UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

PALO ALTO NETWORKS, INC.,
Petitioner,

v.

CENTRIPETAL NETWORKS, INC.,
Patent Owner.

Case IPR2021-01150
Patent No. 10,530,903

PETITION FOR *INTER PARTES* REVIEW

TABLE OF CONTENTS

LIST OF EXHIBITS

Exhibit ("Ex.")	Description
1001	U.S. Patent No. 10,530,903 ("'903")
1002	File History of U.S. Patent No. 10,530,903
1003	Declaration of Dr. Robert Akl, D.Sc. ("Akl")
1004	U.S. Patent Application Publication No. 2014/0280778 ("Paxton")
1005	U.S. Patent No. 8,219,675 ("Ivershen")
1006	U.S. Patent No. 7,185,368 ("Copeland")
1007	U.S. Patent No. 8,413,238 ("Sutton")
1008	U.S. Patent Application Publication No. 2013/0262655 ("Deschenes")
1009	European Patent Application Publication EP 2,482,522 ("McDonald")
1010	U.S. Patent No. 8,621,556 ("Bharali")
1011	U.S. Patent No. 9,628,512 ("Prenger")
1012	U.S. Patent No. 10,931,797 ("Ahn-797")
1013	U.S. Patent Application Publication No. 2006/0048142 ("Roese")
1014	U.S. Patent Application Publication No. 2008/0163333 ("Kasralikar")
1015	U.S. Patent Application Publication No. 2012/0240185 ("Kapoor")
1016	WIPO International Publication No. WO 2014/001773 ("Jarvis")
1017	IPR2018-01654, Pap. 1 (P.T.A.B. Sep. 17, 2018) (Petition for *Inter Partes* Review of U.S. Patent No. 9,560,176)
1018	IPR2018-01655, Pap. 1 (P.T.A.B. Sep. 17, 2018) (Petition for *Inter Partes* Review of U.S. Patent No. 9,560,176)
1019	Email correspondence between the U.S. District Court for the Eastern District of Virginia and Counsel (July 14, 2021)
1020	Amended Complaint, *Centripetal Networks, Inc. v. Palo Alto Networks*, Case No. 2:21-cv-00137, Dkt. 65 (E.D.Va. July 9, 2021)
1021	U.S. Patent No. 5,303,303 ("White")
1022	Postel, J., "Transmission Control Protocol," STD 7, RFC 793, DOI 10.17487/RFC0793, September 1981, <https://www.rfc-editor.org/info/rfc793>
1023	K. Golnabi, R. K. Min, L. Khan and E. Al-Shaer, "Analysis of Firewall Policy Rules Using Data Mining Techniques," 2006 IEEE/IFIP Network Operations and Management Symposium NOMS 2006, 2006, pp. 305-315, doi: 10.1109/NOMS.2006.1687561

1024	R. Dantu, J. Cangussu and A. Yelimeli, "Dynamic control of worm propagation," International Conference on Information Technology: Coding and Computing, 2004. Proceedings. ITCC 2004, 2004, pp. 419-423 Vol.1, doi: 10.1109/ITCC.2004.1286491
1025	T. Baba and S. Matsuda, "Tracing network attacks to their sources," in IEEE Internet Computing, vol. 6, no. 2, pp. 20-26, March-April 2002, doi: 10.1109/4236.991439

I. INTRODUCTION

Petitioner Palo Alto Networks, Inc. ("Petitioner") respectfully requests *inter partes* review ("IPR") of claims 1-18 ("Challenged Claims") of U.S. Patent No. 10,530,903 ("'903 patent") in accordance with §§311-319 and §42.100 et seq.[1]

The '903 patent is directed to correlating packets that pass through a network device—*i.e.*, matching packets received by the network device with packets transmitted by the network device. Ex. 1001, Abstract, 1:38-52. The '903 patent provides a method of correlating the packets by logging data (choosing from a plethora of packet-related information) and matching the packets using the logged data. *Id.*, 3:52-12:55. After packets are correlated, results may be provided to an administrator, or rules may be generated to identify and drop certain packets. *Id.*, 12:56-13:64.

The '903 patent's claims are unpatentable as obvious. Correlating packets was well-known in the art. Known, too, were the various packet information and properties that the '903 patent suggests using for correlation, as well as post-correlation activities such as notifying an administrator or provisioning filter rules. Patent Owner did not discover or invent any new packet information. It merely

[1] Section cites are to 35 U.S.C. or 37 C.F.R. as context indicates, and all emphasis/annotations added unless noted.

applied known networking characteristics and techniques to known packet correlating processes.

As demonstrated below, the prior art renders the Challenged Claims unpatentable, and Petitioner has a reasonable likelihood of prevailing with respect to the same.

II. MANDATORY NOTICES UNDER 37 C.F.R. §42.8

A. Real Party-In-Interest

Pursuant to 37 C.F.R. §42.8(b)(1), Petitioner identifies Palo Alto Networks, Inc., a Delaware corporation, as real party-in-interest.

B. Related Matters

The ’903 patent is the subject of district court litigation: *Centripetal Networks, Inc. v. Palo Alto Networks, Inc.*, Case No. 2-21-cv-00137 (E.D. Virginia., filed March 12, 2021) (“EDVA suit”).

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

Lead Counsel	Backup Counsel
Scott A. McKeown Reg. No. 42,866 ROPES & GRAY LLP 2099 Pennsylvania Avenue, NW Washington, D.C. 20006-6807 Phone: 202-508-4740 Fax: 617-235-9492 scott.mckeown@ropesgray.com **Mailing address for all PTAB correspondence:** ROPES & GRAY LLP IPRM—Floor 43 Prudential Tower 800 Boylston Street Boston, Massachusetts 02199-3600	James Batchelder (*pro hac vice* forthcoming) Mark Rowland Reg. No. 32,077 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 james.batchelder@ropesgray.com mark.rowland@ropesgray.com andrew.radsch@ropesgray.com Victor Cheung Reg. No. 66,229 ROPES & GRAY LLP 2099 Pennsylvania Avenue, NW Washington, D.C. 20006-6807 Phone: 202-508-4641 Fax: 617-235-9492 victor.cheung@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 37 C.F.R. §42.15(a) for this Petition to Deposit Account No. 18-1945. Any additional fees that might be due are also authorized.

IV. REQUIREMENTS FOR *INTER PARTES* REVIEW

A. Grounds for Standing

Pursuant to 37 C.F.R. §42.104(a), Petitioner certifies that the '903 patent is available for IPR and that Petitioner is not barred or estopped from requesting IPR of the Challenged Claims of the '903 patent on the grounds identified herein.

B. Identification of Challenge

Pursuant to 37 C.F.R. §§42.104(b) and (b)(1), Petitioner requests IPR of the Challenged Claims and that the Board cancel the same as unpatentable. The '903 patent matured from U.S. Application 15/413,947, filed 1/24/2017. The '903 patent's earliest priority claim is to U.S. Application 14/618,967, filed 2/10/2015.[2]

1. The Specific Art on Which the Challenge is Based

Petitioner relies upon the following prior art:

Exhibit 1004 – Paxton (U.S. 2014/0280778) published 9/18/2014, and is based on application 14/208,314, filed 3/13/2014, and provisional application 61/778,820, filed 3/13/2013.

Exhibit 1007 – Sutton (U.S. 8,413,238) issued 4/2/2013, and is based on application 12/176,912, filed 7/21/2008.

[2] Petitioner takes no position as to the propriety of the priority claims since the art presented herein pre-dates the earliest filing. Petitioner reserves the right to challenge these priority claims.

Exhibit 1005 – Ivershen (U.S. 8,219,675) issued 7/10/2012, and is based on application 12/636,144, filed 12/11/2009.

Paxton, Sutton, and Ivershen are prior art under AIA §§102(a)(1) and (2).

2. Statutory Grounds on Which the Challenge is Based

Petitioner respectfully requests cancelation of the Challenged Claims on the following ground:

Ground	Statute	Claims	Prior Art
1	§103	1-18	Paxton and Sutton in view of Ivershen

The information required by §§42.204(b)(4)-(5) is provided in Section IX.

This Petition is supported by the Declaration of Dr. Robert Akl (Ex. 1003, ¶¶1-190) ("Akl").

V. THE BOARD SHOULD NOT EXERCISE ITS DISCRETION TO DENY INSTITUTION

The Board should not exercise its discretion to deny institution under §§325(d) or 314(a).

A. §325(d)

Considering the two-part framework discussed in *Advanced Bionics, LLC v. Med-El Elektromedizinische Gerate GMBH*, IPR2019-01469, Pap. 6, *8-9, the Board should not exercise its §325(d) discretion to deny institution.

Neither the art nor the arguments in Ground 1 are the same/substantially the same as those considered during prosecution. Neither Paxton nor Sutton have

been before the Office in any proceedings related to the '903 patent. Ivershen was cited in an IDS during prosecution of the '903 patent and thereafter applied by the Examiner as a tertiary reference in obviousness rejections against pending claims. Ex. 1002, 1486, 1539-1566. Prior to that, Ivershen was presented to the PTAB, as a primary reference, in petitions for IPR filed by an unrelated party against claims of the '903 patent's parent, U.S. 9,560,176 ("'176 patent"). Ex. 1017 (IPR2018-01654); Ex. 1018 (IPR2018-01655).

The ground of rejection herein references Ivershen only for one of the five limitations deemed missing from the prosecution prior art—i.e., determining a difference in timestamps. The prosecution examiner's prior art mappings did not apply Ivershen to this limitation, and thus this limitation was not distinguished with respect to Ivershen. The other prior art references in the ground of rejection, Paxton and Sutton, address the remaining four limitations deemed missing from the prosecution prior art. *See* Ex. 1002, 1551, 1627-1628; Prosecution History, §VI.B below.

Therefore, because prior art presented herein has never been before the Office, because the combination proposed herein has never been presented to the Office, and because the new prior art presented herein squarely address limitations deemed missing from the previously applied prior art, it cannot be said that "the same or substantially the same art previously was presented to the Office" or "the

same or substantially the same arguments were presented to the Office." The Board should not exercise its discretion to deny institution under §325(d).

B. §314(a)

Likewise, co-pending district court proceedings do not warrant the exercise of discretion under §314(a) based on the six factors identified in *Apple Inc. v. Fintiv, Inc.*, IPR2020-00019, Paper 11. **1**: On July 9, Petitioner filed a motion to stay the EDVA suit pending the outcome of this IPR and IPRs addressing the remaining patents asserted in that suit. EDVA courts frequently grant stays pending IPR, including pre-institution. *E.g.*, *RAI Strategic Holdings, Inc. v. Altria Clients Svcs.*, 1:20-cv-393, Dkt. 426 (E.D.Va. Dec. 4, 2020); *Centripetal Networks v. Cisco Sys.*, 2:18-cv-00094, Dkt. 58 (E.D.Va. Feb. 25, 2019); *Sharpe Innovations, Inc. v. T-Mobile USA*, 2:17-cv-351, Dkt. 41 (E.D.Va. Jan. 10, 2018). **2**: As of this filing, the court has not held a scheduling conference or set a firm trial date. The court indicated that it likely will set trial for August 1, 2022 (*see* Ex. 1019), meaning trial would be at least one year away (assuming suit is not stayed). Meanwhile, less than two weeks ago, PO amended its complaint to add a new patent and new allegations to the case. Ex. 1020. **3:** The court has not issued any substantive orders related to the '903. Neither party has produced any discovery or served any contentions. The court's anticipated *Markman* hearing date of March 29, 2022 (Ex. 1019) is approximately three months after the deadline for institution

(assuming suit is not stayed). **4:** The EDVA suit likely will involve multiple grounds of invalidity, including §§101 and 112, and unique grounds under §§102 and 103 not at issue here, enabling the court to focus its limited trial time on different invalidity defenses, if the suit is not stayed. Further, this Petition addresses claims (9, 18) not asserted in the Complaint in the litigation. **5:** The litigation and PTAB parties are the same. **6:** The merits of the asserted grounds in this Petition are particularly strong as shown herein. Accordingly, the Board should not exercise its discretion to deny institution.

VI. BACKGROUND

A. Summary of the '903 Patent

The '903 patent is directed to using logs to correlate packets received by a network device with packets transmitted by the network device.

As shown in annotated FIG. 1 below, an exemplary system includes network device 122 (**green**) that communicates to hosts 108, 110, and 112 (**blue**) in Network A 102 as well as hosts 114, 116, an 118 (**red**) in Network B. The network device also communicates to a packet correlator 128, which includes rules 140 and logs 142 (**orange**). Ex. 1001, 2:36-3:33.

