UNITED STATES DISTRICT COURT
FOR THE EASTERN DISTRICT OF TEXAS
MARSHALL DIVISION

IN RE: TAASERA LICENSING LLC, PATENT LITIGATION **THIS DOCUMENT RELATES TO CASE NOS. 2:23-CV-00113-JRG**	**CIVIL ACTION NO. 2:22-MD-03042-JRG**
TAASERA LICENSING LLC **Plaintiff,** **v.** **PALO ALTO NETWORKS, INC.** **Defendant.**	**CASE NO. 2:23-CV-00113-JRG**

DEFENDANT PALO ALTO NETWORK, INC.'S PATENT L.R. 3-3 INVALIDITY CONTENTIONS AND SUBJECT MATTER INELIGIBILITY CONTENTIONS

TABLE OF CONTENTS

I. INTRODUCTION 1

II. PRIORITY DATE 6

III. PATENT L.R. 3-3(A) 6

A. Prior Art Patents, Applications and Printed Publications 7

B. Prior Art Apparatuses and Systems 33

C. Applicants Admitted Prior Art (AAPA) 36

D. Prior Art Under 35 U.S.C. § 102(f) 78

E. Prior Art Under 35 U.S.C. § 102(g) 78

IV. PATENT L.R. 3-3(B) 78

A. Anticipation 79

B. Obviousness 87

C. Obviousness Combinations and Exemplary Motivations to Combine 95

V. PATENT L.R. 3-3(C) 352

VI. PATENT L.R. 3-3(D) 353

A. Grounds of Invalidity Under 35 U.S.C. § 112 353

B. Grounds of Invalidity Under 35 U.S.C. § 101 393

VII. DOCUMENT PRODUCTION ACCOMPANYING INVALIDITY AND SUBJECT MATTER ELIGIBILITY CONTENTIONS 485

Pursuant to Patent Local Rule 3-3, Defendant Palo Alto Networks, Inc. ("PAN" or "Defendant") hereby serves these Invalidity Contentions on Plaintiff Taasera Licensing LLC ("Taasera" or "Plaintiff"). These Invalidity Contentions address only claims 1, 2, 7, 10, 12, 13, 18, 21, and 23–25 of U.S. Patent No. 6,842,796 ("the '796 Patent"); claims 1-4, 6-14, 16-18, 22, 24-25, and 29 of U.S. Patent No. 7,673,137 ("the '137 Patent"); claims 1-4, 6, 8-10, and 12-20 of U.S. Patent No. 8,127,356 ("the '356 Patent"); claims 1-7 of U.S. Patent No. 8,327,441 ("the '441 Patent"); claims 1-5, 12-17 and 24 of U.S. Patent No. 8,850,517 ("the '517 Patent"); claims 1-33 of U.S. Patent No. 8,955,038 ("the '038 Patent"); claims 1-3 and 5 of U.S. Patent No. 8,990,948 ("the '948 Patent"); claims 1-7 of U.S. Patent No. 9,092,616 ("the '616 Patent"); and claims 1-24 of U.S. Patent No. 9,923,918 ("the '918 Patent") (collectively the "Asserted Claims" and "Patents-in-Suit"), as asserted in Plaintiff's Initial Disclosure of Asserted Claims and Infringement Contentions served on March 28, 2023 to PAN.

I. INTRODUCTION

Defendant has not completed its investigation of the facts and documents relating to this action and has not completed its preparation for trial. Defendant has not taken any depositions in this action, including, without limitation, any depositions of the named inventors of the Patents-in-Suit and/or other persons having potentially relevant information. As discovery in this action provides Defendant with additional information, it is possible that Defendant will discover additional prior art pertinent to the invalidity of the Asserted Claims of the Patents-in-Suit, and Defendant reserves the right to supplement these contentions after becoming aware of additional prior art or information. Defendant reserves the right to rely on any invalidity position and any prior art reference included in the invalidity contentions of any present or former defendant in Plaintiff's lawsuits that Plaintiff did not timely produce pursuant to Defendant's discovery requests. Defendant further reserves the right to introduce and use such supplemental materials at trial.

There has been no claim construction ruling in this case, nor an exchange of proposed constructions. Accordingly, Defendant's Invalidity Contentions reflect the presumed readings of the claims advanced by Plaintiff in its Infringement Contentions and may reflect alternative readings.[1] These Invalidity Contentions, including as to anticipation, obviousness, Section 112, Section 101, the accompanying claim charts and any referenced IPR proceedings, are not admissions or adoptions as to any particular claim scope or construction nor an admission that any particular element is met in any particular way in Defendant's Accused Instrumentalities, nor an admission or adoption as to any particular reading of the intrinsic record. Nothing herein should be treated as an admission that Defendant agrees with Plaintiff's express or implied claim constructions or that Plaintiff has proposed any discernible constructions for claims and/or claim terms in its deficient Infringement Contentions. To the extent that Defendant understands Plaintiff's allegations of infringement, Plaintiff attempts to stretch the language of the Asserted Claims beyond the scope to which the claims could reasonably be entitled in light of the disclosures of the Patents-in-Suit and the prosecution histories. Moreover, as discussed further below, certain terms are indefinite and are therefore invalid. Although Defendant has attempted to map the terms to the prior art, insofar as the Court decides that they are not indefinite, Defendant may amend these contentions to point to additional disclosure of whatever the Court identifies the relevant construction.

Defendant further reserves the right to revise these Invalidity Contentions in view of the Court's construction of terms and phrases recited in one or more of the Asserted Claims, additional information obtained during discovery, additional infringement theories put forth by Plaintiff

[1] Defendant does not concede that Plaintiff's constructions are correct, but rather assert the well-established principle that whatever infringes a claim if later in time anticipates if earlier in time. *Bristol-Myers Squibb Co. v. Ben Venue Labs., Inc.*, 246 F.3d 1368, 1378 (Fed. Cir. 2001). Thus, where Plaintiff for purposes of its infringement case alleges that a feature of an accused product meets a particular limitation recited in one or more of the Asserted Claims, then that feature, should it be found in the prior art, would also cause that limitation to be met for invalidity purposes.

during fact and/or expert discovery, any findings as to the priority date(s) of the Asserted Claims, and/or positions that Plaintiff, its fact witnesses, or its expert witness(es) may take concerning claim construction, infringement, and/or invalidity issues.

Defendant contends that the Asserted Claims are invalid under 35 U.S.C. § 102 and/or § 103 as being anticipated by or rendered obvious by prior art disclosed herein and/or in view of the knowledge of one of ordinary skill in the art. Prior art not included in this disclosure, whether or not now known to Defendant, may become relevant depending on the claim constructions ultimately adopted by the Court. Defendant's ongoing investigation may also uncover additional prior art. Any obviousness combinations of references provided herein pursuant to 35 U.S.C. § 103 are not intended to be exhaustive. Additional obviousness combinations of the references identified below are possible, and Defendant reserves the right to use any such combinations in this litigation. Defendant further reserves the right to supplement their accompanying P.R. 3-4(b) document production should it later discover additional prior art documents, information, testimony, prior art systems and related documentation, and/or software or hardware code, including information provided by third parties after the date of service of these Invalidity Contentions.

In addition, Defendant contends that certain Asserted Claims do not satisfy one or more requirements of 35 U.S.C. §§ 101 and/or 112. In order to fulfill its obligations pursuant Patent L.R. 3-3, however, Defendant has applied the prior art based on an assumption that Plaintiff contends all Asserted Claims satisfy all of the applicable requirements of 35 U.S.C. §§ 101 and/or 112. The application of prior art in these Invalidity Contentions should not be construed as an admission that Defendant agrees that any of the Asserted Claims satisfies all the requirements of 35 U.S.C. §§ 101 and 112.

Defendant may further rely on inventor admissions concerning the scope or state of the prior art relevant to the Asserted Claims, the patent prosecution histories of the Asserted Patents, related patents and/or patent applications, any deposition or trial testimony of the named inventor on the Asserted Patents, and the papers filed and any evidence produced or submitted by Plaintiff

in connection with this case or other related litigation. Defendant reserves the right to contend that one or more of the Asserted Claims are invalid under 35 U.S.C. § 102(f) in the event Defendant obtains evidence that the named inventor(s) did not invent the subject matter in the Asserted Claims.

Defendant intends these disclosures to serve as a first notice under 35 U.S.C. § 282 (c).

Defendant does not believe Plaintiff's Infringement Contentions comply with Patent L.R. 3.1, which has compromised Defendant's ability to prepare its invalidity contentions. As a result, Defendant has elected the prior art references disclosed in these invalidity contentions without an adequate understanding of Plaintiff's infringement positions. To the extent Plaintiff supplements its infringement contentions, Defendant reserves their right to update their invalidity contentions accordingly. Defendant reserves the right to rely on any invalidity position and any prior art reference included in the invalidity contentions of any present or former defendant in Plaintiff's lawsuits and by any potential or actual licensees to the Patents-in-Suit; and any contentions in reexamination or post-grant proceedings (e.g., IPR) involving the Patents-in-Suit or any patents that claim priority to, claim priority from, or are the basis for a claim of priority in the Patents-in-Suit (collectively, the "Related Patents"). Defendant also reserves the right to rely upon any reference cited during the prosecution history or *inter partes* review proceeding, including the prosecution history and any proceeding against patents related to the Asserted Patents. Defendant reserves the right to rely on that prior art, including in a similar manner that the reference was used in either the prosecution history or *inter partes* review proceeding. These proceedings include at least the following *inter partes* review proceedings, and any *inter partes* review proceedings filed by Defendant or any third party subsequent to service of these invalidity contentions: IPR2023-00443 against the '796 Patent; IPR2023-00801 against the '441 Patent; IPR2023-00573 against the '038 Patent; IPR2023-00574 against the '918 Patent; IPR2023-00704 and IPR2023-798 against the '517 Patent; and IPR2023-00706 and IPR2023-00799 against the '616 Patent.

Because Defendant's investigation, prior art search, and analysis are still ongoing, it is likely that Defendant will identify additional prior art or contentions that will add meaning to already known prior art or contentions or possibly lead to additions or changes to these Invalidity Contentions. Without obligating itself to do so, Defendant reserves all rights to amend, modify, or supplement these Invalidity Contentions. Defendant further reserves the right to rely on any facts, documents, or other evidence that are: (i) subsequently discovered; (ii) subsequently determined to be relevant for any purpose; or (iii) subsequently determined to have been omitted from a production, whether inadvertently or otherwise. Defendant further reserves the right to rely on expert testimony. Documents related to expert testimony, if any, will be produced when expert discovery is exchanged pursuant to the Court's Docket Control Order and Discovery Order, to be entered in this case.

The references identified in these Invalidity Contentions, which include the attached claim charts, may disclose the elements of the Asserted Claims explicitly and/or inherently, and/or they may be relied upon to show the state of the art in the relevant time frame. References identified in these Invalidity Contentions, as well as the "References Cited" on the face of the Asserted Patents and the patents or technical references cited within the body of the Asserted Patents, may be used to illustrate, but not limit the scope of, the state of the art to which the Asserted Patents pertain (*i.e.*, at a time prior to the date of alleged inventions of the Asserted Claims of the Asserted Patents). Moreover, Defendant reserves the ability to rely on later identified sources of information, including but not limited to witness testimony and other discovery, to establish the state of the art in the relevant time frame pertaining to the Asserted Patents. Defendant's claim charts cite to particular teachings and disclosures of the prior art as applied to features of the Asserted Claims. However, persons having ordinary skill in the art generally may view an item of prior art in the context of other publications, literature, products, and understanding. As such, the cited portions are only examples, and Defendant reserves the right to rely on uncited portions of the prior art

references and on other publications and expert testimony, including any underlying systems described in the prior art references, references cited therein, or different versions of same prior art references or documents describing same or related systems, as aids in understanding and interpreting the cited portions, as providing context thereto, and as additional evidence that the prior art discloses a claim limitation. Citations to a figure in a reference should be understood to include the text that relates to that figure, and citations to text that relates to a figure should be understood to include the figure. Defendant further reserves the right to rely on uncited portions of the prior art references, other publications, the state of the art, and testimony to establish bases for combinations of certain cited references that render the Asserted Claims obvious.

