafted QTL file or visit a malicious Web page containing an overly long RTSP URL, a remote attacker could overflow a buffer and execute arbitrary code on the system." EX-1098. The claims refer to "a particular real-time transport protocol value." A POSA would understand this term to refer to any particular real-time transport protocol value, which would include RTP, RTSP, and other protocols. EX-1003 ¶¶291-292.

Emerging Threats rule:2003326 specifies a 5-tuple for packets returning from remote web servers to local clients ("tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any"). EX-1020 211; EX-1003 ¶293. If this matches, then in the operator it looks for the content '|22|rtsp|3a|//'. *Id.* Translating the hexadecimal, this is '"rtsp://', the beginning of a URL for transmission of data over RTSP. EX-1003 ¶293. It would be obvious to a POSA to tune this rule to be a drop rule. If this is done, then testing for "rtsp://" would be "<u>comprising dropping packets associated with a particular type of data transfer responsive to a determination that the packets</u>

associated with the particular type of data transfer comprises data corresponding to a particular real-time transport protocol value", since the rule would block based on the evidence of a RTSP URL, thus preventing further transmission of that content. EX-1003 ¶¶293-294.

The '343 patent discloses that a method to "detect if an IP packet contains a Real-time Transport Protocol" as an example of a use "for purposes *other than* network policy enforcement and *exfiltration prevention*." EX-1001 9:1-10 (emphasis added). Thus, these dependent claims appear to recite limitations that are distinct from the packet transformation function configured to prevent an exfiltration operation, as discussed above in the context of the independent claims. EX-1003 ¶295. As shown above, Emerging Threats discloses both rules for preventing exfiltration operations and for dropping packets associated with particular real-time transport protocol values, and a POSA would have found it obvious to use both types of rules with Sourcefire. *See* § VI.A above (showing that multiple custom rules may be imported into Sourcefire, activated, and set to drop).

VII. SECONDARY CONSIDERATIONS

To the extent raised here, PO's purported secondary considerations evidence raised in other proceedings lacks a nexus to the Challenged Claims and is otherwise unpersuasive for the same reasons the Board has found that evidence unpersuasive in other IPRs involving PO's patents. *E.g.*, *Palo Alto Networks, Inc v. Centripetal*

Networks, Inc., IPR2021-01157, Paper 36 at 39 (Mar. 10, 2023) (finding PO "has not demonstrated the requisite nexus with evidence that the secondary considerations are the direct result of the unique characteristics of the claimed invention"). *See also* EX-1069 19-20 (Federal Circuit affirming the Board's finding that no nexus was shown between the challenged claims in the IPR of the '552 Patent and any of the asserted secondary considerations). As discussed by Dr. Staniford, the prior art shows that systems developed years before the priority date of the '343 Patent show that its claims were obvious to a POSA. To the extent Patent Owner argues that secondary considerations are relevant to this proceeding, such information would not outweigh the strong evidence of obviousness of the '343 patent claims as discussed above. EX-1003 ¶¶296-298.

VIII. CONCLUSION

Petitioner respectfully requests that a Trial be instituted and that the Challenged Claims of the ’343 Patent be canceled as unpatentable.

Respectfully submitted,

Dated: September 5, 2023

By: */James L. Davis, Jr./*
James L. Davis, Jr.
Registration No. 57,325

Counsel for Petitioner Palo Alto Networks, Inc.

CERTIFICATE OF COMPLIANCE

Pursuant to 37 C.F.R. § 42.24(a) and (d), the undersigned hereby certify that the Petition For *Inter Partes* Review complies with the type-volume limitation of 37 C.F.R. § 42.24(a)(i) because, exclusive of the exempted portions, it contains 13,566 words as counted by the word processing program used to prepare the paper.

Dated: September 5, 2023

By: */James L. Davis, Jr./*
James L. Davis, Jr.
Registration No. 57,325

CERTIFICATE OF SERVICE

The undersigned certifies service pursuant to 37 C.F.R. §§ 42.6(e) and 42.105(b) on the Patent Owner by Priority Mail Express of a copy of this Petition for *Inter Partes* Review and supporting materials at the correspondence address of record for the ’343 patent:

BANNER & WITCOFF, LTD.
1100 13th STREET, N.W.
SUITE 1200
WASHINGTON DC 20005-4051

A courtesy copy was also sent via electronic mail to the Patent Owner’s litigation counsel and counsel in IPR2023-00446 at the following addresses:

Litigation Centripetal Counsel:

Stephen Edward Noona Kaufman & Canoles, P.C. senoona@kaufcan.com	Paul J. Andre Lisa Kobialka James Hannah Michael Hao Lee Hannah Lee Kris Kastens Christina Marie Finn Kramer Levin Naftalis & Frankel LLP pandre@kramerlevin.com lkobialka@kramerlevin.com jhannah@kramerlevin.com mhlee@kramerlevin.com hlee@kramerlevin.com kkastens@kramerlevin.com cfinn@kramerlevin.com

IPR2023-00446 Centripetal Counsel:

Lead Counsel	Back-up Counsel
James Hannah (Lead Counsel) Reg. No. 56,369 KRAMER LEVIN NAFTALIS & FRANKEL LLP jhannah@kramerlevin.com svdocketing@kramerlevin.com	Jeffrey H. Price (Back-up Counsel) Reg. No. 69,141 KRAMER LEVIN NAFTALIS & FRANKEL LLP jprice@kramerlevin.com Jenna Fuller (Back-up Counsel) Reg. No. 74,212 KRAMER LEVIN NAFTALIS & FRANKEL LLP jfuller@kramerlevin.com Bradley C. Wright (Back-up Counsel) Reg. No. 38,061 BANNER & WITCOFF, LTD. bwright@bannerwitcoff.com Scott M. Kelly (Back-up Counsel) Reg. No. 65,121 BANNER & WITCOFF, LTD. skelly@bannerwitcoff.com John. R. Hutchins (Back-up Counsel) Reg. No. 43,686 BANNER & WITCOFF, LTD. jhutchins@bannerwitcoff.com

IPR2023-00446 Keysight Counsel:

Gerard M. Donovan (Lead Counsel) Reg. No. 67,771 REED SMITH LLP GDonovan@reedsmith.com	Peter J. Chassman (Backup Counsel) Reg. No. 38,841 REED SMITH LLP pchassman@reedsmith.com Walter M. Egbert (Backup Counsel) Reg. No. 37,317 REED SMITH LLP WEgbert@reedsmith.com Jonathan I. Detrixhe (Backup Counsel) Reg. No. 68,556 REED SMITH LLP jdetrixhe@reedsmith.com Sidharth Kapoor (Backup Counsel) Reg. No. 68,159 REED SMITH LLP skapoor@reedsmith.com

Dated: September 5, 2023

By: */Dara Del Rosario/*
Dara Del Rosario
ROPES & GRAY LLP