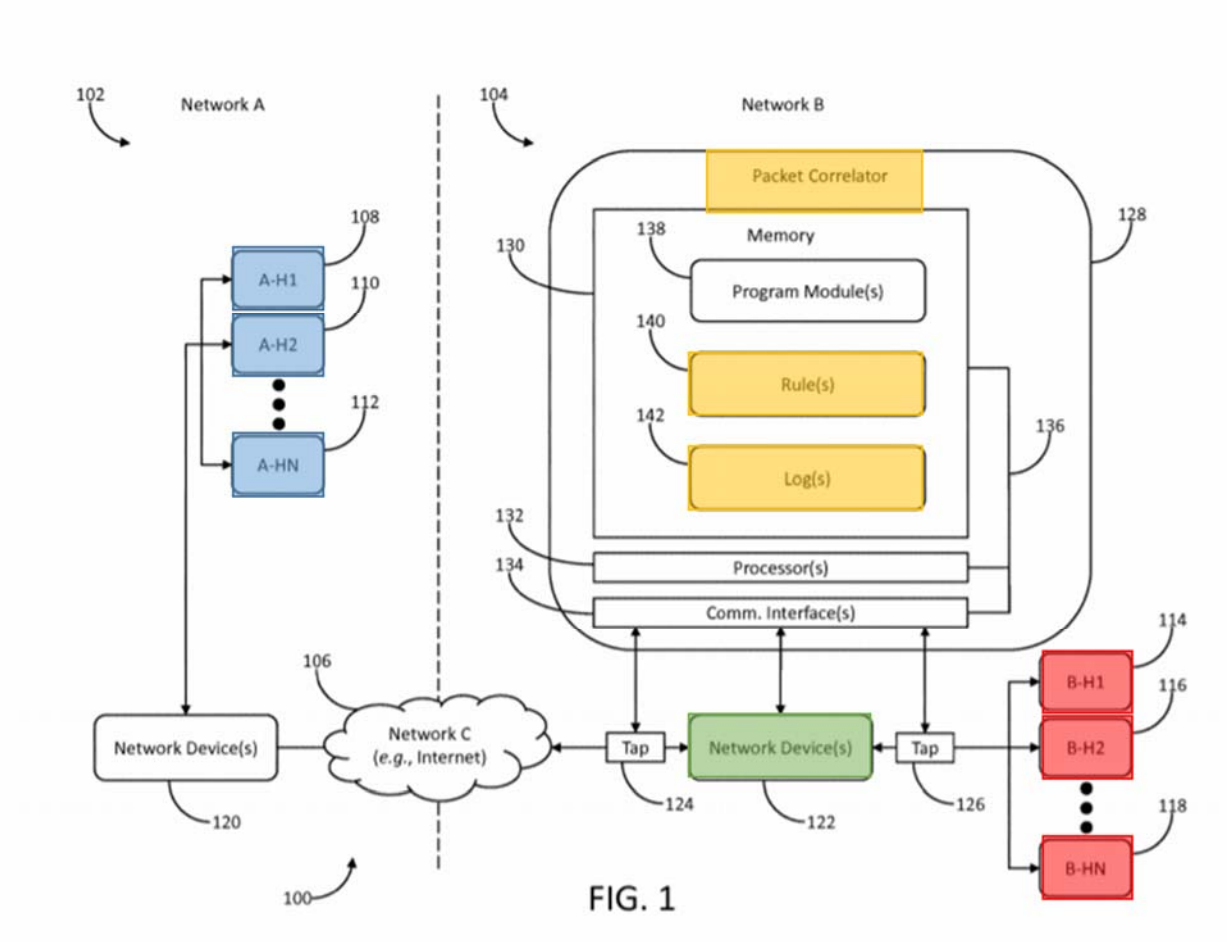


FIG. 1

As shown in the flowchart below, a network device receives packets (402) and generates log entries corresponding to the received packets (404). The network device then transmits another set of packets (406) and generates log entries corresponding to the transmitted packets (408). Finally, the network device correlates the received and transmitted packets based on their log entries (410). *See* FIG. 4 below. Ex. 1001, 13:65-14:21.

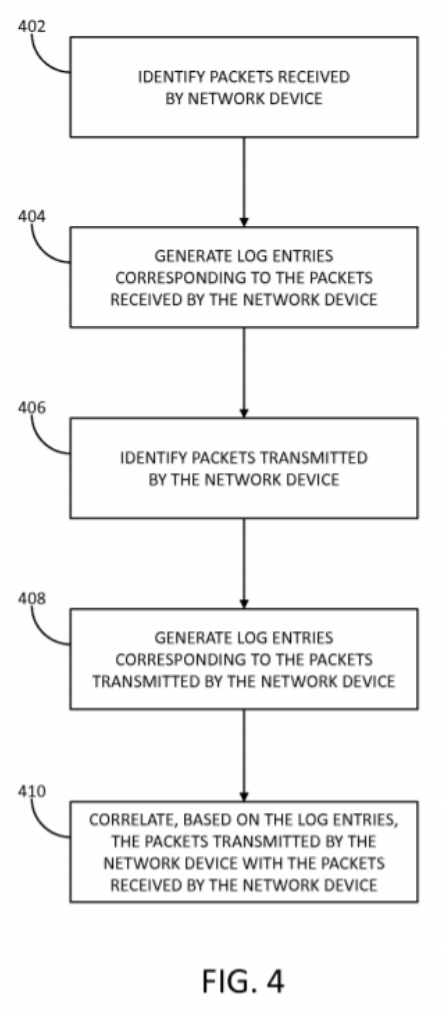


FIG. 4

Correlation may be based on any of a number of factors, such as network-layer information, transport-layer information, application-layer information, and environmental variables. *See*, *e.g.*, Ex. 1001, 4:11-5:3, 8:47-9:44. Once packets are correlated, the system may notify an administrator of the correlation or generate rules to identify and drop packets, for example, after determining that communications were with a malicious entity or if it is suspected that malware is involved. *Id.*, 12:56-13:35. Akl, ¶¶37-50.

B. Prosecution History of the ’903 Patent

U.S. Application 15/413,947, which matured into the ’903 patent, was filed on 1/24/2017 with one claim. A preliminary amendment was filed on 9/5/2017, cancelling the one claim and adding 20 claims, with Applicant remarking that no new matter had been added. Ex. 1002, 1227-1236.

Examiner issued a Non-Final Office Action on 8/9/2018, rejecting a subset of the claims under obviousness-type double patenting and/or as being obvious over combinations of Darisi (U.S. 8,004,994), Curran-Gray (U.S. 2006/0159028), Suzuki (U.S. 2003/0154297), Toumura (U.S. 2006/0114899), and Bostrom (U.S. 2012/0331543). *Id.*, 1344-1410. Claims 8-9, 11, 18-19, and 21 were indicated as reciting allowable subject matter.

Applicant, on 2/8/2019, filed a Terminal Disclaimer to obviate the double patenting rejections and amended the independent claims to allegedly incorporate the allowable subject matter of claims 8 and 18, rendering all claims allowable. *Id.*, 1473-1485.

In a 4/8/2019 Final Rejection, Examiner rejected a subset of the pending claims as being obvious over combinations of Darisi, Rajan (U.S. 8,271,645), Ivershen, Suzuki, Toumura, and Bostrom. Claims 9, 11, 19, and 21 were indicated as reciting allowable subject matter. *Id.*, 1539-1566. In an Interview conducted 5/6/2019, Examiner explained that the Final Rejection was “proper because not all

the features from the objected claims [were] incorporated into the base claims." *Id.*, 1590-1591.

Applicant, in a 10/7/2019 response, introduced substantial amendments to the claims that did not strictly incorporate the indicated allowable subject matter. *Id.*, 1599-1613.

Examiner issued a Notice of Allowance on 11/7/2019 and indicated, as reasons for allowance, that the prior art of record failed to teach the entirety of the steps of "generating…a first plurality of log entries," "generating…a second plurality of log entries," "determining…differences between at least one packet transmission time…and at least one packet receipt time," "generating…an indication of the first host," and "transmitting…the indication of the first host"—*i.e.*, claims [1d]-[1h] as defined in the claim charts below. *Id.*, 1620-1630.

Accordingly, the '903 patent was apparently allowed because the features of claims [1d]-[1h], in combination with the remaining claim limitations, were purportedly missing from or nonobvious over the art of record.

The '903 patent issued on 1/7/2020. Akl, ¶¶51-59.

VII. LEVEL OF ORDINARY SKILL IN THE ART

The level of ordinary skill in the art is evidenced by the prior art. *See In re GPAC Inc.*, 57 F.3d 1573, 1579 (Fed. Cir. 1995) (determining that the Board did not err in adopting the approach that the level of skill in the art was best

determined by references of record). The prior art discussed herein, and in the declaration of Dr. Robert Akl, demonstrates that a POSITA, at the time the '903 patent was filed, had a bachelor's degree in electrical engineering, computer engineering, computer science, or a related field, and approximately 2-3 years of experience in the design or development of telecommunication systems, or the equivalent. Additional graduate education could substitute for professional experience, or significant experience in the field could substitute for formal education. Akl, ¶¶18-20.

VIII. CLAIM CONSTRUCTION

Claim terms subject to *inter partes* review are to be "construed using the same claim construction standard that would be used to construe the claim in a civil action under 35 U.S.C. §282(b), including construing the claim in accordance with the ordinary and customary meaning of such claim as understood by one of ordinary skill in the art and the prosecution history pertaining to the patent." §42.100(b). For purposes of this Petition, Petitioner believes no terms require construction. Only terms necessary to resolve the controversy need to be construed, and should be given their plain and ordinary meaning. *Nidec Motor*

Corp. v. Zhongshan Broad Ocean Motor Co., 868 F.3d 1013, 1017 (Fed. Cir. 2017).[3] Akl, ¶¶60-61.

IX. GROUNDS OF UNPATENTABILITY

Although the '903 patent claims correlating packets between communications networks using packet log entries, such correlation was known prior to the earliest possible priority date of the '903 patent, and the Challenged Claims would have been obvious. Akl, ¶¶37-190.

A. Ground 1: Claims 1-18 are obvious over Paxton and Sutton in view of Iversheen

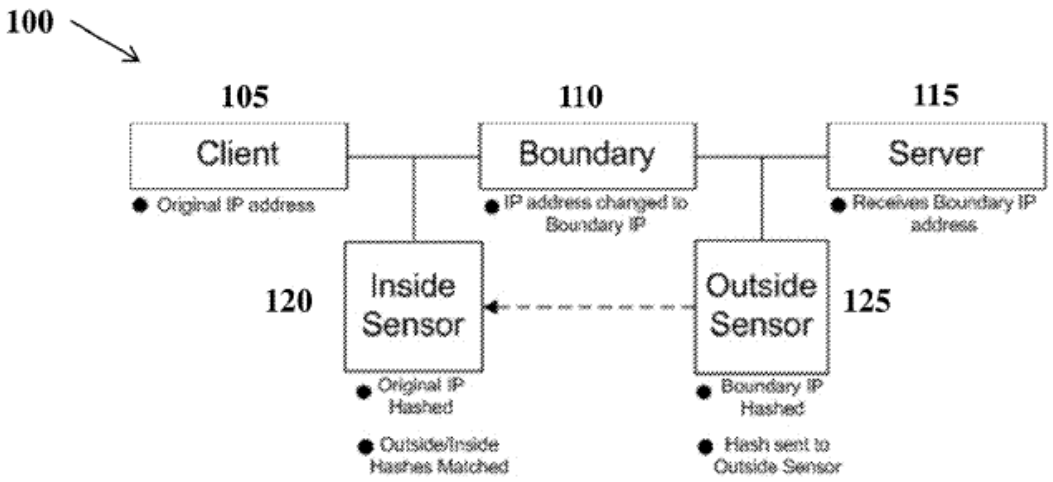
1. Overview of Paxton

Paxton is directed to a "system and method to determine the identity of network packets as they traverse boundaries that perform NAT [Network Address Translation]." Ex. 1004, ¶5. Paxton explains that analyzing an application layer payload before and after a boundary is beneficial to matching packets before and after translation. *Id.*, ¶15. Matching packets and identifying the true source of

[3] *See generally Phillips v. AWH Corp.*, 415 F.3d 1303, 1313 (Fed. Cir. 2005); 37 C.F.R. §42.100(b). The '903 patent is still at issue in pending litigation, so this claim-construction analysis is not a concession as to the scope of any claim term in litigation, or a waiver of any argument in any proceeding that claim terms are indefinite, invalid, or unpatentable.

packet transmissions is useful for network security, providing a way to trace malicious activity sensed at the edge of a network and identify nodes infected with malicious content. *Id.*, ¶30.

Figure 1 shows an example of a system in which packets are sent from client 105 to server 115 across a boundary 110, and vice versa. *Id.*, ¶¶15-16. Paxton's sensors and processing components may be implemented in the same server, in separate servers, and/or in a distributed fashion. *Id.*, ¶¶18, 26.



Inside sensor 120 and outside sensor 125 record traffic before and after it passes through boundary 110 and store, in a database, information including payload hashes, network layer header data, IP addresses, and timestamps of when the payloads are sensed. *Id.*, ¶¶17-20. The database storing information from the inside sensor 120 and outside sensor 125 has direct access to the sensors; alternatively, a consolidated database may be stored at one of the sensors. *Id.*, ¶20.

Payloads can then be matched using at least the hash, time, and IP address data. *Id.*, ¶21. The closest matching hashes, with respect to their timestamps, are identified as matching packets. *Id.*, ¶¶22-23.

Paxton discloses that its computer system is implemented via modules (e.g., an inside sensor module, an outside sensor module, and a matching module) that are implemented via one or more processors executing instructions. *Id.*, ¶32, claims 10-12. Akl, ¶¶62-66.

2. Overview of Sutton

Sutton is directed to "distributed security that monitors communications to identify access attempts to/from darknet addresses." Ex. 1007, Abstract, 2:41-56.