The foregoing statements and reservations of rights are hereby expressly incorporated by reference in their entirety into each of the disclosures below, into the invalidity charts served herewith as Appendices, and into each disclosure corresponding to each element of every claim.

II. PRIORITY DATE

While Plaintiff has alleged in its Infringement Contentions that the Asserted Claims are entitled to certain priority dates earlier than the filing date of the application in some cases, Defendant does not concede that any Asserted Claim is entitled to a priority date earlier than the filing date of the application from which that Asserted Claim directly issued.

III. PATENT L.R. 3-3(A)

In the following subsections, Defendant identifies each item of prior art that Defendant alleges anticipates one or more of the Asserted Claims, renders them obvious, and/or forms part of the state of the art that was well understood to those of skill in the art as of the respective priority dates of the Asserted Patents. Each of the following patents, publications, systems, and products is prior art under at least one or more of 35 U.S.C. § 102(a), (b), (e), (f), or (g) (pre-AIA) or § 102(a) (post-AIA), as applicable. These prior art patents, publications, systems, and products disclose the elements of the Asserted Claims either explicitly, implicitly, inherently, or via an obvious combination, and/or are relied upon to show the state of the art in the relevant timeframes.

Defendant further reserves the right to rely upon the following to show the state of the art in the relevant timeframes: all prior art cited on the face of the Patents-in-Suit and Related Patents, the admitted prior art references[2] in the specification of the Patents-in-Suit and Related Patents, all prior art cited in the prosecution histories of the Patents-in-Suit and Related Patents, including in any reexaminations, reissue, or *inter partes* review proceedings, and the references cited in any invalidity contentions submitted in any action or proceedings involving the Patents-in-Suit or Related Patents.

To the extent that Plaintiff challenges a prior art patent's qualification as prior art, Defendant reserves the right to rely upon: (i) foreign counterparts of U.S. patents identified in these Invalidity Contentions; and (ii) U.S. counterparts of foreign patents and foreign patent applications identified in these Invalidity Contentions, to the extent such references qualify as prior art and contain the same substantive disclosure.

A. Prior Art Patents, Applications and Printed Publications

Defendant has identified each prior art patent or application by its number, country of origin, and date of issue or, if an application, date of publication. Unless otherwise noted, the country of origin for the prior art patent or patent application is the U.S. To the extent feasible, Defendant has identified each prior art publication by its title, date of publication, author, and publisher. The dates provided are not necessarily the earliest date and should be understood accordingly (*e.g.*, the date of issue/publication is at least as early as the date cited below, as with date of publication, filing date, etc.).

1. '796 Patent

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
1	US 6,169,999 to Kanno	1/2/01	4/14/98	Kanno

[2] The applicants admitted prior art (AAPA) of the Patents-in-Suit include the systems and methods described in the "Background" sections of the Patents-in-Suit. *See* examples in Section III.C below.

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
2	US 5,317,509 to Caldwell	5/31/94	1/21/92	Caldwell
3	US 6,212,494 to Boguraev	4/3/01	7/20/98	Boguraev
4	US 5,715,468 to Budzinki	2/3/98	9/30/94	Budzinki
5	EP 886226(A1) to Julliard	12/23/98	6/19/98	Julliard I
6	EP 1043711(A2) to Karaorman *et al.*	10/11/00	3/8/00	Karaorman
7	US 5,371,807 to Register *et al.*	12/6/94	3/20/92	Register
8	US 5,826,258 to Gupta *et al.*	10/20/98	10/2/96	Gupta
9	US 5,963,965 to Vogel	10/5/99	2/18/97	Vogel
10	US 6,182,029 to Friedman	1/30/01	8/6/99	Friedman
11	US 6,202,064 to Julliard	3/13/01	6/18/98	Julliard II
12	US 6,269,189 to Chanod	7/31/01	12/29/98	Chanod
13	US 6,278,996 to Richardson *et al.*	8/21/01	3/30/98	Richardson
14	US 6,505,157 to Elworth	1/7/03	2/23/00	Elworth
15	US 7,581,170 to Baumgartner *et al.*	8/25/09	5/28/02	Baumgartner I
16	Friedl, Jeffrey E.F., Mastering Regular Expressions: Powerful Techniques for Perl and Other Tools, O'Reilly & Associates, Inc. 1997, seventh printing edition December 1998	12/1998	N/A	Friedl
17	Thompson, Ken, Regular Expression Search Algorithm, Bell Telephone Laboratories, Inc., Communications of the ACM, Vol. 11, No. 7, June 1968	6/1968	N/A	Thompson
18	Hosoya, H, Regular Expression Pattern Matching for XML, Dept. of Computer and Information Science – Univ. of Pennsylvania, Nov. 8, 2000	11/8/00	N/A	Hosoya
19	Baumgartner *et al.*, Robert, Visual Web Information Extraction with Lixto, Proceedings of the 27th VLDB Conference, Roma, Italy, 2001	2001	N/A	Baumgartner II

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
20	US 6,311,194 to Sheth *et al.*	10/30/01	3/15/00	Sheth
22	US 5,913,214 to Madnick *et al.*	6/15/99	5/30/96	Madnick
23	US 6,507,643 to Groner	1/14/03	3/16/00	Groner
24	U.S. Pat. App. Pub. 2005/0171932 to Nandhra	8/4/05	2/24/00	Nandhra
25	US 7,073,122 to Sedghi	7/4/06	9/8/00	Sedghi
26	US 7,584,120 to Yun *et al.*	9/1/09	4/7/99	Yun
27	US 7,974,875 to Quilici *et al.*	7/5/11	3/21/00	Quilici
28	US 6,373,985 to Hu *et al.*	4/16/02	8/12/98	Hu
29	US 6,714,939 to Saldanha *et al.*	3/30/04	1/8/01	Saldanha
30	U.S. Pat. App. Pub. 2002/0194223 to Meyers *et al.*	12/19/02	10/16/00	Meyers
31	U.S. Pat. App. Pub. 2005/0129188 to Lee *et al.*	6/16/05	6/3/99	Lee
32	Extraction and Integeration of Data from Semi-Structured Documents into Business Applicatoins (Ph. Bonnet & S. Bressan)	9/1997	NA	Bonnet
33	U.S. 6,678,681 to Brin	1/13/05	3/9/00	Brin
34	U.S. 5,317,509 to Caldwell	5/31/94	1/21/92	Caldwell
35	U.S. 5,685,000 to Cox, Jr.	11/4/97	1/4/95	Cox
36	Ontology-Based Extractoin and Structuring of Information from Data-Rich Unstructured Documents (D. Wembley, et al)	1998	NA	Embly
37	U.S. 6,892,237 to Gai, et al	5/10/05	3/28/00	Gai
38	U.S. 4,866,755 to Hasimoto	9/12/89	9/12/88	Hashimoto
39	U.S. 6,915,254 to Heinze, et al	7/5/05	7/30/99	Heinze
40	U.S. 6,373,985 to Hu, et al	4/16/02	8/12/98	Hu

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
41	U.S. 6,856,988 to Humphrey,et al	2/15/05	12/21/99	Humphrey
42	JP 08314490 to Toru, et al	5/23/95		Toru
43	U.S. 6,202,064 to Julliard	3/13/01	6/18/98	Julliard
44	Technical Terminology: some linguistic properties and an algorithm for identification in text (J. Justeson and S. Katz)	3/1995	NA	Justeson
45	U.S. 7,069,501 to Kunitake, et al	6/27/06	1/22/01	Kunitake
46	U.S. Pat. App. Pub. 2005/0129188 to Lee, et al	6/16/05	2/2/05	Lee
47	U.S. Pat. App. Pub. 2002/0156817 to Lemus	10/24/02	2/21/02	Lemus
48	U.S. 5,913,214 to Madnick, et al	6/15/99	8/8/96	Madnick
49	The Rhetorical Parsing of Natural Language Texts (D. Marcu)	7/7/97	NA	Marcu
50	WO 00/62194 to McKoewn, et al	10/19/00	4/12/00	McKoewn
51	U.S. Pat. App. Pub. 2002/194223 to Meyers, et al	12/19/02	10/16/01	Meyers
52	U.S. Provisional 60/241009 to Meyers	NA	10/16/00	Meyers II
53	U.S. Pat. App. Pub. 2005/0171932 to Nandhra	8/4/05	2/23/01	Nandhra
54	U.S. 7,167,825 to Pötter	1/27/07	3/10/00	Pötter
55	U.S. 7,974,875 to Quilici, et al	7/5/11	3/21/00	Quilici
56	U.S. 6,278,996 to Richardson, et al	8/21/01	3/20/98	Richardson
57	U.S. 6,714,939 to Saldanha, et al	3/30/04	1/8/11	Saldanha
58	Tutorial: Perl, a psychologically efficient reformatting language (Alan Schwartz)	1998	NA	Schwartz
59	U.S. 7,073,122 to Sedghi	7/4/06	9/8/00	Sedghi

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
60	U.S. 6,311,194 to Sheth, et al	10/30/01	8/21/00	Sheth
61	Textual Data Mining of Service Center Call Records (P. Tan, et al)	8/1/00	NA	Tan
62	U.S. 6,167,368 to Wacholder	12/26/00	8/14/98	Wacholder
63	U.S. 6,279,017 to Walker	8/21/01	2/2/98	Walker
64	Segmentation of Expository Texts by Hierarchical Agglomerative Clustering (Y. Yaari)	9/26/97	NA	Yaari
65	U.S. 7,584,120 to Yun, et al	9/1/09	4/7/1999	Yun

2. ’137 Patent

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
1	US 6,418,444 to Raduchel	7/9/02	12/11/97	Raduchel
2	US 6,247,171 to Yellin *et al.*	6/12/01	12/6/99	Yellin
3	US 6,618,769 to Bracha *et al.*	9/9/03	5/27/99	Bracha
4	US 6,643,842 to Angel *et al.*	11/4/03	9/12/01	Angel
5	US 7,093,239 to van der Made	8/15/06	8/18/00	Van der Made
6	US 6,658,571 to O’Brien *et al.*	12/2/03	2/9/99	O’Brien
7	Zwienenberg, "Heuristic Scanners: Artificial Intelligence?"	Sept 20-22, 1995	N/A	Zwienenberg
8	US 7,058,822 to Edery *et al.*	6/6/06	3/30/00	Edery
9	US 5,412,717 to Fischer	5/2/95	5/15/92	Fischer
10	Dan *et al.*, "ChakraVyuha (CV): A Sandbox Operating System	2/20/97	N/A	ChakraVyuha [3]

[3] This reference relates to the ChakraVyuha system and is evidence of that prior art system’s functionality, as well as its availability and that it was on sale and/or offered for sale. Moreover, this reference qualifies independently as prior art under at least 35 U.S.C. §§ 102(a), (b), and (g).

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
	Environment for Controlled Execution of Alien Code”			
11	US 5,825,877 to Dan *et al.*	10/20/98	6/11/96	Dan
12	US 5,944,821 to Angelo	8/31/99	7/11/96	Angelo
13	US 6,253,374 to Dresevic *et al.*	6/26/01	7/2/98	Dresevic
14	US 7,350,204 to Lambert *et al.*	3/25/08	7/24/00	Lambert
15	US 5,958,010 to Agarwal *et al.*	9/28/99	3/20/97	Agarwal
16	US 6,507,909 to Zurko *et al.*	1/14/03	2/13/90	Zurko
17	US 3,996,449 to Attanasio	12/7/76	8/25/75	Attanasio
18	US 7,350,204 to Lambert	7/25/02	6/08/01	Lambert
19	U.S. 4,907,150 to Arroyo, et al	3/6/90	7/26/88	Arroyo
20	U.S. 4,926,476 to Covey	5/15/90	2/3/89	Covey
21	Using PAPI for hardware performance monitoring on Linux systems (D. Dongarra)	8/2009	NA	Dongarra
22	U.S. 6,507,904 to Ellison, et al	1/14/03	3/31/00	Ellison
23	U.S. 6,651,171 eto England, et al	11/18/03	4/6/99	England
24	WO 00/72149 to Gieger, et al	11/30/00	4/28/00	Gieger
25	U.S. 6,823,460 to Hollander, et al	11/23/04	4/28/00	Hollander
26	WO 01/37095 to Hollnader, et al	5/1/02	11/10/11	Hollander II
27	TrustedBox: a Kernel-Level Integrity Checker (P. Iglio)	2/1999	NA	Iglio
28	U.S. Pat. App. Pub 2002/0188649 to Karim	12/12/02	6/12/01	Karim
29	U.S. 7,033,775 to Lacombe, et al	2/21/06	8/17/01	Lacombe
30	U.S. 6,934,857 to Bartleson, et al	8/23/05	11/27/00	Bartleson
31	U.S. 7,085,928 to Schmid, et al	8/1/06	3/30/01	Schmid

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
32	U.S. 6,651,186 to Schwabe	11/18/03	9/14/00	Schwabe
33	U.S. 6,308,275 to Vaswani, et al	10/23/01	7/10/98	Vaswani
34	EP 0,047,060 to Wehr	1/9/91	6/20/90	Wehr

3. '441 Patent

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
1	US 9,298,922 to Berger *et al.*	3/29/16	7/10/08	Berger
2	U.S. Pat. App. Pub. 2007/0240222 to Tuvell *et al.*	10/11/07	4/6/07	Tuvell
3	US 9,176,754 to Kokkevis *et al.*	11/3/15	11/7/08	Kokkevis
4	US 7,904,278 to Wilson *et al.*	3/8/11	5/2/07	Wilson
5	US 8,429,412 to Bleckmann *et al.*	4/23/13	12/8/10	Bleckmann
6	US 7,930,733 to Iftode *et al.*	4/19/11	4/10/06	Iftode
7	US 7,490,352 to Kramer *et al.*	2/10/09	4/7/05	Kramer
8	U.S. Pat. App. Pub. 2007/0209060 to Rhodes	9/6/07	7/24/06	Rhodes
9	U.S. Pat. App. Pub. 2010/0281248 to Lockhart *et al.*	11/4/10	6/21/10	Lockhart
10	US 8,701,198 to Greene *et al.*	4/15/14	8/10/10	Greene
11	US 8,832,829 to Manni *et al.*	9/9/14	9/30/09	Manni
12	US 9,268,945 to Williams *et al.*	2/23/16	2/11/14	Williams I
13	U.S. Pat. App. Pub. 2007/0288904 to Newcomb	12/13/07	5/16/06	Newcomb
14	U.S. Pat. App. Pub. 2012/0260344 to Maor *et al.*	10/11/12	12/14/10	Maor
15	U.S. Pat. App. Pub. 2014/0165204 to Williams *et al.*	6/12/14	2/11/14	Williams II

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
16	U.S. Pat. App. Pub. 2015/0195181 to Birmiwal	7/9/15	9/30/10	Birmiwal
17	US 7,788,729 to Cameron *et al.*	8/31/10	3/4/05	Cameron
18	US 8,397,300 to Tripp	3/12/13	9/22/09	Tripp
19	US 8,880,667 to Novak *et al.*	11/4/14	2/9/11	Novak
20	US 9,087,196 to Wood *et al.*	7/21/15	12/24/10	Wood
21	US 9,218,461 to Friedrichs *et al.*	12/22/15	11/30/11	Friedrichs
22	US 9,832,651 to Korkishko *et al.*	11/28/17	4/16/10	Korkishko
23	US 9,979,707 to Miller *et al.*	5/22/18	6/28/17	Miller
24	U.S. Pat. App. Pub. 2011/0119765 to Hering *et al.*	5/19/2011	11/18/09	Hering
25	U.S. Pat. App. Pub. 2012/0246470 to Nicolson *et al.*	9/27/12	1/27/11	Nicolson
26	U.S. Pat. App. Pub. 2012/0324214 to Asokan *et al.*	12/20/12	2/16/11	Asokan
27	US Pat. App. Pub. 2011/0179477 to Starnes *et al.*	7/21/11	12/9/05	Starnes
28	US 7,302,677 to Reissman *et al.*	11/27/07	5/8/03	Reissman
29	EP 1,296,255 to Matsumoto *et al.*	3/26/03	5/1/00	Matsumoto
30	WO 2006/024216 to Luo	3/9/06	7/29/04	Luo
31	US 7,739,663 to Newcomb	6/15/10	5/16/06	Newcomb
32	US 7,788,640 to Grimaldi	8/31/10	12/28/04	Grimaldi
33	WO 2007/104988 to Curnyn	9/20/07	3/14/06	Curnyn
34	Kil, C., Sezer, E. C., Azab, A. M., Ning, P., & Zhang, X. Remote attestation to dynamic system properties: Towards providing complete system integrity evidence.	June 2009	N/A	Kil
35	US 7,610,273 to Kuppusamy *et al.*	10/27/09	3/22/05	Kuppusamy

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
36	US 8,141,154 to Guzman	9/30/10	6/14/10	Guzman
37	US 8,474,004 to Leone	10/8/09	7/31/06	Leone
38	US 8,881,257 to Cha	1/26/12	1/22/10	Cha
39	U.S. Pat. App. Pub. 20100332593 to Barash, et al	12/3/10	6/29/10	Barash
40	Static Detection of Malicious Code in Executable Programs (J. Bergeron, et al)	1/2009	NA	Bergeron
41	U.S. 8,079,080 to Borders	12/13/11	10/21/05	Borders
42	U.S. Pat. App. Pub. 2010/0263048 to Chang, et al	10/14/10	4/14/09	Chang
43	Unattended Remote Attestation Delegation for Grid Computing (G. Cheng)	7/2009	NA	G. Cheng Paper
44	CN 101895577 to Hanwen, et al	11/24/10	7/6/10	Hanwen
45	WO 2007/104988 to Curnyn	9/20/07	3/14/07	Curyn
46	U.S. 9,100,425 to Friedrichs, et al	8/4/15	11/30/11	Friedrichs
47	U.S. 8,387,138 to Howcroft, et al	2/26/13	3/21/06	Howcraft
48	U.S. 7,831,752 to Illowsky, et al	11/9/10	10/21/08	Illowsky
49	Remote Attestation to Dynamic System Properties: Towards Providing Complete System Integrity Evidence (C. Kil, et al)	9/29/2009	NA	Kil
50	KR 100368810 to Hun	2/9/02		Hun
51	U.S. Pat. App. Pub. 2007/0006304 to Kramer, et al	1/4/07	6/30/05	Kramer
52	EP 2,383,653 to Kwon	11/2/11	12/21/09	Kwon
53	U.S. Pat. App. Pub 2011/0247074	10/6/11	3/30/10	Manring
54	Ensuring Content Integrity for Untrusted Peer-to-Peer Content	4/2007	NA	Michalakis

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
	Distribution Networks (N. Michalakis, et al)			
55	U.S. 7,614,084 to Monastyrsky, et al	11/3/09	10/2/07	Monastyrsky
56	Scalable Web Content Attestation (T. Moyer)	2009	NA	Moyer
57	U.S. 7,739,663 to Newcomb	6/15/10	5/16/06	Newcomb
58	U.S. 7,464,270 to Patankar, et al	12/9/08	7/23/05	Patankar
59	U.S. 7,302,677 to Reissman, et al	11/27/07	5/8/03	Reissman
60	Trusted Isolation Environment: An Attestation Architecture with Usage Control Model (A. Ruan, et al)	2009	NA	Ruan
61	Application-oriented trust in distributed computing (R. Scandariato, et al)	3/2008	NA	Scandariato
62	U.S. 6,643,690 to Duursma, et al	11/4/03	2/18/99	Duursma
63	U.S. 6,823,520 to Susser, et al	11/23/04	11/22/99	Susser
64	WO 2006/024216 to Luo	9/3/06	7/29/05	Luo
65	Secure Virtual Application Distribution (W. Zakaria, et al)	2009	NA	Zakaria
66	OfficeScan 10.5 Administrator's Guide (Trend Micro Inc.)	August 2010	NA	OfficeScan Guide[4]
67	Integrity Management Infrastructure for Trusted Computing (Munetoh *et al.*)	2008	N/A	Munetoh
68	Tisa: Toward Trustworthy Services in a Service-Oriented Architecture (Rajan *et al.*)	December 2008	N/A	Rajan

[4] This reference relates to the OfficeScan system and is evidence of that prior art system's functionality, as well as its availability and that it was on sale and/or offered for sale. Moreover, this reference qualifies independently as prior art under at least 35 U.S.C. §§ 102(a), (b), and (g).

4. ’517 Patent

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
1	U.S. Pat. App. Pub. 2011/0145926 to Dalcher *et al.*	6/16/11	12/15/09	Dalcher
2	U.S. Pat. App. Pub. 2012/0260344 to Maor *et al.*	10/11/12	12/14/10	Maor
3	Swimmer, “Dynamic Detection and Classification of Computer Viruses Using General Behavior Patterns”, Virus Bulletin Conference, September 1995	Sept. 20-22, 1995	N/A	VIDES[5]
4	ThunderBYTE AV User Manual	1995	N/A	ThunderBYTE AV 95[6]
5	ThunderBYTE AV User Manual	1996	N/A	ThunderBYTE AV 96[7]
6	ThunderBYTE AV User Manual	1997	N/A	ThunderBYTE AV 97[8]
7	U.S. Pat. App. Pub. 2013/0254880 to Alperovitch *et al.*	9/26/13	3/21/12	Alperovitch
8	US 9,117,075 to Yeh	8/25/15	11/22/10	Yeh
9	US 9,117,078 to Chien *et al.*	8/25/15	9/17/08	Chien
10	U.S. Pat. App. Pub. 2007/0143851 to Nicodemus *et al.*	6/21/07	12/21/05	Nicodemus

[5] This reference relates to the VIDES system and is evidence of that prior art system’s functionality, as well as its availability and that it was in public use, on sale, and/or offered for sale. Moreover, this reference qualifies independently as prior art under at least 35 U.S.C. §§ 102(a), (b), and (g).

[6] This reference relates to the ThunderBYTE system and is evidence of that prior art system’s functionality, as well as its availability and that it was on sale and/or offered for sale. Moreover, this reference qualifies independently as prior art under at least 35 U.S.C. §§ 102(a), (b), and (g).

[7] This reference relates to the ThunderBYTE system and is evidence of that prior art system’s functionality, as well as its availability and that it was on sale and/or offered for sale. Moreover, this reference qualifies independently as prior art under at least 35 U.S.C. §§ 102(a), (b), and (g).