Figure 2 shows a detailed diagram of the security system.

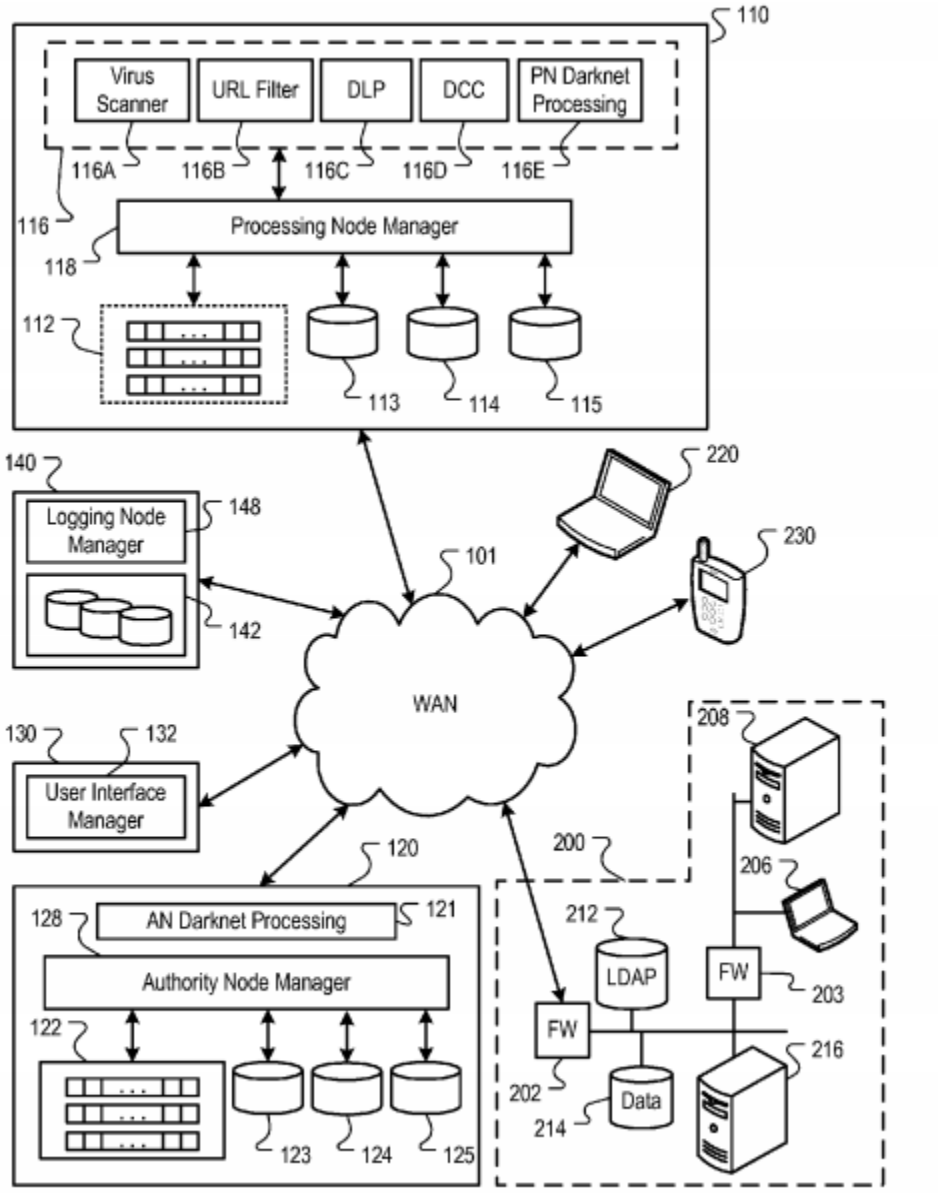


FIG. 2

A wide area network (WAN) 101, such as the Internet, connects the communication of several external systems 200, 220, and 230 through network devices (e.g., routers, gateways). *Id.*, Fig. 2, 5:22-31. Processing node 110, implemented by a plurality of devices including servers, gateways, and switches, processes data communicated through these systems, e.g., serving as a proxy. *Id.*,

3:24-4:4, 4:45-51, 5:5-21. Processing node 110 stores security policies 113 and monitors content items requested by or sent from the external systems. Processing node 110 includes, for example, a detection process filter 112, threat data 114, and data inspection engines 116 to perform threat detection processes. *Id.*, 5:45-7:17. One data inspection engine 116 is PN darknet processing 116E, which identifies communications to or from darknet addresses. *Id.*, 6:15-22.

Once malicious communications are detected or suspected, network administrators may be notified of such communications, devices associated with the malicious activity are identified, and rules may be implemented to prevent or filter further communications. *Id.*, 10:37-11:34, 12:57-13:14. For example, if a malicious host is outside of a network, rules may be generated to prevent communications with that host. *Id.*, 10:66-11:3. Hosts identified as running malicious code may have their communications actively filtered. *Id.*, 11:4-18. Further, other nodes may be instructed to filter and inspect communications for similar activity, and system filters may be updated. *Id.*, 12:57-13:14.

Sutton's processing node 110, which receives and analyzes packets in the overall computer system, may act as a proxy for a device or network, blocks packets when necessary, and updates and implements policies and rules used for analysis. *Id.*, 3:24-53, 4:45-5:4, 5:54-65, 10:37-11:34, 12:25-41, 12:57-13:14, 15:29-35, 15:65-16:5, 17:21-18:16. Sutton discloses using policy data to

implement rules to prevent communications, where policy data define security policies for protected systems, access privileges, disallowed websites and content, etc., and are distributed to processing nodes, which apply the rules to communications. *Id.*, 3:1-23, 3:54-4:4, 7:19-27, 8:11-16, 10:60-11:3. Those processing node functions may be implemented in a plurality of individual devices (e.g., servers, gateways, switches) and/or in a distributed manner across devices. *Id.*, 3:24-37. The various hardware and software of Sutton's system may be integrated into few components or separated into multiple components. *Id.*, 13:26-42, 14:29-57. Akl, ¶¶67-72.

3. Overview of Ivershen

Ivershen is directed to systems and methods for correlating IP flows across a NAT firewall. Ex. 1005, Abstract. Figure 2 shows the components of the system used to correlate IP flows.

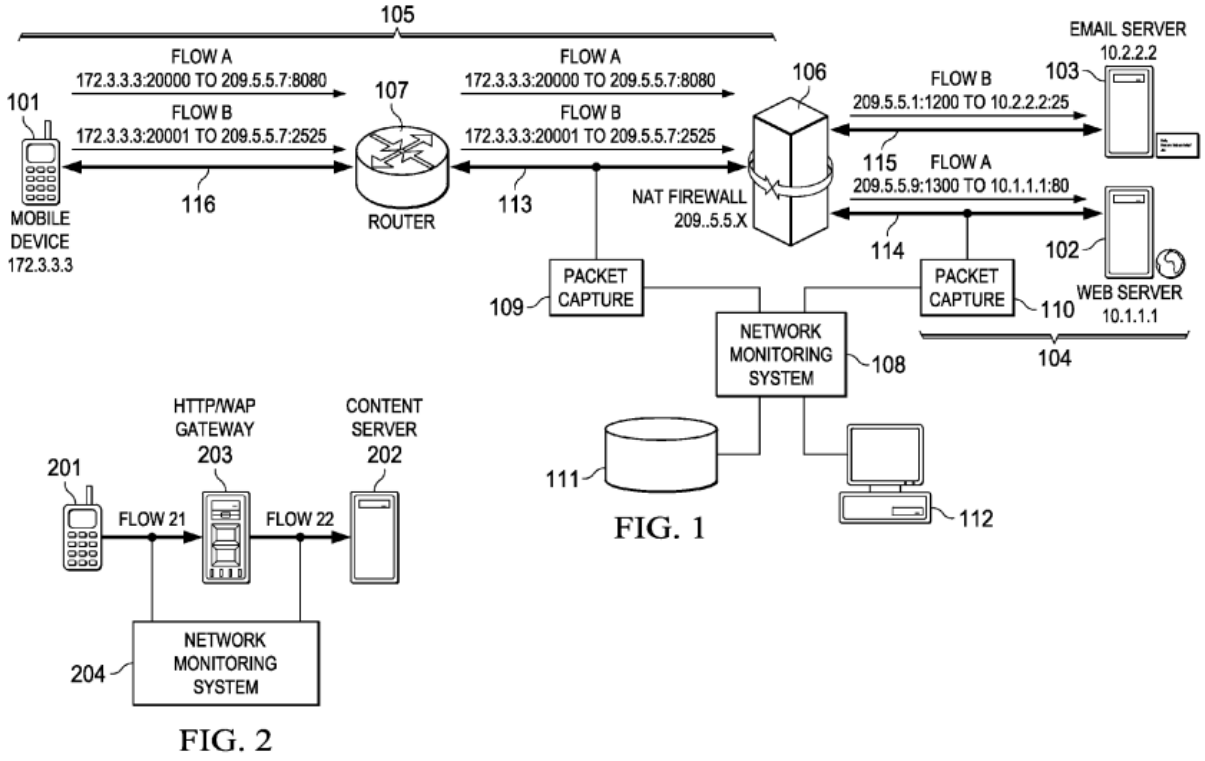


FIG. 1

FIG. 2

Packet capture devices 109 and 110 capture data before and after packets cross a NAT firewall 106. *Id.*, 5:11-13. The network monitoring system 108 then matches (correlates) the packets at both capture devices by searching for flows within close time windows having similar timestamps (differences of a few milliseconds) and determining whether they have matching information. *Id.*, 2:43-51, 5:62-6:11, 8:4-10.

A variety of information may be used to correlate the packets, including L5/L7 data, flow starting timestamps, L7 protocols, HTTP URIs, flow duration, and HTTP header information. *Id.*, 5:36-7:3. This information is typically unmodified through NAT traversal. *Id.*, 5:27-35. Additionally, Iversheh discloses

that while NAT modifies address information passing between networks, a routing table (e.g., containing IP addresses and ports) could be used to correlate packets if it is available and updated. *Id.*, 2:3-8, 4:19-41, 5:11-26. Akl, ¶¶73-76.

4. Motivation to Combine (Sutton)

To the extent Patent Owner argues that Paxton does not explain in detail what actions are taken with respect to identified malicious activity, a POSITA would have been motivated to modify Paxton's computing system to, after the correlating, notify administrators of devices involved with the malicious activity (e.g., as in claim limitations [1g], [10h]) and generate rules to be provisioned to a packet-filtering device (e.g., a gateway, server, or packet inspecting device within the system such as, but not limited to, Paxton's sensor 120 and/or boundary 110, which are, e.g., servers, gateways, and firewalls in the first network; or, alternatively, a similar but separate device in Paxton's multi-device system performing inspecting and filtering functions that would have been included in the first network alongside Paxton's plural sensor and boundary devices to the extent Patent Owner argues a separate device is required) and used for identifying, filtering, and/or blocking host devices' future packet communications (e.g., as in claims 8-9, 17-18), as taught by Sutton. Akl, ¶¶77, 116-123, 151-161, 65-66. As explained above, Sutton teaches network security including the identification of malicious activity, such as communications with darknet addresses. Paxton

discloses a method of tracing sensed malicious activity to its source via correlating packets when, e.g., a network boundary changes the source address of a packet. And, Sutton teaches notifying administrators of devices suspected of association with darknet communications and malicious activity, and generating rules to prevent and/or filter future communications. Thus, when a packet is detected as communicated to/from a darknet address (post-boundary), and Paxton discloses the ability to identify the hosts transmitting/receiving the packet (pre-boundary), Sutton teaches making that identification known to administrators and/or implementing rules to identify or drop future packets to prevent further malicious communications. Accordingly, it would have been obvious to a POSITA to add Sutton's functionality, as discussed above, to Paxton's computing system (e.g., to the device implementing Paxton's matching module, which detects the sources of communications, or to a separate device in Paxton's multi-device system to perform remedial steps) to improve network security. Akl, ¶¶77, 39-40, 45-46, 65-66.

Paxton leaves, to a POSITA, remedial steps (e.g., uses of the correlation results), which are taught by Sutton. The application of known techniques (e.g., Sutton's implementation of rules and data to define security policies, disallowed websites, etc.) to improve similar devices (e.g., servers, gateways, firewalls, etc. in Sutton's and Paxton's systems) to provide predictable results in the same way (e.g.,

to provide packet-filtering functions preventing communications with potentially malicious hosts) would have been obvious to a POSITA. *KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 415-17 (2007).

A POSITA would have had a reasonable expectation of success implementing the detection, identifying, notifying, filtering, and blocking functions, taught by Sutton, with Paxton's system. Paxton explains that correlation is useful for network security and is "highly modular," "can be implemented atop open source technology on commodity hardware," "can provide a stable foundation for building tiered enterprise network architectures with an inherent capability for attribution of malicious activity," and can be integrated by "[e]nterprises with significant visibility and monitoring investments into the network backbone." Ex. 1004, ¶30. Likewise, Sutton is a distributed security system over a network that "ensures…all enterprise traffic…is available for inspection," and its processing node functions are implemented via well-known devices (e.g., servers, gateways, switches, etc.). Ex. 1007, 2:41-56, 3:24-26, 10:37-11:18. The addition of functions for specific packet detection (e.g., identifying darknet communications) and for generating notification messages and rules for existing packet-filtering devices include, generally, simple lookup and text-based data generation routines, which would have been obvious and well within the skill of a POSITA. Akl, ¶¶78, 65-66.