[8] This reference relates to the ThunderBYTE system and is evidence of that prior art system’s functionality, as well as its availability and that it was on sale and/or offered for sale. Moreover, this reference qualifies independently as prior art under at least 35 U.S.C. §§ 102(a), (b), and (g).

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
11	US 8,401,982 to Satish	3/19/13	1/14/10	Satish
12	US 8,713,684 to Bettini	8/29/13	1/11/13	Bettini
13	CN 1417690 to Jun, et al	5/14/03	12/3/02	Jun
14	CN 101562609 to Ling, et al	10/21/09	5/27/09	Ling
15	CN 101572691 to Meijuan, et al	11/4/09	4/30/08	Meijuan
16	CN 101924762 to Hongyi, et al	12/22/10	8/18/10	Hongyi
17	U.S. 8,156,558 to Goldfeder, et al	4/10/12	5/17/03	Goldfeder
18	Multi-layer Monitoring for Cloud Computing (Gonzalez, et al)	2011	NA	Gonzalez
19	KR 100432421 to Kim, et al		12/21/01	Kim
20	U.S. 8,825,578 to Lay, et al	9/2/14	10/11/10	Lay
21	Dynamic Security Monitoring for Visualized Environments in Cloud Computing (A. Mana, et al)	2011	NA	Mana
22	Host-Based Intrusion Detection for Advanced Mobile Devices (M. Miettinen, et al)	2006	NA	Miettinen
23	U.S. 9,501,644 to Niemelä, et al	11/22/16	3/15/10	Niemelä
24	Malware Detection using Windows Api Sequence and Machine Learning (C. Ravi, et al)	4/2012	NA	Ravi
25	Google Android: A State-of-the-Art Review of Security Mechanisms (A. Shabtai, et al)	2009	NA	Shabtai
26	U.S. 4,224,664 to Trinchieri	9/23/80	5/7/76	Trinchieri
27	U.S. Pat. Pub. App 2012/0191660 to Hoog	7/26/12	1/26/12	Hoog
27	U.S. 9,047,441 to Xie, et al	6/2/15	5/24/11	Xie
29	WO 2008/060010 to Yoo, et al	5/22/08	12/8/06	Yoo

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
30	OfficeScan 10.5 Administrator's Guide (Trend Micro Inc.)	August 2010	NA	OfficeScan Guide[9]
31	US Pat. App. Pub. 2011/0023115 to Wright	January 27, 2011	March 31, 2010	Wright
32	US 8,806,647 to Daswani *et al.*	August 12, 2014	April 27, 2021	Daswani

5. '038 Patent & '918 Patent

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
1	US 8,065,712 to Cheng *et al.*	11/22/11	5/25/05	Cheng
2	U.S. Pat. App. Pub. 2006/0070129 to Sobel *et al.*	3/30/06	11/9/05	Sobel I
3	U.S. Pat. App. Pub. 2002/0199116 to Hoene *et al.*	12/26/02	6/25/01	Hoene
4	US 8,407,345 to Lim	3/26/13	10/30/07	Lim
5	U.S. Pat. App. Pub. 2007/0124803 to Taraz	5/31/07	11/29/05	Taraz
6	US 6,892,241 to Kouznetsov *et al.*	5/10/05	9/28/01	Kouznetsov
7	US 8,635,661 to Shahbazi	1/21/14	12/22/04	Shahbazi
8	U.S. Pat. App. Pub. 2004/0167984 to Herrmann	8/26/04	3/17/04	Herrmann I
9	US 6,873,988 to Herrmann *et al.*	3/29/05	7/9/02	Herrmann II
10	US 8,381,297 to Touboul	2/19/13	3/15/06	Touboul
11	US 9,100,431 to Oliphant *et al.*	8/4/15	9/28/14	Oliphant I
12	US 9,224,168 to Thirumalai *et al.*	12/29/15	12/11/12	Thirumalai

[9] This reference relates to the OfficeScan system and is evidence of that prior art system's functionality, as well as its availability and that it was on sale and/or offered for sale. Moreover, this reference qualifies independently as prior art under at least 35 U.S.C. §§ 102(a), (b), and (g).

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
13	U.S. Pat. App. Pub. 2003/0055994 to Herrmann	3/20/03	7/9/02	Herrmann III
14	U.S. Pat. App. Pub. 2004/0111643 to Farmer	6/10/04	12/2/03	Farmer
15	US 6,466,779 to Moles *et al.*	10/15/02	3/7/00	Moles
16	US 6,772,345 to Shetty	8/3/04	2/8/02	Shetty
17	US 6,904,458 to Bishop *et al.*	6/7/05	4/26/00	Bishop
18	US 7,353,533 to Wright *et al.*	4/1/08	4/11/03	Wright
19	US 7,992,199 to Winick *et al.*	8/2/11	12/31/03	Winick
20	US 8,544,078 to Lee	9/24/13	12/27/04	Lee
21	US 8,984,644 to Oliphant *et al.*	3/17/15	9/28/14	Oliphant III
22	U.S. Pat. App. Pub. 2006/0242686 to Toda *et al.*	10/26/06	2/20/04	Toda
23	U.S. Pat. App. Pub. 2007/0130457 to Kamat *et al.*	6/7/07	12/2/05	Kamat
24	USH1944 to Cheswick *et al.*	2/6/01	3/24/98	Cheswick
25	US 7,178,166 to Taylor *et al.*	2/13/07	9/19/00	Taylor
26	US 10,339,162 to Baum *et al.*	7/2/19	1/31/16	Baum
27	U.S. Pat. App. Pub. 2006/0098692 to D'Angelo	5/11/06	12/13/05	D'Angelo
28	US 8,200,818 to Freund *et al.*	6/12/12	8/30/01	Freund
29	U.S. Pat. App. Pub. 2004/0107360 to Herrmann	6/3/04	3/13/03	Herrmann IV
30	US 7,249,187 to Sobel *et al.*	7/24/07	11/27/02	Sobel II
31	US 7,827,607 to Sobel *et al.*	11/2/10	11/9/05	Sobel III
32	US 7,694,343 to Sobel *et al.*	4/6/10	11/9/05	Sobel IV
33	US 7,805,752 to Newstadt *et al.*	9/28/10	11/9/05	Newstadt

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
34	US 8,001,610 to Chickering *et al.*	8/16/11	9/28/05	Chickering
35	US 5,987,611 to Freund	11/16/99	5/6/97	Freund II
36	US 7,546,629 to Albert *et al.*	6/9/09	3/6/02	Albert
37	US 8,200,773 to Bluestone *et al.*	6/12/12	9/28/01	Bluestone
38	US 7,627,893 to Corley *et al.*	12/1/09	12/20/05	Corley
39	U.S. Pat. App. Pub. 2006/0203815 to Couillard	9/14/06	3/10/05	Couillard
40	US 8,938,799 to Kuo	1/20/15	6/28/04	Kuo
41	U.S. 6,098,173 to Elgressy, et al	8/1/00	11/3/98	Elgressy
42	U.S. Pat. App. Pub. 2005/0254652 to Engler, et al	11/17/05	7/14/03	Engler
43	U.S. 8,255,973 to Hopen, et al	8/28/12	12/10/04	Hopen
44	JP 2002-247033	8/30/02		JP 2002-247033
45	JP 2002-366525	12/20/02		JP 2002-366525
46	JP 2004-362075	12/24/04		JP 2004-362075
47	U.S. Pat. App. Pub 2006/008850 to Mayfield, et al	4/20/06	2/14/05	Mayfield
48	U.S. Pat. App. Pub. 2007/0123214 to Mock	5/31/07	11/25/05	Mock
49	Active Networks: Applications, Security, Safety, and Architectures (K. Psounis)	1999	NA	Psounis
50	U.S. Pat. App. Pub. 2006/0174115 to Rao et al	8/3/06	10/21/05	Rao
51	U.S. 7,669,237 to Shahbazi	2/23/10	8/27/03	Shahbazi
52	U.S. Pat. App. Pub. 2007.0055752 to Wiegand, et al	3/8/07	9/8/05	Wiegand

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
53	WO 2005/086681 to Miltonberger, et al	9/22/05	3/4/05	Miltonberger
54	WISE – Building Simple Intelligence into Web Services (S. Yuan and K. Lin)	2003	NA	Yuan
55	CyberGatekeeper monitors policies	2003	NA	Behr[10]
56	Compliance Monitor, InfoExpress CyberGatekeeper LAN is thorough, if tedious to configure	2004	NA	Demaria[11]
57	CyberGatekeeper LAN Overview	2004	NA	InfoExpress Web[12]
58	https://web.archive.org/web/20041214033336/http://www.infoexpress.com/security_products/lan_security_benefits.php	2004	NA	InfoExpress lan security benefits[13]
59	https://web.archive.org/web/20041214032534/http://www.infoexpress.com/security_products/lan_security_how_works.php	2004	NA	InfoExpress lan Security[14]
60	https://web.archive.org/web/20041214034330/http://www.infoexpress.com/security_products/lan_security_faq.php	2004	NA	InfoExpress lan Security FAQ[15]

[10] This reference relates to the CyberGatekeeper system and is evidence of that prior art system's functionality, as well as its availability and that it was on sale. Moreover, this reference qualifies independently as prior art under at least 35 U.S.C. §§ 102(a), (b), and (g).
[11] This reference relates to the CyberGatekeeper system and is evidence of that prior art system's functionality, as well as its availability and that it was on sale. Moreover, this reference qualifies independently as prior art under at least 35 U.S.C. §§ 102(a), (b), and (g).
[12] This reference relates to the CyberGatekeeper system and is evidence of that prior art system's functionality, as well as its availability and that it was on sale. Moreover, this reference qualifies independently as prior art under at least 35 U.S.C. §§ 102(a), (b), and (g).
[13] This reference relates to the CyberGatekeeper system and is evidence of that prior art system's functionality, as well as its availability and that it was on sale. Moreover, this reference qualifies independently as prior art under at least 35 U.S.C. §§ 102(a), (b), and (g).
[14] This reference relates to the CyberGatekeeper system and is evidence of that prior art system's functionality, as well as its availability and that it was on sale. Moreover, this reference qualifies independently as prior art under at least 35 U.S.C. §§ 102(a), (b), and (g).
[15] This reference relates to the CyberGatekeeper system and is evidence of that prior art system's

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
61	https://web.archive.org/web/20041208153410/http://www.infoexpress.com/security_products/pe_overview.php	2004	NA	InfoExpress Security Products[16]
62	https://web.archive.org/web/20050218091824/http://infoexpress.com/security_products/product_documentation.php	2005	NA	InfoExpress Product Documentation[17]
63	Enforcing Network Integrity Using CyberGatekeeper from InfoExpress	2004	NA	Enforcing Network Integrity[18]
64	The Elemental Compliance System	2005	NA	Elemental Web[19]
65	The iPass System	2004	NA	iPass
66	https://web.archive.org/web/20040815195440/http://www.ipass.com:80/company/company_story_policyorch.html	2004	NA	iPass Company Story[20]
67	https://web.archive.org/web/20040816014856/http://www.ipass.com/company/company_story_policyorch2.html	2004	NA	iPass Company Story II[21]

functionality, as well as its availability and that it was on sale. Moreover, this reference qualifies independently as prior art under at least 35 U.S.C. §§ 102(a), (b), and (g).

[16] This reference relates to the CyberGatekeeper system and is evidence of that prior art system's functionality, as well as its availability and that it was on sale. Moreover, this reference qualifies independently as prior art under at least 35 U.S.C. §§ 102(a), (b), and (g).

[17] This reference relates to the CyberGatekeeper system and is evidence of that prior art system's functionality, as well as its availability and that it was on sale. Moreover, this reference qualifies independently as prior art under at least 35 U.S.C. §§ 102(a), (b), and (g).