5. Motivation to Combine (Ivershen)

With respect to claims 1 and 10, to the extent Patent Owner argues that Paxton finds matches "based on at least three criteria: hash, time, and IP address," and does so by finding the "closest matching hash (with respect to the timestamp)," but Patent Owner argues that Paxton does not explicitly disclose correlating by functions including determining differences between the timestamps, a POSITA would have been motivated to log packet receipt times and packet transmission times and perform correlation by determining differences between the times, as taught by Ivershen. Akl, ¶¶79, 105-110. As discussed above, Ivershen discloses correlating by determining which packets are within a close time window, matching a "CHKEY, flow starting timestamp, and L7 protocol," and "look[ing] for a flow start timestamp that is within a few milliseconds of the beginning of the flow on the first probe." Ex. 1005, 5:62-6:4, 8:4-10. This accounts for timestamp drift and travel time across interfaces. *Id.*, 2:43-51, 5:62-6:4. A POSITA would have been motivated to apply Ivershen's function of matching for flow starting timestamps to Paxton's receipt and transmission timestamps, and/or to apply Ivershen's function of matching flow starting timestamps in addition to Paxton's timestamp matching, in order to provide more accurate packet correlation results. Akl, ¶¶79.

While Paxton discloses that a first-in-first-out ("FIFO") routine for matching timestamps is an "approach [that] can be leveraged" in order to match hashes with respect to their timestamps, Paxton leaves to the POSITA exact implementation details and preferences for matching packets based on times. Ex. 1004, ¶¶21-22. For example, while Paxton discloses matching packets based on hashes, Paxton does not disclose a preference for where or how to begin searching for a matching hash. As Ivershen teaches that a search for matching timestamps should be within a few milliseconds of the timestamp to be matched, a POSITA would have been motivated to implement Ivershen's matching functions to provide both a determinable starting point to match packets as well as an increased likelihood of matching the correct packets, as Ivershen's functions account for timestamp drift and packet travel time. A POSITA would have recognized that narrowing the field of comparisons to be made increases the overall speed of correlating. Ex. 1005, 7:36-39, 8:4-10. In situations where multiple potential matching hashes and timestamps exist, Paxton's FIFO approach may still be used towards making a final correlation determination. A POSITA would have recognized that there exist many such approaches and packet characteristics at their disposal to resolve ambiguities between potential correlation matches. Ex. 1016, 9:25-10:11. The search for matching packets having similar timestamps, as taught by Ivershen,

serves to increase the likelihood of a unique and correct correlation result. Ex. 1009, ¶¶21-26, 37. Akl, ¶¶80.

Additionally with respect to claims 2-5 and 11-14, to the extent Patent Owner argues that payload and payload hash data are not application-layer data[4] and that Paxton does not explicitly disclose correlating based on a comparison of ports (claims 2, 11), protocol types (claims 3, 12), application-layer data (claims 4, 13), or network-interface identifiers (claims 5, 14), a POSITA would have been motivated to log and these parameters in Paxton's correlation system, as taught by Ivershen. Akl, ¶¶81, 124-144. Both Paxton and Ivershen seek to use typically invariant information to correlate packets; for example, Paxton identifies the application layer payload and Ivershen identifies L5 and L7 data as examples of such information. Ex. 1004, ¶15; Ex. 1005, 5:27-35. Paxton also utilizes other packet information and, in some embodiments, varies hashing functions to make

[4] *See* Ex. 1001, 4:35-43 ("application-layer information (e.g., information derived from one or more application-layer header fields of the packet, such as a domain name, a uniform resource locator (URL), a uniform resource identifier (URI), an extension, a method, state information, media-type information, a signature, a key, a timestamp, an application identifier, a session identifier, a flow identifier, sequence information, authentication information, or the like)."

the best match—making use of "**at least** three criteria: hash, time, and IP address" and employing fuzzy hashing in embodiments where identically matching payload hashes are not expected. Ex. 1004, ¶¶21, 28-29. Ivershen similarly teaches that additional packet properties, beyond a calculated checksum, are used to find matches, narrow match results, and discard false positives—using checksum keys, flow starting timestamps, L7 protocols, flow durations, HTTP URIs, and "other properties." Ex. 1005, 5:62-6:12. A POSITA would have thus recognized that the specific information used to correlate packets is not limited in number or type, and Paxton's correlation system would produce results with increased confidence—e.g., by narrowing multiple potential results as taught by Ivershen—by comparing additional types of packet information. As such, Ivershen teaches that information known to be used for correlation includes ports and network-interface identifiers (Ex. 1005, 2:3-8, 2:52-63, 7:40-8:3), protocol types (*id.*, 5:62-6:7), and application-layer data (*id.*, 6:57-65, 6:8-11). Port information, for example, is packet information that systems at network boundaries already inspect to route packets to their proper destinations, and so it would have been an obvious choice for conducting correlation analysis, which also relates to the routing of those packets. It would have been obvious to a POSITA to utilize this information in addition to the information in Paxton to increase the likelihood of correlating the correct packets. Akl, ¶¶81, 41-44.

Additionally with respect to claims 6 and 15, Paxton in view of Ivershen discloses that the correlating comprises comparing times (first and second times corresponding to the first and second packets), as discussed with respect to claim 1 above (e.g., identifying the closest timestamps). Broadly read, claim 6 and 15's comparison of "first times" and "second times" may refer to the receipt and transmission timestamps of claim 1 or to additional time values. In either case, the prior art renders one or more time comparisons obvious. A POSITA would have been motivated to modify Paxton to correlate first and second packets by also comparing flow durations corresponding to the first and second packets, as taught by Ivershen. It would have been obvious to compare time duration information for the reasons discussed immediately above—to utilize more information to increase the likelihood of correlating the correct packets. Akl, ¶¶82, 145-147.

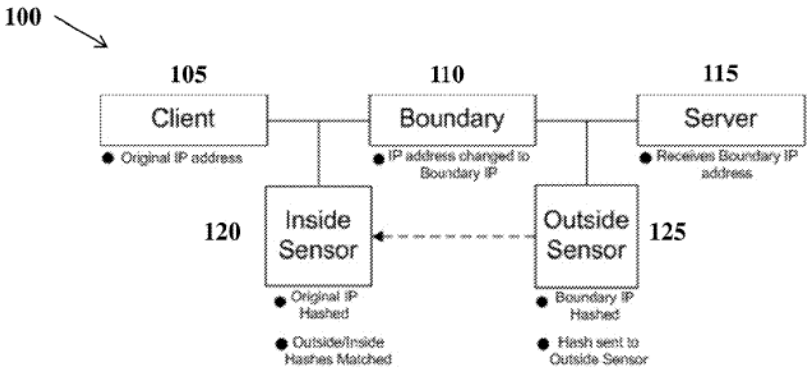
The application of known techniques (e.g., Ivershen's correlation via comparison of packet timestamps, durations, ports, protocol types, application-layer data, network-interface identifiers, etc.) to improve similar devices (e.g., Paxton's and Ivershen's correlation systems) to provide predictable results in the same way (e.g., to provide additional packet information in order to make accurate correlations) would have been obvious to a POSITA. *KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 415-17 (2007).

A POSITA would have had a reasonable expectation of success using the various packet information, taught by Ivershen, in Paxton's system. Paxton's system already maintains a database of hash values, timestamps, and IP addresses, and other header data for correlation. Ex. 1004, ¶¶17, 20. The inclusion of other logged packet information, and the processing of such information, involves little more than the same data processing functions existing in Paxton (e.g., saving and recalling information, comparing values, etc.), which would have been obvious and well within the skill of a POSITA. Akl, ¶83.

6. Claim Chart

'903 Patent	Paxton, Sutton, Ivershen
[1pre] A method comprising:	**Paxton** discloses a **method**. • "The present disclosure relates generally to **identifying network packets, and more particularly, to determining the identity of network packets as they traverse boundaries that perform Network Address Translation (NAT)**." (Paxton, ¶2) Akl, ¶¶84-86.
[1a] determining, by a computing system, that a network device has received, from a first host located in a first network, a plurality of first packets corresponding	**Paxton** discloses **determining, by a computing system** (e.g., determining by a computer system implementing modules, including a boundary, sensors, database), **that a network device** (boundary 110) **has received, from a first host located in a first network** (received from client 105 in client network), **a plurality of first packets corresponding to first requests for content from a second host located in a second network** (packets corresponding to a request for response from a server 115).

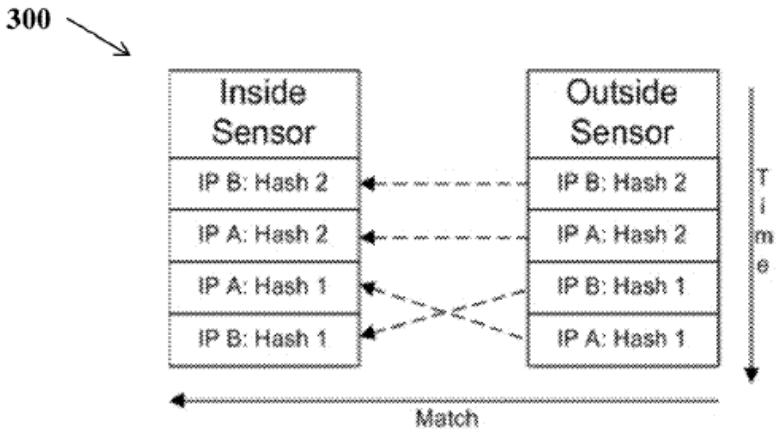
’903 Patent	Paxton, Sutton, Ivershen
to first requests for content from a second host located in a second network,	• “**FIG. 1 is a <u>system diagram 100</u>** for tracking packets across translation boundaries….” (Paxton, ¶15) • “[T]he modules described herein, such as the inside sensor module, outside sensor module, and matching module, **can be implemented in a <u>computer system</u>** that comprises instructions stored in a machine-readable medium and a processor that executes the instructions.” (Paxton, ¶32) • “[T]he exemplary system is described by referring to **<u>packets being sent from a client 105</u> across a boundary 110 to a server 115**. However, one of ordinary skill in the art would understand that this method could be reversed….” (Paxton, ¶16) • “The inside sensor 120 and outside sensor 125 can be two commodity servers running full packet capture in a promiscuous mode via a software package. **While one of ordinary skill in the art would understand that a single server with two network interfaces could suffice for the inside sensor 120 and outside sensor 125, the process can implemented in a distributed fashion as described above in order to scale to the demanding requirements of full packet capture, especially on high bandwidth links. <u>The first server, or inside sensor 120, can passively record traffic on the client 105 network before the contents are altered by a boundary.</u>** The second server, or outside sensor 125, can passively record traffic externally after it has been modified by the boundary.” (Paxton, ¶18) • “Therefore, while **<u>client requests</u> are sourced from the client**, boundary requests alter the original client requests to appear from the boundary. Likewise, **<u>server responses</u> are addressed to the boundary**, whereas boundary responses are altered to appear addressed directly to the client. The boundary alters the source IP address, the

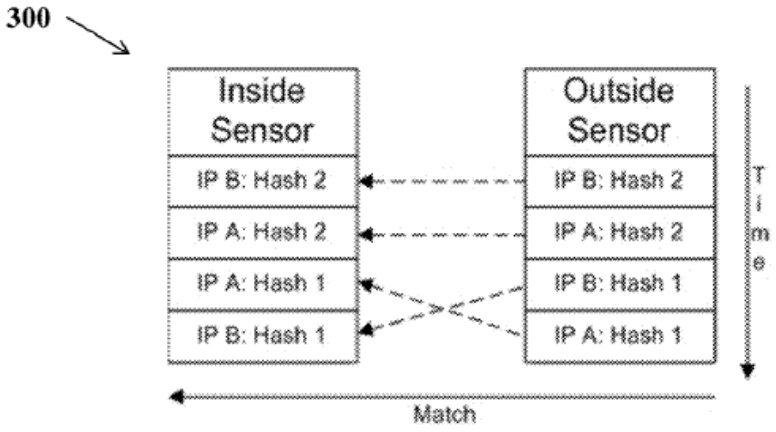
’903 Patent	Paxton, Sutton, Ivershen
	source application ports and their associated checksums within each packet header.” (Paxton, ¶4) • Fig. 1.  • *See also* ¶¶20, 26, 31-32, claims 10-12 (computer system including inside sensor module implemented via processor and stored instructions, outside sensor module implemented via processor and stored instructions, and matching module implemented via processor and stored instructions) (describing architectures for the sensors, boundary, database, and modules). Akl, ¶¶87-91.
[1b] wherein the network device comprises a proxy;	**Paxton** discloses **the network device comprising a proxy**. • “**A boundary can include** routers, **<u>proxies</u>**, gateways, firewalls….” (Paxton, ¶3) Akl, ¶¶92-93.
[1c] determining, by the computing system, that the network device has generated a plurality of second packets corresponding	**Paxton** discloses **determining, by the computing system, that the network device** (boundary 110) **has generated a plurality of second packets corresponding to second requests, wherein the second requests correspond to the first requests** (first packets sensed by the outside sensor server after passing through, and being modified by, the boundary, as second packets), **and wherein the second requests are configured to cause the second host to transmit, to the**

’903 Patent	Paxton, Sutton, Ivershen
to second requests, wherein the second requests correspond to the first requests, and wherein the second requests are configured to cause the second host to transmit, to the network device, the content;	**network device, the content** (client requests sent to the server for response). • *See* [1a]. • “The first server, or inside sensor 120, can passively record traffic on the client 105 network before the contents are altered by a boundary. **The second server, or outside sensor 125, can passively record <u>traffic externally after it has been modified by the boundary</u>.**” (Paxton, ¶18) • “Therefore, while **client requests are sourced from the client**, boundary requests alter the original client requests to appear from the boundary. Likewise, **<u>server responses</u> are addressed to the boundary**, whereas boundary responses are altered to appear addressed directly to the client.” (Paxton, ¶4) Akl, ¶¶94-96.
[1d] generating, by the computing system, a first plurality of log entries corresponding to the plurality of first packets, wherein each of the first plurality of log entries comprises a	**Paxton** discloses **generating, by the computing system, a first plurality of log entries corresponding to the plurality of first packets** (generating first hash data records and storing them in a database), **wherein each of the first plurality of log entries comprises a receipt timestamp indicating a packet receipt time** (hash data records include timestamps of when the payload was sensed[5]), **and wherein the first plurality of log entries comprise first data from the first requests** (hash data records include, e.g., hash, header, address data). • “The packets can be transmitted from a client to a server, or from a server to a client, and the boundary is between the client and the server. **The <u>first hash data record</u> includes** a hash value, an IP address, and **a <u>timestamp</u>** for the first hash of the application layer payload. The second

[5] *See* Ex. 1001, 4:58-65 (receipt times include, e.g., a time when the packet is received, identified, logged, or the like).