[18] This reference relates to the CyberGatekeeper system and is evidence of that prior art system's functionality, as well as its availability and that it was on sale. Moreover, this reference qualifies independently as prior art under at least 35 U.S.C. §§ 102(a), (b), and (g).

[19] This reference relates to the Elemental Security system and is evidence of that prior art system's functionality, as well as its availability and that it was on sale. Moreover, this reference qualifies independently as prior art under at least 35 U.S.C. §§ 102(a), (b), and (g).

[20] This reference relates to the iPass system and is evidence of that prior art system's functionality, as well as its availability and that it was on sale. Moreover, this reference qualifies independently as prior art under at least 35 U.S.C. §§ 102(a), (b), and (g).

[21] This reference relates to the iPass system and is evidence of that prior art system's functionality, as well as its availability and that it was on sale. Moreover, this reference qualifies independently as prior art under at least 35 U.S.C. §§ 102(a), (b), and (g).

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
68	https://web.archive.org/web/20040812184749/http://www.ipass.com/platform/platform_policyorch.html	2004	NA	iPass Platform[22]
69	https://web.archive.org/web/20050404010936/http://www.ipass.com/pressroom/pressroom_releases.html?rid=132	2004	NA	IPass Policy Orchestration [23]
70	IPass Endpoint Policy Management	2005	NA	IPass Endpoint Policy Management [24]
71	IPass Enpoint Technical Brief	2005	NA	IPass Endpoint Technical Brief[25]

6. '948 Patent

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
1	U.S. Pat. App. Pub. 2011/0126111 to Gill *et al.*	5/26/11	11/23/10	Gill
2	US 7,904,278 to Wilson *et al.*	3/8/11	5/2/07	Wilson
3	US 7,467,067 to Marvasti	12/16/08	9/27/06	Marvasti
4	US 8,141,157 to Farley *et al.*	3/20/12	12/7/09	Farley

[22] This reference relates to the iPass system and is evidence of that prior art system's functionality, as well as its availability and that it was on sale. Moreover, this reference qualifies independently as prior art under at least 35 U.S.C. §§ 102(a), (b), and (g).
[23] This reference relates to the iPass system and is evidence of that prior art system's functionality, as well as its availability and that it was on sale. Moreover, this reference qualifies independently as prior art under at least 35 U.S.C. §§ 102(a), (b), and (g).
[24] This reference relates to the iPass system and is evidence of that prior art system's functionality, as well as its availability and that it was on sale. Moreover, this reference qualifies independently as prior art under at least 35 U.S.C. §§ 102(a), (b), and (g).
[25] is reference relates to the iPass system and is evidence of that prior art system's functionality, as well as its availability and that it was on sale. Moreover, this reference qualifies independently as prior art under at least 35 U.S.C. §§ 102(a), (b), and (g).

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
5	US 10,692,032 to Datta Ray (US 61547610 Provisional)	6/23/20	1/4/19	Datta Ray I
6	US 9,241,008 to Powell *et al.*	1/19/16	12/2/13	Powell
7	US 9,286,469 to Kraemer *et al.*	3/15/16	8/4/06	Kraemer
8	US 9,298,917 to Uner *et al.*	3/29/16	1/13/12	Uner
9	US 9,311,495 to Readshaw *et al.*	4/12/16	12/9/10	Readshaw
10	US 9,460,303 to MacLeod *et al.*	10/4/16	3/6/12	MacLeod
11	US 9,607,142 to Dharmarajan *et al.*	3/28/17	9/9/11	Dharmarajan
12	US 10,331,904 to Sher-Jan *et al.*	6/25/19	10/17/17	Sher-Jan
13	U.S. Pat. App. Pub. 2006/0020866 to Lo *et al.*	1/26/06	6/15/05	Lo
14	US Pat. App. Pub. 2009/0328222 to Helman *et al.*	12/31/09	6/25/08	Helman
15	US Pat. App. Pub. 2011/0145711 to Njemanze *et al.*	6/16/11	12/2/2	Njemanze
16	US Pat. App. Pub. 2013/0097660 to Das *et al.*	4/18/13	10/17/11	Das
17	US Pat. App. Pub. 2009/0100518 to Overcash	4/16/09	9/21/07	Overcash II
18	US 10,198,581 to Sreenivas *et al.*	2/5/19	3/7/12	Sreenivas
19	US 8,474,004 to Leone	6/25/13	7/31/06	Leone
20	US 10,476,878 to Faitelson	1/3/19	9/7/18	Faitelson
21	US 2009/0320021 to Pan	12/24/09	6/19/08	Pan
22	Williams (US20110231936A1)	9/22/11	8/27/10	Williams
23	Bose *et al.*, "Behavioral Detection of Malware on Mobile Handsets", MobiSys '08: Proceedings of the 6th International Conference on Mobile Systems, Applications, and Services, pages 225–238.	June 2008	N/A	Bose

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
24	U.S. 8,949,797 to Aaraj, et al	2/3/15	4/16/10	Aaraj
25	Self-Healing and Secure Adaptive Messaging Middleware for Business-Critcal Systems (H. Abie, et al)	2010	NA	Abie
26	Runtime Monitoring of Metric First-order Temporal Properties (D. Basin, et al)	2008	NA	Basin
27	U.S. 8,555,385 to Bhatkar, et al	10/8/13	3/14/11	Bhatkar
28	U.S. 10,805,331 to Boyer, et al	10/13/20	9/22/11	Boyer
29	U.S. 7,739,737 to Christodorescu, et al	6/15/10	7/29/03	Christodorescu
30	U.S. 8,453,234 to Dawson, et al	5/28/13	9/20/06	Dawson
31	U.S. 8,955,122 to Gu, et al	2/10/15	4/4/08	Gu
32	U.S. 8,397,301 to Hering, et al	3/12/13	11/18/09	Hering
33	Topological Vulneribility Analyis: A Powerful New Approach For Network Attack Prevention, Detection and Response (S. Jajodia and S. Noel)	11/2008		Jajodia
34	U.S. Pat. App. Pub. 2013/0055401 to Kim, et al	2/28/12	12/22/11	Kim
35	KR 2005-0051223 to Seo, et al	6/1/05		Seo
36	U.S. 8,789,192 to LaBumbard	7/22/14	4/9/12	LaBumbard
37	WO 2006/002071 to Lo, et al	1/5/06	6/15/05	Lo
38	U.S. 8,499,353 to Lockhart, et al	7/30/13	2/15/08	Lockhart
39	WO 2012/027588 to Mahaffey, et al	3/1/12	8/25/11	Mahaffey
40	U.S. 8,056,130 to Njemanze, et al	11/8/11	4/4/08	Njemanze
41	U.S. 2011/0231361 to Patchava, et al	9/22/11	12/31/10	Patchava
42	U.S. 7,415,130 to Rundle, et al	8/19/08	10/30/03	Rundle

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
43	Design and Implementation of a TCG-based Integrity Measurement Architecture (R. Sailer, et al)	9/2004	NA	Sailer
44	U.S. 8,533,836 to Saunders, et al	9/10/13	1/13/12	Saunders
45	U.S. Pat App. Pub. 2011/0098973 to Siedman	4/28/11	10/23/09	Seidman
46	A Trust Evaluation Fremework in Distributed Networks: Vulnerability Analysis and Defense Against Attacks (Y. Sun, et al)	4/2006	NA	Sun
47	OfficeScan 10.5 Administrator's Guide (Trend Micro Inc.)	August 2010	NA	OfficeScan Guide[26]
48	US 7,451,488 to Cooper *et al.*	November 11, 2008	April 29, 2004	Cooper
49	Network Intrusion Detection (Mukherjee *et al.*)	May/June 1994	N/A	Mukherjee

7. '616 Patent

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
1	US 8,434,147 to Jaeger *et al.*	4/30/13	11/7/05	Jaeger
2	US 9,298,922 to Berger *et al.*	3/29/16	7/10/08	Berger
3	US 7,904,278 to Wilson *et al.*	3/8/11	5/2/07	Wilson
4	US 8,429,412 to Bleckmann *et al.*	4/23/13	12/8/10	Bleckmann
5	US 7,930,733 to Iftode *et al.*	4/19/11	4/10/06	Iftode
6	US 7,490,352 to Kramer *et al.*	2/10/09	4/7/05	Kramer

[26] This reference relates to the OfficeScan system and is evidence of that prior art system's functionality, as well as its availability and that it was on sale and/or offered for sale. Moreover, this reference qualifies independently as prior art under at least 35 U.S.C. §§ 102(a), (b), and (g).

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
7	U.S. Pat. App. Pub. 2005/0005169 to Kelekar	1/6/05	4/9/04	Kelekar
8	U.S. Pat. App. Pub. 2005/0005171 to Oliphant	1/6/05	7/1/04	Oliphant II
9	US 7,451,488 to Cooper *et al.*	11/11/08	4/29/04	Cooper
10	US 9,628,501 to Datta Ray	4/18/17	9/4/14	Datta Ray II
11	US 10,452,851 to Nunez Di Croce	10/22/19	2/4/16	Nunez Di Croce
12	U.S. Pat. App. Pub. 2006/0191007 to Thielamay	8/24/06	2/24/05	Thielamay
13	US 7,315,801 to Dowd *et al.*	1/1/08	1/14/00	Dowd
14	US 7,849,497 to Hurst *et al.*	12/7/10	12/14/06	Hurst
15	US 8,601,587 to Powell *et al.*	12/3/13	9/3/10	Powell II
16	US 8,850,565 to Patrick *et al.*	9/30/14	1/10/05	Patrick
17	US 9,098,335 to Muthiah *et al.*	8/4/15	12/22/10	Muthiah I
18	US 9,111,088 to Ghai *et al.*	8/18/15	8/14/07	Ghai
19	US 10,795,855 to DiFalco	10/6/20	7/2/18	DiFalco I
20	US 10,846,136 to Muthiah *et al.*	11/24/20	7/30/15	Muthiah II
21	U.S. Pat. App. Pub. 2006/0282886 to Gaug	12/14/06	6/9/06	Gaug
22	U.S. Pat. App. Pub. 2007/0192867 to Miliefsky	8/16/07	1/23/06	Miliefsky
23	U.S. Pat. App. Pub. 2009/0235359 to Abdulhayoglu *et al.*	9/17/09	8/8/08	Abdulhayoglu
24	Nessus 4.2, vulnerability scanner circa August 2010: https://web.archive.org/web/20100825	2010	N/A	Nessus 4.2[27]

[27] This reference relates to the Nessus 4.2 system and is evidence of that prior art system's functionality, as well as its availability and that it was on sale and/or offered for sale. Moreover, this reference qualifies independently as prior art under at least 35 U.S.C. §§ 102(a), (b), and (g).

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
	005603/http://www.nessus.org/documentation/index.php?doc=faq; https://en.wikipedia.org/wiki/Nessus_(software)"			
25	U.S. Pat. App. Pub. 2012/0110174 to Wootton *et al.*	5/3/12	12/22/11	Wootton
26	U.S. Pat. 7,934,253 to Overcash *et al.*	4/26/11	7/20/06	Overcash
27	U.S. Pat. App. Pub. 2013/0097660 to Das *et al.*	4/18/13	10/17/11	Das
28	U.S. Pat. App. Pub. 2009/0100518 to Overcash	4/16/09	9/21/07	Overcash II
29	US 10,198,581 to Sreenivas *et al.*	2/5/19	3/7/12	Sreenivas
30	US 8,474,004 to Leone	6/25/13	7/31/06	Leone
31	Brandwine (US 8,745,734)	6/3/14	12/29/10	Brandwine
32	Croce (US 10,452,851)	6/2/16	2/4/16	Croce
33	Mutnuru (US 8,825,859)	6/23/11	12/22/10	Mutnuru
34	Schrecker (US 8,392,998)	3/5/13	11/30/09	Schrecker
35	Singla (US 9,438,616)	3/13/14	11/6/13	Singla
36	DiFalco (US 10,264,022)	8/11/16	2/5/16	DiFalco II
37	U.S. 8,949,797 to Aaraj, et al	2/3/15	4/16/10	Aaraj
38	Self-Healing and Secure Adaptive Messaging Middleware for Business-Critcal Systems (H. Abie, et al)	2010	NA	Abie
39	Runtime Monitoring of Metric First-order Temporal Properties (D. Basin, et al)	2008	NA	Basin
40	U.S. 8,555,385 to Bhatkar, et al	10/8/13	3/14/11	Bhatkar
41	U.S. 10,805,331 to Boyer, et al	10/13/20	9/22/11	Boyer
42	U.S. 7,739,737 to Christodorescu, et al	6/15/10	7/29/03	Christodorescu

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
43	U.S. 8,453,234 to Dawson, et al	5/28/13	9/20/06	Dawson
44	U.S. 8,955,122 to Gu, et al	2/10/15	4/4/08	Gu
45	U.S. 8,397,301 to Hering, et al	3/12/13	11/18/09	Hering
46	Topological Vulneribility Analyis: A Powerful New Approach For Network Attack Prevention, Detection and Response (S. Jajodia and S. Noel)	11/2008		Jajodia
47	U.S. Pat. App. Pub. 2013/0055401 to Kim, et al	2/28/12	12/22/11	Kim
48	KR 2005-0051223 to Seo, et al	6/1/05		Seo
49	U.S. 8,789,192 to LaBumbard	7/22/14	4/9/12	LaBumbard
50	WO 2006/002071 to Lo, et al	1/5/06	6/15/05	Lo
51	U.S. 8,499,353 to Lockhart, et al	7/30/13	2/15/08	Lockhart
52	WO 2012/027588 to Mahaffey, et al	3/1/12	8/25/11	Mahaffey
53	U.S. 8,056,130 to Njemanze, et al	11/8/11	4/4/08	Njemanze
54	U.S. 2011/0231361 to Patchava, et al	9/22/11	12/31/10	Patchava
37	U.S. 7,415,130 to Rundle, et al	8/19/08	10/30/03	Rundle
38	Design and Implementation of a TCG-based Integrity Measurement Architecture (R. Sailer, et al)	9/2004	NA	Sailer
39	U.S. 8,533,836 to Saunders, et al	9/10/13	1/13/12	Saunders
40	U.S. Pat App. Pub. 2011/0098973 to Siedman	4/28/11	10/23/09	Seidman
41	A Trust Evaluation Fremework in Distributed Networks: Vulnerability Analysis and Defense Against Attacks (Y. Sun, et al)	4/2006	NA	Sun

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
42	OfficeScan 10.5 Administrator's Guide (Trend Micro Inc.)	August 2010	NA	OfficeScan Guide[28]
43	Network Intrusion Detection (Mukherjee *et al.*)	May/June 1994	N/A	Mukherjee

8. '356 Patent

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
1	US 7,707,305 to Afek, et al	4/27/2010	8/14/2001	Afek
2	WO 03/050644 to Afek, et al	6/19/2003	12/10/2002	Afek II
3	US 7,069,436 to Akachi	6/17/2006	11/1/2000	Akachi
4	US 7,448,084 to Apap, et al	11/4/2008	1/27/2003	Apap
5	US 8,161,145 to Carpenter, et al	4/17/2012	2/27/2003	Carpenter
6	US Pat. App. Pub 2004/0042470 to Cooper	3/4/2004	6/2/2003	Cooper
7	US 7,290,283 to Copeland, III	10/30/2007	1/31/2002	Copeland
8	US Pat. App. Pub. 2003/0188189 to Desai, et al	10/2/2003	3/27/2002	Desai
9	US 6,662,230 to Eichstaedt, et al	12/9/2003	10/20/2009	Eichstaedt
10	US Pat. App. Pub 2003/0236995 to Fretwell	12/25/2003	6/21/2002	Fretwell
11	US 8,640,234 to Gassen, et al	1/28/2014	5/4/2004	Gassen
12	US 5,948,104 to Gluck, et al	9/7/1999	5/23/2997	Gluck
13	US 6,564,261 to Gudjonsson, et al	5/13/2003	5/9/2000	Gudjonsson
14	US 7,234,168 to Gupta, et al	6/19/2007	6/13/2002	Gupta

[28] This reference relates to the OfficeScan system and is evidence of that prior art system's functionality, as well as its availability and that it was on sale and/or offered for sale. Moreover, this reference qualifies independently as prior art under at least 35 U.S.C. §§ 102(a), (b), and (g).

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
15	US 6,988,208 to Hrabik, et al	1/17/2006	7/16/2002	Hrabik
16	US 7,644,151 to Jerrim, et al	1/5/2010	3/25/2002	Jerrim
17	US Pat. App. Pub. 2002/035683 to Kaashoek, et al	3/21/2002	8/16/2001	Kaashoek
18	US 7,574,740 to Kennis	8/11/2009	4/28/2000	Kennis
19	US Pat. App. Pub 2002/0166063 to Lachman, et al	11/7/2002	2/28/2002	Lachman
20	KR2003-0035142 to Lee, et al	5/9/2003	10/30/2001	Lee
21	US 5,530,703 to Liu, et al	6/25/1996	9/23/1994	Liu
22	*Network Traffic Anomaly Detection Based on Packet Bytes (*Mahoney, M. V.)	9/6/2002		Mahoney
23	US Pat. App. Pub. 2005/0018618A1 to Haulem, et al	1/27/2005	7/22/2004	Maulem
24	US 6,289,462 to McNabb, et al	9/11/2001	9/28/1999	McNabb
25	US 6,910,077 to Najork	6/21/2005	1/4/2002	Najork
26	*Representing TCP/IP Connectivity For Topological Analysis of Network Security (*Ritchey, R. *et al.*)	Dec-02		Ritchey
27	WO 2002/098100 to Roux	12/5/2002	5/31/2001	Roux
28	*Detection of Promiscuous Nodes Using ARP Packets, Version 1.0 (*Sanai, D.)	8/1/2001		Sanai
29	US 7,159,237 to Schneier, et al	1/2/2007	1/19/2001	Schneier
30	US 8,205,259 to Stute	6/19/2012	3/28/2003	Stute
31	US 7,409,707 to Swander, et al	8/5/2008	6/6/2003	Swander
32	US 7,509,673 to Swander, et al	3/24/2009	6/6/2003	Swander II
33	US 7,594,009 to Truilzi, et al	9/22/2009	9/13/2001	Triulzi
34	US 7,028,009 to Wang, et al	4/11/2006	6/3/2002	Wang

No.	Patent No. or Title (Primary Inventor/Author)	Date of Issue / Publication	Filing Date	Herein Referred As
35	US 8,065,725 to Zheng, et al	11/22/2011	5/30/2003	Zheng
36	US Pat. App. Pub. 2004/196843 to Zinin	10/7/2004	2/19/2004	Zinin

B. Prior Art Apparatuses and Systems

Defendant contends that the following systems and products are prior art under at least one or more of 35 U.S.C. § 102(a), (b), (e), (f), or (g) (pre-AIA) or § 102(a) (post-AIA), as applicable, and/or § 103.[29] Defendant's reference to any particular component, device, machine, or other product in these Invalidity Contentions should be interpreted as a reference to the product itself and any corresponding patents, publications, or product literature cited in Exhibits A-P that relate to the cited component, device, machine, or other product. Insofar as Defendant is relying on a system or product as prior art, rather than a publication, Defendant may use additional evidence to show the structure, function, operation and prior art status of that system or product insofar as that evidence is obtained during discovery and in a manner that is consistent with the invalidity theory disclosed in these Invalidity Contentions. At least some of the products were publicly disclosed, used, sold, or offered for sale in the United States before, and potentially more than a year before, the earliest priority date of the Asserted Claims of the Patents-in-Suit. Moreover, many of the inventive research, design, and development activities concerning these products and technologies occurred in the United States before the earliest priority date of the Asserted Claims of the Patents-in-Suit.

Patent	Prior Art Apparatus / System
'137 Patent	• ChakraVyuha was made, used, sold, and/or offered for sale in the United States by IBM at least as early as 1997.

[29] A reference to a version of a commercial product or system is also a reference to any earlier or later version of the commercial product or system that qualifies as prior art to an Asserted Claim.

Patent	Prior Art Apparatus / System
’441 Patent	• Virus Total was made, used, sold, and/or offered for sale in the United States by Hispasec Sistemas to customers at least as early as June 2004. • OfficeScan was made, used, sold, and/or offered for sale in the United States by Trend Micro, Inc. to customers at least as early as August 2010, including version 10.5 that was made, used, sold, and/or offered for sale as early as August 2010.
’517 Patent	• Virus Intrusion Detection Expert System (VIDES) was made, used, sold, and/or offered for sale in the United States at least as early as 1995. • ThunderBYTE Anti-Virus was made, used, sold, and/or offered for sale in the United States by ESSaSS B.V. to customers at least as early as 1996. • Norman Scanner Engine • Norman Virus Control • OfficeScan was made, used, sold, and/or offered for sale in the United States by Trend Micro, Inc. to customers at least as early as August 2010, including version 10.5 that was made, used, sold, and/or offered for sale as early as August 2010.
’038 & ’918 Patents	• Checkpoint R60 and/or Check Point’s NGX system, including its Management Server (SmartCenter), Virtual Private Network gateways, corresponding SecuRemote/SecureClient remote access clients (SRSC or “RAVS”) using the Secure Configuration Verification (SCV) feature as well as Integrity Client (also known as Integrity Agent and Integrity Flex) and Integrity Server (also known as Integrity Advanced Server or IAS) (together, the Integrity Advanced Server and Client system, or “IASC”), and associated publications (collectively, “Check Point R60”) • CyberGatekeeper was made, used, sold, and offered for sale in the United States by InfoExpress to customers at least as early as December 5, 2001 and continuously thereafter (*e.g.*, https://www.helpnetsecurity.com/2002/04/08/infoexpress-launches-cybergatekeeper-a-remote-system-policy-enforcer-for-corporate-networks/). • Elemental Compliance System was made, used, sold, and offered for sale in the United States by Elemental, Inc. or Elemental Cyber Security, Inc. to customers at least as early as January 1, 2005 and thereafter. • iPass was made, used, sold, and offered for sale in the United State by iPass Inc. to customers at least as early as 2004 and thereafter.

Patent	Prior Art Apparatus / System
’948 Patent	• Tenable Nessus 4.2 vulnerability scanner • OfficeScan was made, used, sold, and/or offered for sale in the United States by Trend Micro, Inc. to customers at least as early as August 2010, including version 10.5 that was made, used, sold, and/or offered for sale as early as August 2010.
’616 Patent	• Tenable Nessus 4.2 vulnerability scanner • OfficeScan was made, used, sold, and/or offered for sale in the United States by Trend Micro, Inc. to customers at least as early as August 2010, including version 10.5 that was made, used, sold, and/or offered for sale as early as August 2010.

Each of these systems anticipates and/or renders obvious one or more of the Asserted Claims of the Patents-in-Suit as set forth in the associated chart, and/or forms part of the state of the art that was well understood to those of skill in the art as of the respective priority dates of the Asserted Patents. Defendant relies on the references cited above at least as evidence of the functionality of the associated system. Defendant also reserves the right to rely on each of the references cited above as itself anticipating or rendering obvious, alone or in combination with any other prior art in the associated charts, the Asserted Claims.