’903 Patent	Paxton, Sutton, Ivershen
receipt timestamp indicating a packet receipt time, and wherein the first plurality of log entries comprise first data from the first requests;	hash data record includes a hash value, an IP address, and a timestamp for the second hash of the application layer payload.” (Paxton, ¶6) • “In accordance with an exemplary embodiment of the invention, **as a packet is transmitted from the client 105, the inside sensor 120 can calculate a hash, e.g., a MD5 algorithm hash, of the application layer payload and store it alongside network layer header**. After the packet traverses the boundary 110, the outside sensor 125 can calculate a hash e.g., a MD5 algorithm hash, of the payload along with the header data of the packet.” (Paxton, ¶17; *see also* ¶¶28-29 regarding fuzzy hashing) • “**The hash value from each payload can be stored in a database that has direct access to the inside sensor and outside sensor and is configured to store the first hash data record and the second hash data record along with the IP address and timestamp of when it was sensed.** Alternatively, the first hash data record and the second hash data record can be stored on the inside sensor and outside sensor, respectively. This process can occur on both the inside sensor 120 and outside sensors 125. Furthermore, a separate process can mirror the contents of each sensor's database into a single instance on the inside sensor 120, or the second hash data record can be transmitted to the inside sensor. This process can be performed in order to construct a unified location for data in order to match payloads.” (Paxton, ¶20) • *See also* Paxton, ¶27 (live or recorded capture modes). • Fig. 3, ¶¶24-25 (showing multiple entries for multiple packets).

’903 Patent	Paxton, Sutton, Iversheen
	 Akl, ¶¶97-100.

’903 Patent	Paxton, Sutton, Ivershen
[1e] generating, by the computing system, a second plurality of log entries corresponding to a plurality of second packets, wherein each of the second plurality of log entries comprises a transmission timestamp indicating a packet transmission time, and wherein the second plurality of log entries comprise second data from the second requests;	**Paxton** discloses **generating, by the computing system, a second plurality of log entries corresponding to a plurality of second packets** (generating second hash data records and storing them in a database), **wherein each of the second plurality of log entries comprises a transmission timestamp indicating a packet transmission time** (hash data records include timestamp of when packet was sensed[6]), **and wherein the second plurality of log entries comprise second data from the second requests** (hash data records include, e.g., hash, header, address data). • Paxton, ¶¶6, 17, 20, 28-29 (*see* citations in claim [1d] above regarding the “second hash data record includ[ing] a hash value, an IP address, and a timestamp for the second hash” for packets detected by the outside sensor 125 after traversing the boundary 110). • *See also* Paxton, ¶27 (live or recorded capture modes). • Fig. 3, ¶¶24-25 (showing multiple entries for multiple packets).  Akl, ¶¶101-104.
[1f] determining, by the computing system and for each transmission	**Paxton** discloses **determining, by the computing system and for each transmission timestamp** (determining for timestamps of packets being correlated)**, differences between at least one packet transmission time indicated by transmission timestamps and at least one packet receipt time indicated by receipt timestamps** (each first hash data

’903 Patent	Paxton, Sutton, Ivershen
timestamp, differences between at least one packet transmission time indicated by transmission timestamps and at least one packet receipt time indicated by receipt timestamps;	record is matched with a second hash data record, which includes determining a second hash data record closest in time, i.e., comparatively a smallest difference). • “Hashes from the inside sensor 120 and outside sensor 125 can be matched via a first-in-first-out queue based on recorded timestamps in the first hash data record and the second hash data record. A First-In-First-Out approach can be leveraged in order to match outside and inside hashes with respect to their observed timestamp. **After a hash is observed on the outside, the closest matching hash (with respect to the timestamp) on the inside can be identified as the corresponding match.** The combination of identifiable inside and outside header data can serve as the identity of the packet.” (Paxton, ¶22) • “FIG. 2 is a screenshot 200 of a log that illustrates a matching payload, in accordance with an exemplary embodiment of the invention. The two hashes, preceded by the MD5 label, are identical in FIG. 2. **Furthermore, it is also observed that the time in TimeSecs (seconds) are equal, but the time in TimeMSecs (milliseconds) differ by 814 milliseconds. In other words, the inside packet arrived 814 milliseconds before the outside packet, which is consistent with the inside packet sensing the packet first.** In this case, the identity of the packet is the SrcAddr (source address) of the packet sensed from each side, which is 132.XXX.XXX.102/172.XXX.XXX.240.” (Paxton, ¶23) As explained in §IX.A.5 above, a POSITA would have been motivated to perform correlation in **Paxton’s** system by determining differences between packet receipt times and

[6] *See* Ex. 1001, 6:48-55 (transmission times include, e.g., a time when the packet is transmitted, identified, logged, or the like).

’903 Patent	Paxton, Sutton, Ivershen
	packet transmission times, or between flow start times of first and second packets, as taught by **Ivershen**. **Ivershen** discloses **determining differences** (e.g., determining whether two timestamps are within a few milliseconds of each other; determining whether two timestamps are within a close time window) **between at least one packet transmission time indicated by transmission timestamps** (e.g., timestamps of a first-side packet in relation to a NAT) **and at least one packet receipt time indicated by receipt timestamps** (e.g., timestamps of a second-side packet in relation to a NAT): • “Using a known checksum key from a packet on the first side of the NAT firewall, the checksum keys for all packets with a timestamp within a specified time on the second side of the NAT firewall can be analyzed. For example, **to account for packet transit time, firewall delay and clock errors, the checksum keys for all packets on the second side of the NAT firewall having a timestamp within milliseconds of the first-side packet are analyzed**.” (Ivershen, 2:43-51) • “**Monitoring system 108 can then pull together two or more legs of the session on demand. Starting with a first one of the legs,** such as the session flow and CHKEY created by probe 109 on interface 113, **the other probe 110 is queried with the** CHKEY, **flow starting timestamp**, and L7 protocol **from the first leg of the flow. The second probe 110, searches for a session that matches these parameters. The search on the second probe should look for a flow start timestamp that is within a few milliseconds of the beginning of the flow on the first probe. This allows for timestamp drift among the probes and network travel time across interfaces 113,114 and NATF/WAPG 106. If a match is found, two session flows are successfully correlated together into a single call record.**” (Ivershen, 5:62-6:7)

’903 Patent	Paxton, Sutton, Ivershen
	• “In step 405, the checksum key for an IP flow on the first side of the NAT firewall is compared to the checksum keys for flows on the second side of the NAT firewall. **The flows that are used for comparison on the second side may be limited by using only <u>flows or packets occurring within a time window that is close to or similar to the timestamp</u> of the first-side packet.**” (Ivershen, 8:4-10) Akl, ¶¶105-110.
[1g] correlating, based on the differences and by comparing the first data and the second data, at least a portion of the plurality of first packets and at least a portion of the plurality of second packets; and	**Paxton** discloses **correlating** (matching), **based on the differences** (closest timestamp) **and by comparing the first data and the second data** (determining a closest matching hash), **at least a portion of the plurality of first packets and at least a portion of the plurality of second packets** (identifying the source address of packets from each side of the boundary as matching). • “Hashes from the inside sensor 120 and outside sensor 125 can be matched via a first-in-first-out queue based on recorded timestamps in the first hash data record and the second hash data record. A First-In-First-Out approach can be leveraged in order to match outside and inside hashes with respect to their observed timestamp. **<u>After a hash is observed on the outside, the closest matching hash (with respect to the timestamp) on the inside can be identified as the corresponding match.</u> The combination of identifiable inside and outside header data can serve as the identity of the packet.**” (Paxton, ¶22) • “FIG. 2 is a screenshot 200 of a log that illustrates a **<u>matching payload</u>**, in accordance with an exemplary embodiment of the invention. **<u>The two hashes, preceded by the MD5 label, are identical</u> in FIG. 2. Furthermore, it is also observed that the time in TimeSecs (seconds) are equal, but <u>the time in TimeMSecs (milliseconds) differ by 814 milliseconds</u>. In other words, the inside packet arrived 814 milliseconds before the outside packet, which is consistent with the inside packet**

’903 Patent	Paxton, Sutton, Ivershen
	sensing the packet first. In this case, <u>the identity of the packet is the SrcAddr (source address) of the packet sensed from each side</u>, which is 132.XXX.XXX.102/172.XXX.XXX.240.” (Paxton, ¶23) As discussed with respect to claim limitation [1f], **Paxton** in view of **Iversheen** discloses the correlating based on the differences in timestamps. Akl, ¶¶111-115.
[1h] responsive[7] to the correlating: generating, by the computing system,[8] an	**Paxton** discloses, **responsive to the correlating** (responsive to finding a match), **generating, by the computing system, an indication of the first host** (generating a match log including the identity of the packet source address).

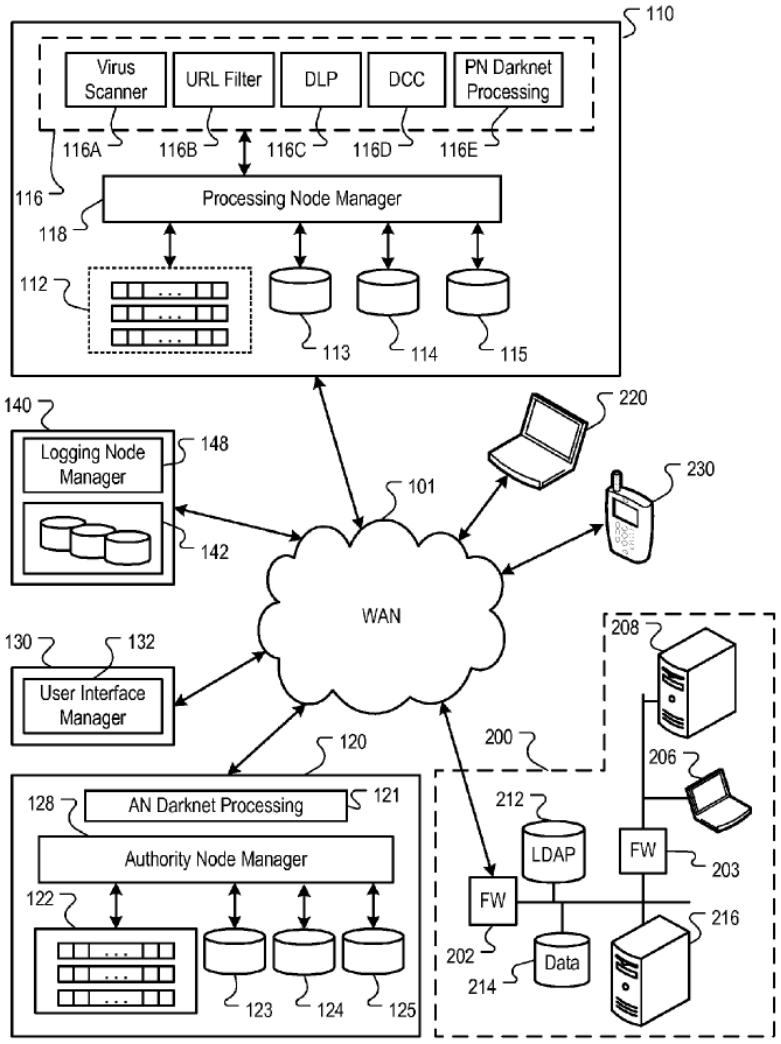
[7] Steps performed “responsive” to correlating are not necessarily performed immediately after correlating. *See* Ex. 1001, 12:59-13:38 (describing a progression of steps including: (step 26) “responsive to correlating,” determining a network address associated with a transmitted packet, (step 27) determining that the packet was transmitted to a malicious entity, (steps 28-29) notifying a user of the communication with the malicious entity, and (steps 30-32) provisioning rules to drop packets and prevent the spread of malware).