Defendant does not yet have complete information regarding the dates by which these systems were publicly disclosed, made, used, sold, or offered for sale, the circumstances under which the research, design, and development activities were conducted, and the identities of the particular individuals involved in such activities. Defendant anticipates that the actual dates, circumstances, and identities of individuals will be the subject of third-party discovery during this lawsuit. Defendant therefore reserves the right to amend and/or supplement their Invalidity Contentions and associated charts if additional information becomes available during the course of discovery. The date the prior art systems cited herein were “offered for sale or publicly used or known,” as well as “the date the offer or use of the item took place or the information became known” is at least as early as the date the related publications were published. The “identity of the

person or entity which made the use or which made and received the offer, or the person or entity which made the information known or to whom it was made known" includes at least those persons associated with the publication(s) or other documents associated with each prior art system, including those identified in the claim charts. Defendant also reserves the right to rely on the public use, offer for sale, or sale of the systems and products described in the publications identified herein or in Exhibits A-P after Defendant has had a fair opportunity to take discovery on these subjects.

C. Applicants Admitted Prior Art (AAPA)

In addition to the prior art identified above and charted in the accompanying claim charts, Defendant intends to rely on applicants admitted prior art (AAPA) in each Asserted Patent as part of the state of the art that was well known to those of skill in the art at the time of the respective priority dates of the Asserted Patents. In further detail below, the Asserted Patents admit various problems, as well as devices and methods for addressing those problems, were well known to those of skill in the art. Examples of these problems, devices, and methods disclosed in the Asserted Patents include the following:

1. The '796 Patent AAPA

- In the past, several techniques have been developed to solve the somewhat simpler problem of "named entity extraction." '796 Patent, Background.
- In this task, the goal is to identify all occurrences of certain classes of words in a document. For example, all the person-names, city-names, dates, and times might be identified. *Id*.
- One way of identifying such entities is to train a statistical classification system to tag each word in the document as either a "person-name," "city-name," "date," "time," or "other" word. Examples of this sort of approach are disclosed in, e.g., Ratnaprkhi, "A Maximum Entropy Part of Speech Tagger," Conference on Empirical Methods in Natural

Language Processing, University of Pennsylvania, 1996; and Brill, "Transformation-Based Error-Driven Learning and Natural Language Processing: A Case Study in Part of Speech Tagging," Computational Linguistics, December 1995, the disclosures of which are incorporated by reference herein. *Id*.

- While the problem of extracting key pieces of information can also be viewed as a tagging problem, in which each word is tagged as either "key information" or "irrelevant,…" *Id*.
- Regular expressions, as have been discussed in Aho *et al.*, "Compilers: Principles, Techniques, and Tools," Addison-Wesley, 1985, the disclosure of which is incorporated by reference herein, comprise symbolic strings, typically word sequences, that can be constructed by the basic operations of "and" and "or" acting on an atomic alphabet of symbols. 3:2-8.
- A wide variety of extensions to regular expressions exist, for example, restrictions on where in a document a pattern is found, or restrictions on the length of repeated sequences in a regular expression. In accordance with an illustrative embodiment of the invention, the term "regular expression" is taken to mean any form of pattern that is matchable in the flex, lex, or perl programming languages, although the invention is not limited thereto. The programming languages flex and lex are described in J. R. Levine *et al.*, "Lex & Yacc," 1992, the disclosure of which is incorporated by reference herein. The programming language perl is described in L. Wall *et al.*, "Programming Perl," 1996, the disclosure of which is incorporated by reference herein. Similarly, in accordance with such an illustrative embodiment, the term "matching" refers to the processes by which these programs match their input. 3:17-34.
- Referring now to FIG. 2, a block diagram illustrates a **generalized hardware architecture of a data processing system** suitable for implementing an information

extraction methodology according to the present invention. As shown, the system 30 includes a data capture device 32, a data signal processor 34, a data output device 36, and memory 38. 11:43-48.

- The data capture device 32 generally obtains the data stream that is to be processed by the system 30. The device's specific structure and operations depend on the application in which the system is employed. For example, in a voice mail application, the data capture device may be a speech processing device capable of taking an audio stream recorded in accordance with a voice mail system and converting or transcribing the audio stream to a discrete data or symbol sequence. The device may, for example, **utilize conventional speech recognition techniques or some other known transcription techniques**. In a text based application, the device may be a digital scanner capable of converting a text document into a discrete data or symbol sequence. The device may, for example, **utilize conventional optical character recognition techniques.** Of course, depending on the data type, the data capture device may take on alternate forms. 11:49-65.
- It is to be appreciated that the term “processor” as used herein is intended to include any processing device, such as, for example, one that includes a CPU (central processing unit) or other processing circuitry. For example, the processor may be a digital signal processor, **as is known in the art**. Also the term “processor” may refer to one or more individual processors. The term “memory” as used herein is intended to include any memory devices associated with a processor or CPU, such as, for example, RAM, ROM, a fixed memory device (e.g., hard drive), a removable memory device (e.g., diskette), flash memory, etc. It is to be understood that the previously-generated word classes and regular expressions used in the class-annotation and expression matching operations, described above in accordance with the invention, may be stored in memory 38 and accessed when required. 12:30-44.

- The data output device 36 generally presents the data extracted from the input data stream to a user. Again, the device's specific structure and operations depend on the application in which the system is employed. For example, in a voice mail application, the data capture device may be an audio output system capable of taking the extracted discrete data and converting the data to a form to be audibly output to the user. The device may, for example, **utilize conventional text-to-speech or other speech synthesis techniques**. Such techniques may also be used in a text based application. Alternatively, the device may be a display for visually presenting the extracted data to the user. Of course, depending on the data type, the data output device may take on alternate forms. 12:52-65.

2. The ’137 Patent AAPA

- With the growth of wide area networks (WANs), such as the Internet and the World Wide Web, people rely on computing networks to locate, transfer, and store an increasing amount of valuable information. This is also true of local area networks (LANs) used by companies, schools, organizations, and other enterprises. LANs generally are used by a bounded group of people in an organization to communicate and store electronic documents and information. LANs typically are coupled to or provide access to other local or wide area networks. Greater use and availability of computing networks produces a corresponding increase in the size and complexity of computing networks. ’137 Patent, Background
- One approach to securing larger and more complex computer networks is to use a greater number and variety of security assessment and intrusion detection devices. Security assessment devices can be used to evaluate elements in the network such as desktop computers, servers, and routers, and determine their respective vulnerability to attack from hackers. Intrusion detection devices, on the other hand, identify and prevent entry of foreign or malicious computer programs and can notify a network manager of the

presence or attempted entry of such a computer program. Security assessment and intrusion detection devices can also be used more frequently to monitor the activity or status of the elements in a computing network. *Id*.

- Threats to the security of a device or network can take a variety of forms including the introduction of harmful computer code, unauthorized attempts to gain access, and misuse by people with authority to use a device or network. The various types of harmful computer code that can threaten a computing device or distributed computing system can generally be categorized as either a virus or some form of "malware". Computer viruses harm computing devices and systems by entering and then propagating. In some respects, the propagating nature of computer viruses makes them easier to detect and there are many commercially available products that detect and exclude viruses from computing devices and systems. *Id*.
- In contrast, malware is a general description for other types of programs and computer code that are designed to harm a computing device or system in ways other than simply propagating as a virus does. Malware presents a more sophisticated challenge for network security and traditional anti-virus software is not designed to prevent malware from harming computing devices and networks. Malware can take a variety of forms including corrupted applications and applications that retrieve corrupted code that is modified to harm a computing device or network. *Id*.
- There are generally two different approaches to protecting against malware. The first approach involves virtual execution of a computer code to attempt to identify harmful code before it is allowed to actually execute. Virtual execution is limited in its ability to detect harmful code because it does not actually execute every process of the code. Instead, the virtual execution technique performs a quick and high-level "walk through" of the processes in the code to attempt to detect suspicious patterns in the code. By its

nature, the virtual execution technique is limited in its ability to detect suspicious activities embedded in a piece of code. As a result, when the virtual execution technique is implemented, it must be used conservatively which produces a high number of false positive alerts. In other words, because the virtual execution security technique is not as accurate as actually running the code, it is implemented to identify a broader scope of potentially suspicious code and produces a greater number of alerts to the user. A high percentage of false positive security alerts is undesirable because it translates into a greater number of security interruptions for the user. *Id*.

- The second approach involves controlling and monitoring a computing device in real time while it is actually running a program and attempting to anticipate any harmful activity the program may try to initiate. One example of a real-time solution is set forth in U.S. Pat. No. 5,987,611, which describes a client-based monitoring system for filtering network access in conjunction with a centralized enforcement supervisor. The supervisor maintains access rules for the client-based filtering and verifies the existence and proper operation of the client-based filter application. Access rules specify network access criteria for a client, such as (1) total time a user can be connected to the Internet (e.g., per day, week, month, or the like), (2) time a user can interactively use the Internet (e.g., per day, week, month, or the like), (3) a list of applications or application versions that a user can or cannot use in order to access the Internet, (4) a list of URLs (or WAN addresses) that a user application can (or cannot) access, (5) a list of protocols or protocol components that a user application can or cannot use, and (6) rules to determine what events should be logged (including how long are logs to be kept). *Id*.
- By intercepting process loading and unloading and keeping a list of currently-active processes, each client process can be checked for various characteristics, including checking executable names, version numbers, executable file checksums, version header

details, configuration settings, and the like. With this information, a determination can be made whether a particular process in question should have access to the Internet and what kind of access (i.e., protocols, Internet addresses, time limitations, and the like) is permissible for the given specific user. *Id*.

- The limitation with the solution presented in U.S. Pat. No. 5,987,611 and other similar real-time prior art solutions is that they are packet based. In other words, the security decisions are based on the data packets that are passing between the computing device that is being monitored and external networks or computing resources. When security decisions are based on the traffic of data packets, it is more likely the security systems will not detect harmful activities until after the harm has already begun. Accordingly, the second approach is not satisfactory because conventional real-time security monitoring solutions do not detect security problems early enough and allow time for a response before the malicious program does harm. *Id*.
- Examples of such distributed computing environments include local area networks of an office, enterprise-wide computer networks, and the global Internet. 5:8-11
- The detailed description that follows is represented largely in terms of processes and symbolic representations of operations in a distributed computing environment by **conventional computer components**, such as database servers, application servers, routers, security devices, firewalls, clients, workstations, memory storage devices, display devices and input devices. Each of these **conventional distributed computing components** is accessible via a communications network, such as a wide area network or local area network. ’137 patent at 5:12-20
- The present invention also includes computer programs that embody the functions described herein and illustrated in the appended flow charts. 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. Further, a skilled programmer would be able to write such a computer program to implement the disclosed invention based on the flow charts and associated description in the application text, for example. Therefore, disclosure of a particular set of program code instructions is not considered necessary for an adequate understanding of how to make and use the invention. ’137 patent at 5:32-44.

- The validation module 108 can be any one of a variety of pieces of software that are used to verify that a program has not been tampered with. For example, the validation module can represent the MD5 software module commonly known to those in the art. The MD5 software module calculates a checksum for each allowed program and that checksum is stored for later comparison. As described in connection with Fig. 2, the checksum data 205 can be stored in database 110 in step 320. ’137 Patent, 8:7-19.