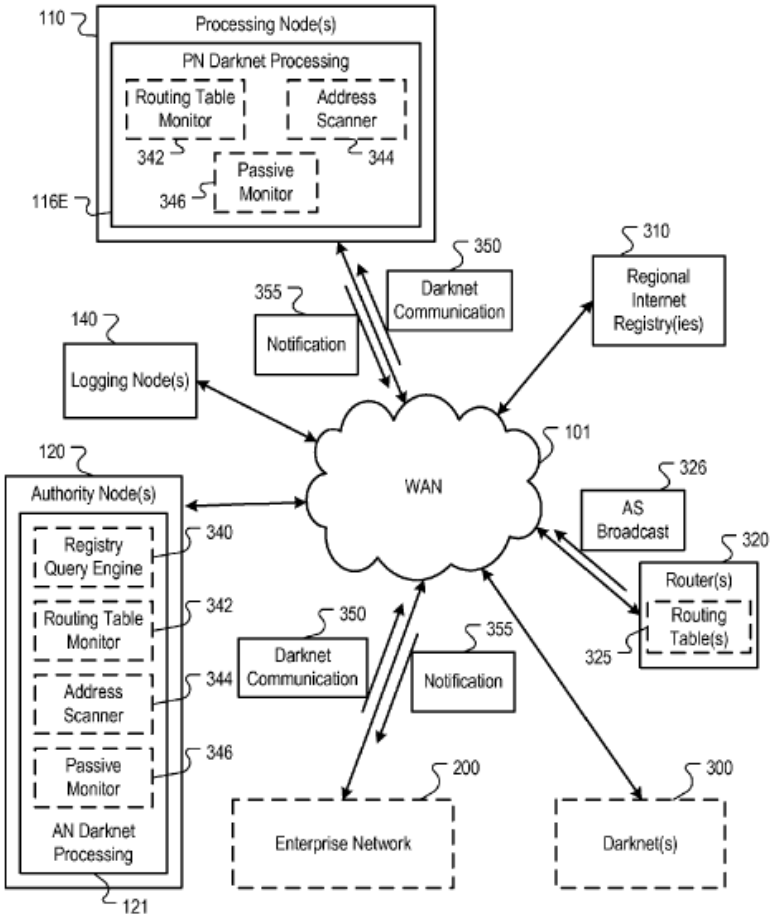
[8] Steps performed “by the computing system” do not preclude user input to or interaction with the computer system in performing those steps. *See* Ex. 1012 (U.S. Patent 10,931,797, a continuation of the ’903 patent) at 15:45-47 and 16:48-50

’903 Patent	Paxton, Sutton, Ivershen
indication of the first host; and transmitting, by the computing system, the indication of the first host.	• “FIG. 2 is a screenshot 200 of a log that illustrates a matching payload…. In this case, **the identity of the packet is the SrcAddr (source address) of the packet sensed from each side**, which is 132.XXX.XXX.102/172.XXX.XXX.240.” (Paxton, ¶23) • “**The ability to identify the true source of packet transmission through a boundary can provide significant benefits to network security.** Current technology that attempts to discover the identity of network packet suffers from authentication and integrity problems. **It can provide a way to quickly identify nodes that are infected with malicious content, which can allow the network administrator to better identify the scope of the malicious incident.** The system and method described herein can be highly modular and can be implemented atop open source technology on commodity hardware. Furthermore, it can provide a stable foundation for building tiered enterprise network architectures with an inherent capability for attribution of malicious activity. **Enterprises with significant visibility and monitoring investments into the network backbone can utilize this technique to attribute malicious activity sensed at the edge of a network back to its original source.**” (Paxton, ¶30) As explained in §IX.A.4 above, **Paxton** discloses correlating packets to identify malicious activity and leaves specific usage and remedial steps to a POSITA. A POSITA would have been motivated to transmit the indication of the first host, e.g., to an administrator, as taught by **Sutton**, responsive to the correlating disclosed by **Paxton**.

(representing that “generating, by the computing system…, one or more rules” comprises “receiving user input defining the one or more rules”).

’903 Patent	Paxton, Sutton, Iversheen
	Sutton discloses **transmitting the indication of the first host** (e.g., notifying an administrator) **responsive to** identifying a device associated with malicious activity. • “Systems, methods and apparatus for a distributed security that monitors communications to identify access attempts to/from darknet addresses. Such attempts can be inferred to be associated with malicious activity and **a notification or other corrective action can be provided identifying such potentially malicious activity.**” (Sutton, Abstract) • “In those implementations where all communications 350 are inspected, the communications 350 can be processed to **identify devices which are likely associated with malicious activity. If such devices reside within the enterprise network, a notification 355 can be provided to the enterprise network (e.g., a network administrator) indicating that such devices are potentially infected with malicious software code. If such devices are outside of the enterprise network, the authority node policy data can be used to implement a rule preventing such devices from communicating with devices within the protected enterprise network.**” (Sutton, 10:60-11:3) • “**In those implementations where only those communications 350 originating from devices within a protected enterprise network are inspected, notification 355 can be provided to the enterprise network (e.g., a network administrator) indicating that the device is potentially infected with malicious software code.** In some implementations, the processing node(s) 110 can attempt to remove the malicious program code from the device originating communications destined for the darknet address space 300. **In other implementations, communications 350 originating from the device can be actively filtered (e.g., through various application specific malware identification programs, such as, performed by processing node**

’903 Patent	Paxton, Sutton, Iversheen
	manager 118 and data inspection engines 116) based upon identification of the probability that malicious code exists on the device.” (Sutton, 11:4-18) • Fig. 2 (showing processing node 110 with processing node manager 118 and data inspection engines 116).  FIG. 2 • Fig. 3 (showing notifications 355 transmitted electronically—i.e., to devices in the computing system)

’903 Patent	Paxton, Sutton, Iversher
	 FIG. 3 • “Each processing node 110 can be implemented by a plurality of computer and communication devices, e.g., server computers, gateways, switches, etc…. In some implementations, each processing node 110 can include Internet gateways and a plurality of server computers, and the processing nodes 110 can be distributed through a geographic region, e.g., throughout a country.” (Sutton, 3:24-37; *see also* 3:38-4:4) Akl, ¶¶116-123.
[2] The method of claim 1, wherein	**Paxton** does not explicitly disclose the correlating based on comparing ports indicated by portions of the first and second packets.

’903 Patent	Paxton, Sutton, Iversheen
correlating the at least a portion of the plurality of first packets and the at least a portion of the plurality of second packets comprises: comparing one or more first ports indicated by the at least a portion of the plurality of first packets and one or more second ports indicated by the at least a portion of the plurality of second packets.	• *See* Paxton, ¶¶4-5. As discussed in §IX.A.5 above, it would have been obvious for a POSITA to correlate first and second packets by **comparing one or more first ports indicated by the at least a portion of the plurality of first packets and one or more second ports indicated by the at least a portion of the plurality of second packets**. **Iversheen** teaches this correlation of packets by comparing port information of portions of the first packets and second packets (e.g., using a NAT translation table to match an [address:port] of a transmission received at the interface on one side of a NAT with an [address:port] of a transmission transmitted from the interface on the opposite side of the NAT). • “Packet capture device 109 captures substantially all of the packets on interface 113, and packet capture device 110 captures substantially all of the packets on interface 114. **As discussed above, NATF/WAPG 106 modifies the address information in the data packets that is passes between networks. As a result, monitoring system 108 cannot use the source or destination address to correlate the packets on interfaces 113 and 114 since the IP addresses and ports are quite different on[] each interface for related messages. <u>The routing table used by NATF/WAPG 106, such as Table 1, could be used to correlate messages on interfaces 113 and 114, but this information may not be available to monitoring system 108. Even if the NAT translation table data was available, monitoring system 108 would require immediate notification of updates or changes to the translation table in order to correlate the packets on legs 113 and 114.</u>**” (Iversheen, 5:11-26) A POSITA would have understood that correlating using the NAT translation table includes comparing ports (and IP addresses) of packets received/transmitted on interfaces 113/114 (e.g., comparing a first packet identified as 209.5.5.7:8080 with second packets to find the corresponding

’903 Patent	Paxton, Sutton, Ivershen
	value 10.1.1.1:80), or at minimum would have found such correlation obvious because the NAT table, e.g., as shown in Ivershen Table 1, distinguishes the public addresses (i.e., relating to the transmitted packets) by port numbers (e.g., 8080 vs 2525). Akl, ¶¶130-133. • “NATF/WAPG 106 uses a NAT translation table, such as Table 1 below, to determine the IP address to use on network 104 for the incoming packets in flows A and B. TABLE 1 NAT Translation Table Destination IP address/port number in incoming packets from network 105 | Corresponding destination IP address/port number on private network 104 209.5.5.7:8080 | 10.1.1.1:80 209.5.5.7:2525 | 10.2.2.2:25 As illustrated in FIG. 1, NATF/WAPG 106 modifies the source and destination address information in the packets of flows A and B. For example, in flow A, the source and destination IP address is 172.3.3.3 and 209.5.5.7, respectively, in network 105. However, in network 104, the source and destination IP address for packets in flow A are changed by NATF/WAPG 106 to 209.5.5.9 and 10.1.1.1, respectively. The port numbers for the incoming and outgoing packets at NATF/WAPG 106 are also changed.” (Ivershen, 4:19-41) Akl, ¶¶124-133.
[3] The method of claim 1, wherein correlating the at least a portion of the	**Paxton** discloses correlating packets on each side of a NAT firewall by comparing network layer header information.[9] • “In accordance with an exemplary embodiment of the invention, as a packet is transmitted from the client 105, **the inside sensor 120 can calculate a hash, e.g., a MD5 algorithm hash, of the application layer payload and store it alongside network layer header**. After the packet

[9] *See* Ex. 1001, 4:24-30 (protocol type is an example of network layer information).

’903 Patent	Paxton, Sutton, Ivershen
plurality of first packets and the at least a portion of the plurality of second packets comprises: comparing one or more first protocol types indicated by the at least a portion of the plurality of first packets and one or more second protocol types indicated by the at least a portion of the plurality of second packets.	traverses the boundary 110, the **outside sensor 125 can calculate a hash e.g., a MD5 algorithm hash, of the payload <u>along with the header data</u> of the packet**.” (Paxton, ¶17) • “**The combination of <u>identifiable inside and outside header data</u> can serve as the identity of the packet.**” (Paxton, ¶22) • “**Payloads can be matched based on at least three criteria: hash, time, and IP address.** When an identical hash is observed on the outside sensor 125 and inside sensor 120, there is a high probability that the hashes belong to the same payload.” (Paxton, ¶21) As discussed in §IX.A.5 above, to the extent Patent Owner argues that **Paxton** does not explicitly disclose **comparing one or more first protocol types indicated by the at least a portion of the plurality of first packets and one or more second protocol types indicated by the at least a portion of the plurality of second packets**, it would have been obvious to a POSITA to correlate packets by comparing first and second packets’ protocol type information, as taught by **Ivershen**. • “Monitoring system 108 can then pull together two or more legs of the session on demand. **Starting with a first one of the legs, such as the session flow and CHKEY created by probe 109 on interface 113, <u>the other probe 110 is queried with</u> the CHKEY, flow starting timestamp, and <u>L7 protocol</u> from the first leg of the flow. The second probe 110, searches for a session that matches these parameters…. <u>If a match is found, two session flows are successfully correlated</u> together into a single call record.**” (Ivershen, 5:62-6:7) Akl, ¶¶134-137.
[4] The method of claim 1,	**Paxton** discloses the correlating based on comparing application-layer data indicated by portions of the first and second packets (e.g., hashes of the application layer payload).

’903 Patent	Paxton, Sutton, Ivershen
wherein correlating the at least a portion of the plurality of first packets and the at least a portion of the plurality of second packets comprises: comparing first application-layer data corresponding to the at least a portion of the plurality of first packets and second application-layer data corresponding to the at least a portion of the plurality of second packets.	• “In accordance with an exemplary embodiment of the invention, **as a packet is transmitted from the client 105, the inside sensor 120 can calculate a hash, e.g., a MD5 algorithm hash, of the application layer payload and store it alongside network layer header**. After the packet traverses the boundary 110, the outside sensor 125 can calculate a hash e.g., a MD5 algorithm hash, of the payload along with the header data of the packet.” (Paxton, ¶17) • “**Payloads can be matched based on at least three criteria: hash, time, and IP address.** When an identical hash is observed on the outside sensor 125 and inside sensor 120, there is a high probability that the hashes belong to the same payload.” (Paxton, ¶21) As discussed in §IX.A.5 above, to the extent Patent Owner argues that **Paxton** does not explicitly disclose comparing application layer data[10] **corresponding to the at least a portion of the plurality of first packets and application-layer data corresponding to the at least a portion of the plurality of second packets**, it would have been obvious to a POSITA to correlate packets using application-layer data as taught by **Ivershen**. **Ivershen** discloses correlating packets on each side of a NAT firewall by comparing first and second packet application-layer data (e.g., header information, domain name, URI, method). • “Preferably, instead of using IP address data, monitoring system 108 would use an invariant correlation key in the flow that is not violated during NAT firewall traversal.” (Ivershen, 5:27-31) • “FIGS. 3A and 3B show that **five headers remain constant during the NAT traversal: Request Method (“GET”) 31A/31B, Host 32A/32B, URI 33A/33B, UE**

[10] *See* footnote 4 regarding “application-layer information.”

’903 Patent	Paxton, Sutton, Ivershen
	Profile (x-wap-profile) 34A/34B, and Cookie 35A/35B. <u>Therefore, a correlation key can be created using these invariant portions of the HTTP header.</u> For example, CHKEY may be created from RequestMethod+Host+URI+x-wap-profile in the first HTTP headers in each flow 21, 22. The Cookie header may also be added to the checksum to reduce false positives.” (Ivershen, 6:57-65) • “comparing a correlation key for one of the first group to the correlation keys for each of the second group of session records to identify session records with matching correlation keys, wherein **the correlation keys are created using data including each of a Request Method, a Host, a URI, and a UE Profile header**.” (Ivershen, 9:12-17) • “**If more than one match is found, then false positives can be identified and discarded by comparing other properties of the flow, such as, for example, the closest flow duration or <u>an exact HTTP URI match</u>.**” (Ivershen, 6:8-11) Akl, ¶¶138-141.
[5] The method of claim 1, wherein correlating the at least a portion of the plurality of first packets and the at least a	**Paxton**, in view of **Ivershen**, discloses the **correlating** based on **comparing network-interface identifiers**[11] (e.g., ports) **indicated by the at least a portion of the plurality of first packets and the at least a portion of the plurality of second packets**. *See* [2]. Akl, ¶¶142-144.

[11] *See* Ex. 1001, 4:47-51 (network-interface identifiers include identifiers of physical or logical ports).

’903 Patent	Paxton, Sutton, Ivershen
portion of the plurality of second packets comprises: comparing first network-interface identifiers indicated by the at least a portion of the plurality of first packets and second network-interface identifiers indicated by the at least a portion of the plurality of second packets.	
[6] The method of claim 1, wherein correlating the at least a portion of the plurality of first packets and the at least a portion of the plurality of second packets comprises: comparing one or more first	**Paxton** discloses **wherein correlating the at least a portion of the plurality of first packets and the at least a portion of the plurality of second packets comprises comparing one or more first [times] corresponding to the at least a portion of the plurality of first packets and one or more second times corresponding to the plurality of second packets** (correlating first and second packets based on closest timestamps). *See* [1g]. As discussed in §IX.A.5 above, it would have been obvious to a POSITA to modify **Paxton** to correlate first and second packets by comparing times (e.g., flow duration times) corresponding to or indicated by the first and second packets, as taught by **Iversheh**. **Iversheh** discloses:

’903 Patent	Paxton, Sutton, Ivershen
[times[12]] corresponding to the at least a portion of the plurality of first packets and one or more second times corresponding to the plurality of second packets.	• “Using a known checksum key from a packet on the first side of the NAT firewall, the checksum keys for all packets with a timestamp within a specified time on the second side of the NAT firewall can be analyzed. For example, **to account for packet transit time, firewall delay and clock errors, the checksum keys for all packets on the second side of the NAT firewall having a timestamp within milliseconds of the first-side packet are analyzed**.” (Iversheh, 2:43-51) • “**Monitoring system 108 can then pull together two or more legs of the session on demand. Starting with a first one of the legs,** such as the session flow and CHKEY created by probe 109 on interface 113, **the other probe 110 is queried with the** CHKEY, **flow starting timestamp**, and L7 protocol **from the first leg of the flow. The second probe 110, searches for a session that matches these parameters. The search on the second probe should look for a flow start timestamp that is within a few milliseconds of the beginning of the flow on the first probe. This allows for timestamp drift among the probes and network travel time across interfaces 113,114 and NATF/WAPG 106. If a match is found, two session flows are successfully correlated together into a single call record.** [¶] If more than one match is found, then false positives can be identified and discarded by comparing other properties of the flow, such as, for example, the **closest flow duration** or an exact

[12] Claim 6 of the ’903 patent recites “comparing one or more first corresponding,” not “comparing one or more first times corresponding.” No Certificate of Correction issued for the ’903 patent. Petitioner assumes this to be a typographical error and that this limitation ought to mean “first times,” as shown in analogous claim 15.

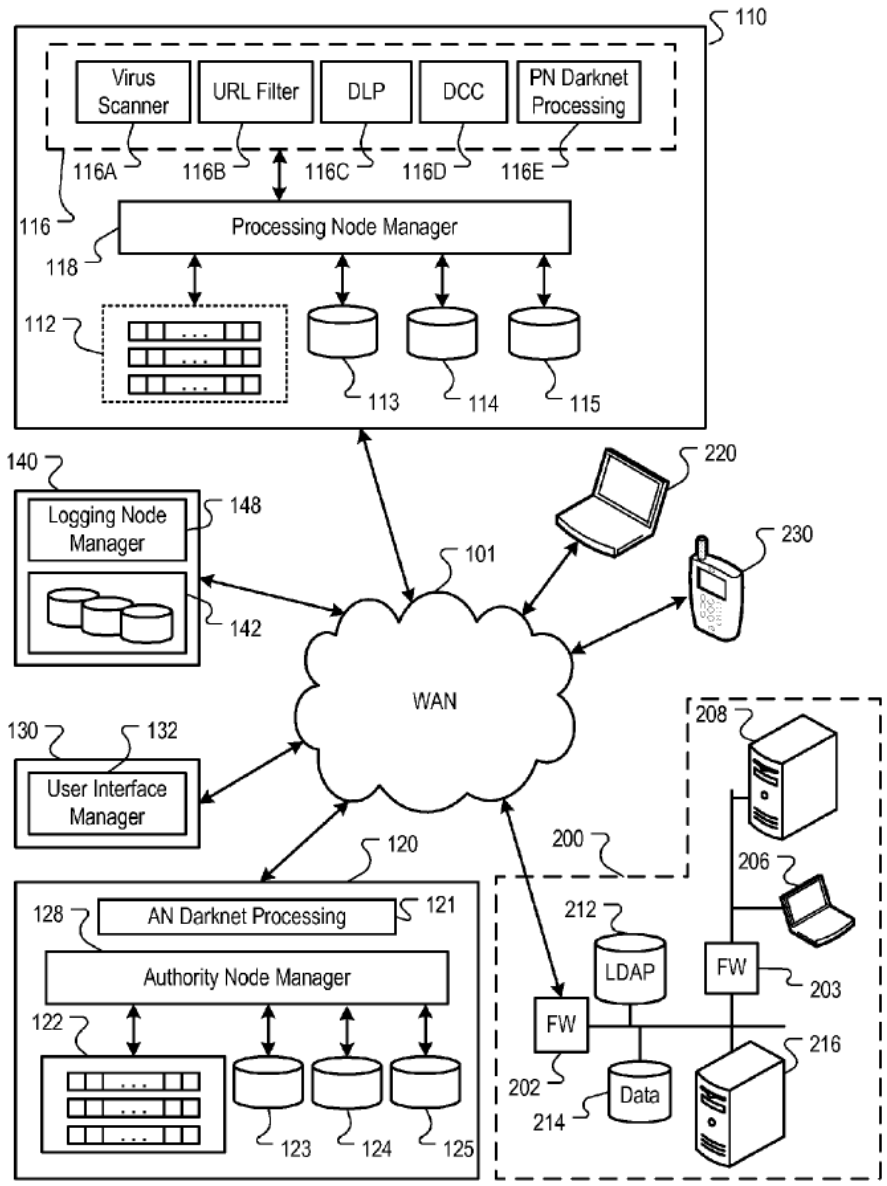
’903 Patent	Paxton, Sutton, Ivershen
	HTTP URI match.” (Ivershen, 5:62-6:11) • “In step 405, the checksum key for an IP flow on the first side of the NAT firewall is compared to the checksum keys for flows on the second side of the NAT firewall. **The flows that are used for comparison on the second side may be limited by using only <u>flows or packets occurring within a time window that is close to or similar to the timestamp</u> of the first-side packet.**” (Ivershen, 8:4-10) Akl, ¶¶145-147.
[7] The method of claim 1, wherein the network device comprises a gateway.	**Paxton** discloses **the network device comprising a gateway**[13] (e.g., routers, proxies, gateways, firewalls). • “**A boundary can include <u>routers, proxies, gateways, firewalls</u>**, and other types of computer network components.” (Paxton, ¶4) Akl, ¶¶148-150.
[8] The method of claim 1, wherein the second host is associated with a malicious entity, the method further comprising: generating, by the computing	**Paxton** discloses that the correlation can be used to identify nodes (hosts) infected with malicious content and identify the scope of malicious incidents. • “**The ability to <u>identify the true source of packet transmission through a boundary</u> can provide significant benefits to network security.** Current technology that attempts to discover the identity of network packet suffers from authentication and integrity problems. **It can provide a way to quickly <u>identify nodes that are infected with malicious content</u>, which can allow the network administrator to better identify the scope of the malicious incident.** The system and method

[13] *See* Ex. 1001, 5:51-52, 7:51-52, 10:66-11:3 (examples of gateways include bridges, intermediaries, VPNs, and tunneling gateways).

’903 Patent	Paxton, Sutton, Iversher
system, data configured to cause the first network to drop packets transmitted by the first host.	described herein can be highly modular and can be implemented atop open source technology on commodity hardware. Furthermore, it can provide a stable foundation for building tiered enterprise network architectures with an inherent capability for attribution of malicious activity. **Enterprises with significant visibility and monitoring investments into the network backbone can <u>utilize this technique to attribute malicious activity sensed at the edge of a network back to its original source</u>.**” (Paxton, ¶30) As explained in §IX.A.4 above, **Paxton** discloses correlating packets to identify malicious activity and leaves specific usage and remedial steps to a POSITA. A POSITA would have been motivated to determine whether **the second host is associated with a malicious entity and, if so generate data to cause the first network to drop packets transmitted by the first host**, as taught by **Sutton**, in **Paxton’s** system. **Sutton** discloses that **the second host** (outside the enterprise network) **is associated with a malicious entity** (associated with activities of a malicious actor; at a darknet address), and **generating, by the computing system, data/rules configured to cause the first network to drop packets transmitted by the first host** (causing malware identification programs to filter communications; using policy data to implement a rule to prevent all communication with malicious external host; blocking further/future communications; performed using, e.g., processing node manager 118 and data inspection engines 116 (e.g., URL filter 116B, PN darknet processing 116E) of processing node 110, a network device such as a server, gateway, or proxy). • “[T]he processing node 110 may act as a forward proxy that receives user requests to external servers addressed directly to the processing node 110.” (Sutton, 3:38-53; *see also* 4:45-51) • “The PN darknet processing 116E can also interrogate communications to **determine whether the communication is associated with (e.g., destined to or**

’903 Patent	Paxton, Sutton, Iversher
	originating from) an address in the darknet address database 115. The PN darknet processing 116E is described in detail at FIG. 3.” (Sutton, 6:18-22) • “§4.0 Monitoring Communications to Identify Potentially Malicious Activity Once the list of darknet addresses 115 is received from the authorization node(s) 120, the processing node(s) 110 can begin monitoring communications 350. In some implementations, **the processing node(s) 110 can inspect all communications 350 for inclusion of a destination address** that is included in the list of darknet addresses. In other implementations, **the processing node(s) 110 can inspect only those communications 350 originating from the enterprise network 200 to determine whether those communications 350 are destined for an address on the list of darknet addresses**. In further implementations, **the origin information of communications 350 can be inspected to identify communications 350 that purport to originate from the darknet address space**. Such communications 350 can be presumed to be non-legitimate as the source address has been spoofed (faked). **The determination that a destination or origination address is a darknet address can be made by comparing the destination and/or origination address on monitored communications 350 to the list of darknet addresses. If a match is found, the communication 350 is either destined to, or falsely originates from a darknet address.**” (Sutton, 10:37-59) • “In those implementations where all communications 350 are inspected, the communications 350 can be processed to **identify devices which are likely associated with malicious activity…. If such devices are outside of the enterprise network, the authority node policy data can be used to implement a rule preventing such devices from communicating with devices within the protected**

’903 Patent	Paxton, Sutton, Ivershen
	enterprise network.” (Sutton, 10:60-11:3) • “In those implementations where only those communications 350 originating from devices within a protected enterprise network are inspected, notification 355 can be provided to the enterprise network (e.g., a network administrator) indicating that **the device is potentially infected with malicious software code**…. In other implementations, **communications 350 originating from the device can be actively filtered (e.g., through various application specific malware identification programs, such as, performed by processing node manager 118 and data inspection engines 116) based upon identification of the probability that malicious code exists on the device**.” (Sutton, 11:4-18) • “At stage 450, **a notification of potential malicious activity originating from the protected network is provided and/or automated blocking/filtering is implemented.... Additionally, traffic may be automatically blocked, redirected or filtered based on predefined rules.** [¶] The various data exchange and malicious activity identification processes of FIG. 4 are example **processes for which the threat data and/or detection process filters can be updated in the system 100** of FIGS. 1 and 2. Other update processes, however, can also be used.” (Sutton, 12:57-13:14) • “**[B]locking further communications from a device originating communications** with a destination address on the list of darknet addresses and **taking appropriate additional steps to control future communications from the device** and provide notification of a potential infection.” (Sutton, 15:67-16:5)

’903 Patent	Paxton, Sutton, Iversheen
	• Fig. 2 (showing processing node 110 with processing node manager 118 and data inspection engines 116).  FIG. 2 • “Each processing node 110 can be implemented by a plurality of computer and communication devices, e.g., server computers, gateways, switches, etc…. In some implementations, each processing node 110 can include Internet gateways and a plurality of server computers, and the processing nodes 110 can be distributed through a

'903 Patent	Paxton, Sutton, Iversher
	geographic region, e.g., throughout a country." (Sutton, 3:24-37; *see also* 3:38-4:4 (distributing policy data to processing nodes), 13:26-42, 14:29-57) A POSITA would have recognized, based on **Sutton's** disclosure, that for a computing system to block future communications from a device (e.g., blocking a device within a network from communicating to an outside darknet address), when the device had not been previously identified as having malicious communications and/or not previously blocked, data/rules would have been generated and provisioned on a boundary or other monitoring device to identify those communications from the blocked device (i.e., a packet-filtering device; a device with access to the packet or tasked with inspecting the packet would have been provisioned with rules, thereby providing a way to identify the device's communications and accurately block undesired communications)—or at minimum found it obvious to do so. The generation of data/rules for boundaries to drop packets of specific devices was well-known in the art. Akl, ¶¶151-154.
[9] The method of claim 1, further comprising: generating, by the computing system, one or more rules configured to identify packets received from the first host located in the first network;	**Paxton**, in view of **Sutton**, discloses a computing system (the system including, e.g., a boundary and sensors, implementing methods taught by **Sutton**): **generating one or more rules configured to identify packets received from the first host located in the first network** (based on communications to a darknet address and identifying potential malicious code, filtering or blocking further communications from the host; preventing communications using rules; taking action to make such rules available); • *See* [8] above (*citing*, *e.g.*, Sutton, 10:37-11:18, 12:57-13:14). **provisioning a packet-filtering device in the first network with the one or more rules** (to block further communications, rules would have been provisioned on a device that analyzes

’903 Patent	Paxton, Sutton, Iversheen
provisioning, by the computing system, a packet-filtering device in the first network with the one or more rules; receiving, from the packet-filtering device, an indication of at least one packet received from the first host, wherein the at least one packet was identified by the packet-filtering device based on the one or more rules; and responsive to receiving the indication, dropping the at least one packet.	the communications, e.g., the sensors and servers already configured to analyze the packets, another gateway, proxy, firewall, etc. that is part of the first network; **Sutton** teaches that the processing nodes and functions for receiving, analyzing, and processing packets may be comprised of multiple devices with distributed hardware and software acting together); • *See* [8] above (*citing*, *e.g.*, Sutton, 3:24-37, 3:38-4:4). • “Systems, methods and apparatus for a distributed security that monitors communications to identify access attempts to/from darknet addresses. Such attempts can be inferred to be associated with malicious activity and a notification or **other corrective action can be provided identifying such potentially malicious activity.**” (Sutton, Abstract) **receiving, from the packet-filtering device, an indication of at least one packet received from the first host, wherein the at least one packet was identified by the packet-filtering device based on the one or more rules; and** **responsive to receiving the indication, dropping the at least one packet** (successfully dropping a packet based on rules includes determining, using the rules provisioned on the packet-filtering device, a packet that meets the rules, and based on an affirmative result from that determination, blocking the packet). • *See* [8] above (*citing*, *e.g.*, Sutton, Abstract, 10:37-11:18, 12:57-13:14, 15:67-16:5). • “**Each processing node 110 can generate a decision vector D=[d1, d2,…, dn] for a content item of one or more parts C=[c1, c2,…, cm].** Each decision vector can identify a threat classification, e.g., clean, spyware, malware, undesirable content, innocuous, unknown, etc. **For example, the output of each element of the decision vector D can be based on the output of one or more**

’903 Patent	Paxton, Sutton, Iversher
	data inspection engines. In some implementations, the threat classification can be reduced to a subset of categories e.g., violating, non-violating, neutral, unknown. Based on the subset classification, a processing node 110 may allow distribution of the content item, preclude distribution of the content item, allow distribution of the content item after a cleaning process, or perform threat detection on the content item. In some implementations, the actions taken by a processing node 110 can be determinative on the threat classification of the content item and on a security policy of the external system to which the content item is being sent from or from which the content item is being requested by. A content item is violating if, for any part C=[c1, c2,…, cm] of the content item, at any processing node 110, any one of the data inspection engines generates an output that results in a classification of ‘violating.’” (Sutton, 3:1-23) Akl, ¶¶155-161.
[10pre] An apparatus comprising:	**Paxton** discloses **an apparatus** (a computer system implementing modules). *See* [1a] and [10a]. Akl, ¶¶162-163.
[10a] at least one processor; and memory storing instructions that when executed by the at least one processor cause the apparatus to:	**Paxton** discloses **a plurality of processors** and **memory** (machine-readable media) **storing instructions for execution by the processors** to perform a method. • “In particular regard to the various functions performed by the above described **components (processor-executed processes, assemblies, devices, systems, circuits, and the like)**, the terms…used to describe such components are intended to correspond…to any component, such as **hardware, processor executed software, or combinations thereof**, which performs the specified

’903 Patent	Paxton, Sutton, Ivershen
	function….” (Paxton, ¶31) • “Portions of the invention can comprise a computer program that embodies the functions described herein. **Furthermore, the modules described herein, such as the inside sensor module, outside sensor module, and matching module, can be implemented in a computer system that comprises instructions stored in a machine-readable medium and a processor that executes the instructions.** However, it should be apparent that there could be many different ways of implementing the invention in computer programming, and the invention should not be construed as limited to any one set of computer program instructions.” (Paxton, ¶32) • “A computer implemented system, comprising: an inside sensor module…implemented in a computer system that comprises instructions stored in a machine-readable medium and a processor that executes the instructions; an outside sensor module…implemented in a computer system that comprises instructions stored in a machine-readable medium and a processor that executes the instructions; a matching module…implemented in a computer system that comprises instructions stored in a machine-readable medium and a processor that executes the instructions.” (Paxton, claim 10) Akl, ¶¶164-165.
[10b] determine that a network device has received, from a first host located in a first network, a	*See* [1a]. Akl, ¶166.

’903 Patent	Paxton, Sutton, Iversheen
plurality of first packets corresponding to first requests for content from a second host located in a second network,	
[10c] wherein the network device comprises a proxy;	*See* [1b]. Akl, ¶167.
[10d] determine that the network device has generated a plurality of second packets corresponding to second requests, wherein the second requests correspond to the first requests, and wherein the second requests are configured to cause the second host to transmit, to the network device, the content;	*See* [1c]. Akl, ¶168.

’903 Patent	Paxton, Sutton, Iversheh
[10e] generate a first plurality of log entries corresponding to the plurality of first packets, wherein each of the first plurality of log entries comprises a receipt timestamp indicating a packet receipt time, and wherein the first plurality of log entries comprise first data from the first requests;	*See* [1d]. Akl, ¶169.
[10f] generate a second plurality of log entries corresponding to the plurality of second packets, wherein each of the second plurality of log entries comprises a transmission	*See* [1e]. Akl, ¶170.

’903 Patent	Paxton, Sutton, Iversheen
timestamp indicating a packet transmission time, and wherein the second plurality of log entries comprise second data from the second requests;	
[10g] determine, for each transmission timestamp, differences between at least one packet transmission time indicated by transmission timestamps and at least one packet receipt time indicated by receipt timestamps;	*See* [1f]. Akl, ¶171.
[10h] correlate, based on the differences and by comparing the first data	*See* [1g]. Akl, ¶172.

’903 Patent	Paxton, Sutton, Iversheen
and the second data, at least a portion of the plurality of first packets and at least a portion of the plurality of second packets; and	
[10i] responsive to the correlating: generate an indication of the first host; and transmit the [indication] of the first host.	*See* [1h]. Akl, ¶173.
[11] The apparatus of claim 10, wherein the instructions, when executed by the at least one processor, further cause the apparatus to correlate the at least a portion of the plurality of first packets and the at least a portion of the plurality of second packets further based	*See* [2]. Akl, ¶174.

’903 Patent	Paxton, Sutton, Iversheh
on a comparison of one or more first ports indicated by the at least a portion of the plurality of first packets and one or more second ports indicated by the at least a portion of the plurality of second packets.	
[12] The apparatus of claim 10, wherein the instructions, when executed by the at least one processor, cause the apparatus to correlate the at least a portion of the plurality of first packets and the at least a portion of the plurality of second packets further based on a comparison of one or more	*See* [3]. Akl, ¶175.

’903 Patent	Paxton, Sutton, Iversheen
first protocol types indicated by the at least a portion of the plurality of first packets and one or more second protocol types indicated by the at least a portion of the plurality of second packets.	
[13] The apparatus of claim 10, wherein the instructions, when executed by the at least one processor, cause the apparatus to correlate the at least a portion of the plurality of first packets with the at least a portion of the plurality of second packets further based on a comparison of first application-layer data	*See* [4]. Akl, ¶176.

’903 Patent	Paxton, Sutton, Iversheen
indicated by the at least a portion of the plurality of first packets and second application-layer data indicated by the at least a portion of the plurality of second packets.	
[14] The apparatus of claim 10, wherein the instructions, when executed by the at least one processor, cause the apparatus to correlate the at least a portion of the plurality of first packets and the at least a portion of the plurality of second packets further based on a comparison of first network-interface identifiers indicated by	*See* [5]. Akl, ¶177.

’903 Patent	Paxton, Sutton, Ivershen
the at least a portion of the plurality of first packets and second network-interface identifiers indicated by the at least a portion of the plurality of second packets.	
[15] The apparatus of claim 10, wherein the instructions, when executed by the at least one processor, cause the apparatus to correlate the at least a portion of the plurality of first packets and the at least a portion of the plurality of second packets further based on a comparison of one or more first times indicated by the at least a	*See* [6]. Akl, ¶178.

’903 Patent	Paxton, Sutton, Ivershen
portion of the plurality of first packets and one or more second times indicated by the at least a portion of the plurality of second packets.	
[16] The apparatus of claim 10, wherein the network device comprises a gateway, and wherein the network device is configured to generate the plurality of second packets by encapsulating data received in the first requests.	**Paxton** discloses **the network device comprising a gateway**. • *See* [7]. **Paxton** discloses **the network device generating the plurality of second packets by encapsulating data received in the first requests** (NAT functions by altering the IP address and port information in the header of information passing through it; the header “encapsulates” the data). • “When NAT is implemented, the source address of a packet changes from the original sender of the packet to the address of the boundary performing NAT.” (Paxton, ¶3) • “NAT is typically performed on boundaries that sit in the path of communication between a client and server. Boundaries can intercept and relay the client's request to the server as well as the server's response to the client. Therefore, while client requests are sourced from the client, boundary requests alter the original client requests to appear from the boundary. Likewise, server responses are addressed to the boundary, whereas boundary responses are altered to appear addressed directly to the client. The boundary alters the source IP address, the source application ports and their associated checksums within each packet header.” (Paxton, ¶4) A POSITA would have recognized that performing NAT encapsulates (e.g., repackages) the data.

’903 Patent	Paxton, Sutton, Iversheen
	Akl, ¶¶179-182, 42.
[17] The apparatus of claim 10, wherein the second host is associated with a malicious entity, and wherein the instructions, when executed by the at least one processor, cause the apparatus to generate data configured to cause the first network to drop packets transmitted by the first host.	*See* [8]. Akl, ¶183.
[18] The apparatus of claim 10, wherein the instructions, when executed by the at least one processor, cause the apparatus to: generate one or more rules configured to identify	*See* [9]. Akl, ¶184.

’903 Patent	Paxton, Sutton, Iversheen
packets received from the first host; and configure a packet-filtering device to: identify, based on the one or more rules, at least one packet received from the first host; and responsive to identifying the at least one packet, drop the at least one packet.	

X. SECONDARY CONSIDERATIONS

Petitioner is unaware of any secondary considerations relevant to the Challenged Claims. Akl, ¶186.

XI. CONCLUSION

Substantial, new, and noncumulative technical teachings have been presented for each Challenged Claim, which are disclosed and/or rendered obvious for the reasons set forth above. There is a reasonable likelihood that Petitioner will prevail as to each of these Challenged Claims. *Inter partes* review of claims 1-18 of the '903 patent is accordingly requested.

Respectfully submitted,

Dated: July 20, 2021

By: /Scott A. McKeown/
Scott A. McKeown
Registration No. 42,866

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 13,593 words as counted by the word processing program used to prepare the paper.

Dated: July 20, 2021

By: /Scott A. McKeown/
Scott A. McKeown
Registration No. 42,866

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 Fedex of a copy of this Petition for Inter Partes Review and supporting materials at the correspondence address of record for the '903 patent:

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

Dated: July 20, 2021

By: /Scott A. McKeown/
Scott A. McKeown
Registration No. 42,866