3. The ’441 Patent AAPA

- The presently disclosed technology also may be embodied as an additional method, apparatus and/or system for a runtime monitor to inspect web and non-web applications (e.g., all web and non-web applications) running on the instrumented target platform. The system may include an attestation service (or broker) using an attestation server and configured to generate application artifacts and/or to issue application statements or claims. The application artifact may represent an identified application instance (e.g., at runtime, globally unique, opaquely and/or in a time-sensitive manner, for example, using an application artifact having a specified or predetermined lifetime) on an instrumented platform (such as but not limited to a, Personal Digital Assistant (PDA), a tablet computing device, an iPhone ™, an iPod™, an iPad™, a device operating the Android operating system (OS) from Google Inc., a device running the Microsoft Windows® Mobile OS, a device running the Microsoft Windows® Phone OS, a device running the

Symbian OS, a device running the webOS from Hewlett Packard, Inc., a mobile phone, a BlackBerry® device, a smartphone, a hand held computer, a netbook computer, a palmtop computer, a laptop computer, an ultra-mobile PC, a portable gaming system, a gaming console, or another similar type of computing device having a computer, microprocessor). The application statements may include at least one statement (or claim) about an inspected runtime execution context and/or an intelligence based security context.. ’441 Patent, 1:52-2:11

- Many enterprises are moving there IT infrastructures to cloud computing environments (e.g., in which third parties may provide shared computing resources and applications running on those resources are offered as services to a plurality of customers) for various reasons including: (1) increased computing power; (2) increased storage capacity, and/or (3) increased network bandwidth, among others. Enterprise applications and mission critical applications may be executed in the cloud. Without adequate system and application security, the cloud can compromise these applications causing large financial losses. Data confidentiality, data integrity and data availability can be maintained in the cloud computing environment even when such applications may be controlled by these third parties. For example, enterprises are moving from self-managed on-premise data centers to service-provider managed outsourced virtual data centers (e.g., the cloud computing environments). The cloud may enable traditional information technology systems and applications to transparently execute from these service-providers managed outsourced virtual data centers. As newer technologies emerge, the cloud infrastructure may evolve transparent to and scalable with enterprise operations. Security in such environments may be an emerging challenge. ’441 Patent, 5:45-67.
- Applications that are hosted in self-managed and in-house enterprise data centers may be gradually migrating to outsourced service provider managed virtual data centers in

private/public clouds. There may be a loss of visibility and direct control from this migration. Various exemplary embodiments include instrumentation to monitor the applications in process to increase visibility and enable attestation of applications executing in the outsourced service provider managed virtual data centers. ’441 Patent, 6:20-32.

- The attestation broker 109 (or the collaboration service 110) may be configured to generate application statements or reports 117, which may include assertions about the running target application, and may deliver the statements or reports 117 to the requestor, the third party, and/or end-user. The platform 100 may include any computing device 101 and/or virtual platform, for example, (e.g., a computer, microprocessor, Personal Digital Assistant (PDA), smartphone, notebook computer, and/or a virtual server running on hardware, among others). The computing device may include a microprocessor 130 for execution, for example, of computing platform software such as operating system software, applications, and runtime monitor software, among others and a memory 140 for data, metadata, and software storage. ’441 Patent, 7:13-26.
- The application artifact may include previous assertions or may provide an identifier that associates the application to the previous assertions. The application execution context 105 may include at least a base (or static), temporal (or dynamic) and/or environmental aspects or attributes of the target application. A requestor (for example, a network access enforcer 113, such as a network firewall, a load balancer, a VPN gateway, and/or an identity provider 112, such as an Security Token Service (STS), Single Sign On (SSO) solution may subscribe for application statements or reports 117 from the attestation broker 109. ’441 Patent, 7:61-8:4.
- The application statements or reports 117 may include attribute value assertions that may describe the runtime execution context 105 and application security context 111 of the

application instance on the instrumented platform 100, as a machine identifier. The application identifier may be a publisher designated product name or a registered service principle name in a services directory. The machine identifier may be an IPv4/IPv6 address, a directory network services (DNS) hostname, a MAC address, a serial number (e.g., an IT or manufacturer issued serial number), an RFID, and/or other device identifiers, among others. ’441 Patent, 8:33-46.

- In certain exemplary embodiments, the application statements or reports 117 may be used by passive clients 116 (for example, a web browser) to render an icon or other indicator (either visual on the web page or audio indicator, for example, an icon at the bottom of the web page). The indicator may be an indicator (or an attestation) of the web application's authenticity to the user 115. For example, the user 115 may click on the attestation icon to view web application specific information issued by the attestation broker 109 for the accessed web application. The attestation broker 109 may be a third party independent and trusted entity that provides such attestation services. The user 115 may confirm that the application attestation offered by the attestation broker 109 is current (e.g., that the lifetime of the application attestation has not expired). ’441 Patent, 8:60-9:7.
- A network access enforcer 113 may subscribe with the attestation broker 109 for application statements or reports 117 and may discover a plurality of target applications 103 or 104 on a plurality of target machines or platforms 100. The communication protocol for the communications 118 between the attestation broker 109 and the network access enforcer 113 may be any standards based protocol and message exchange data structures or schemas, such as for example: (1) Simple Object Access Protocol (SOAP); (2) Representational State Transfer (REST); or (3) Trusted Computing Group's (TCG's) Interface for Metadata Access Points (IF-MAP)

specification, among others. ’441 Patent, 10:47-58.

- An active client application 116 and/or a server application 104 (e.g., using the Simple Authentication and Security layer (SASL) protocol) may request and/or use the received application statements or reports 117 programmatically for a mutual attestation handshake. For example, using SASL protocol such a mutual attestation handshake may be defined by an integrity exchange profile, before initiating an authentication handshake with proof of possession of credentials. 11:4-11
- The web application 302 may include (for example, may embed, as a markup) the received application artifact 309 as an assertion (e.g., in a security assertion markup language (SAML or other common form of markup)) to the identity provider 303 which may use back-channel communications 310 with the attestation broker 304 to verify the application artifact 309. 12:57-63
- The application statements or reports 316 may include, for example, information about the user's identity, one or more authentication factors such as password, PIN, smart card, among others, and preexisting roles in a user directory (for example, Active Directory, among others), a role based management system (for example, Oracle or SAP, among others) and/or a relational database (for example, MS-SQL, MySQL, or Oracle DB, among others). 13:47-55
- Although a Simple Authentication Security Layer (SASL) protocol is illustrated, it is contemplated that any application layer protocol between client-server and peer-to-peer applications, or network security protocols, for example IPSec Authenticated Internet Protocol (AuthIP) or Internet Key Exchange (IKE), may be extended and/or used to provide the functions of the client application 500 and the server application 501. In certain exemplary embodiments, the client application 501 and the server application 501 may be statically or dynamically linked to the API library 516 (e.g., a

Trust-as-a-Service (TAAS) API library). The API library 516 may implement a set of procedure calls for applications to exchange messages (e.g., opaque and/or secure messages) with an attestation service or broker 502 that may hide the implementation details from the higher level application. 15:52-67

- In alternative implementations, secondary memory 1410 may include other similar means for allowing computer programs or other instructions to be loaded into computer system 1400. Such means may include, for example, a removable storage unit 1422 and an interface 1420. Examples of such means may include a program cartridge and cartridge interface (such as that found in video game devices), a removable memory chip (such as an EPROM, or PROM) and associated socket, and other removable storage units 1422 and interfaces 1420 which allow software and data to be transferred from the removable storage unit 1422 to computer system 1400. Computer system 1400 may also include a communications interface 1424. Communications interface 1424 allows software and data to be transferred between computer system 1400 and external devices. Communications interface 1424 may include a modem, a network interface (such as an Ethernet card), a communications port, a PCMCIA slot and card, or the like. Software and data transferred via communications interface 1424 may be in the form of signals, which may be electronic, electromagnetic, optical, or other signals capable of being received by communications interface 1424. These signals may be provided to communications interface 1424 via a communications path 1426. Communications path 1426 carries signals and may be implemented using wire or cable, fiber optics, a phone line, a cellular phone link, an RF link or other communications channels. 24:54-25:12
- Embodiments of the invention also may be directed to computer program products comprising software stored on any computer useable medium. Such software, when

executed in one or more data processing device, causes a data processing device(s) to operate as described herein. Embodiments of the invention employ any computer useable or readable medium. Examples of computer useable mediums include, but are not limited to, primary storage devices (e.g., any type of random access memory), secondary storage devices (e.g., hard drives, floppy disks, CD ROMS, ZIP disks, tapes, magnetic storage devices, and optical storage devices, MEMS, nanotechnological storage device, etc.), and communication mediums (e.g., wired and wireless communications networks, local area networks, wide area networks, intranets, etc.). 25:42-56

4. The ’517 Patent AAPA

- A variety of programs and approaches have been developed to provide security and protect end users from harmful threats. Such programs that have been developed include antivirus programs, firewalls, and intrusion detection/prevention systems (IDSs/IPSs). These programs can be beneficial in protecting a computing system and its end user from a number of threats. ’517 Patent, Background.
- The computing system 100 may include a processing unit 102. The processing unit 102 may include a single processor or a plurality of processor, each of which may include one or more processor cores. The processing unit 102 may be a general purpose processing unit or a special purpose processing unit, such as a general purpose processing unit programmed for performing a specific purpose, such as the identification of the runtime risk of an application or program. 2:55-64
- The communications infrastructure 110 may be a bus, message queue, network, multi-core message-passing scheme, a combination thereof, or any other suitable type or configuration of communications infrastructure as will be apparent to persons having skill in the relevant art. The computing system 100 may further include a display unit 104. The display unit 104 may be configured to control a display device 106, which

may be connected to the computing system 100 physically (e.g., via a cable, such as a VGA, DVI, or HDMI cable) or wirelessly (e.g., via Bluetooth, etc.). The display unit 104 may be a video card, video adapter, graphics card, display card, graphics board, display adapter, graphics adapter, video controller, graphics controller, etc. and may be integrated into the computing system 100 or may be removable. 2:66-3:12

- Suitable types of display devices for use as the display device **106** will be apparent to persons having skill in the relevant art and may include a liquid crystal display (LCD), light-emitting diode (LED) display, thin film transistor (TFT) LCD, capacitive touch display, etc. 3:16-20
- The memory unit 106 may be any type of memory suitable for the storage of data and performing of the functions disclosed herein, such as a hard disk drive, floppy disk drive, magnetic tape drive, optical disk drive, solid state drive, or other non-transitory computer readable medium. In some embodiments, the memory unit 106 may be removable storage (e.g., flash memory, a compact disc, digital versatile disc, Blu-ray disc, etc.) or a combination of non-removable and removable storage. In one embodiment, the memory unit 106 may be external to the computing system 100 and accessed via a network by a communications interface 108, discussed in more detail below, such as cloud storage. The memory unit 106 may include random access memory (RAM), read-only memory (ROM), or a combination thereof. Suitable types and configurations of the memory unit 106 will be apparent to persons having skill in the relevant art. 3:22-37
- The runtime monitor 108 may be configured to perform additional functions in the computing system 100, such as those described in U.S. Pat. No. 8,372,441, filed as U.S. application Ser. No. 13/399,065, entitled “System and Method for Application Attestation,” filed on Feb. 17, 2012; U.S. application Ser. No. 13/559,707, entitled “Systems and Methods for Orchestrating Runtime Operational Integrity” and filed on Jul.

27, 2012; U.S. application Ser. No. 13/559,766, entitled "Systems and Methods for Threat Identification and Remediation" and filed on Jul. 27, 2012; U.S. application Ser. No. 13/559,665, entitled "Systems and Methods for Providing Mobile Security Based on Dynamic Attestation" and filed on Jul. 27, 2012; U.S. application Ser. No. 13/559,692, entitled "Systems and Methods for Using Reputation Scores in Network Services and Transactions to Calculate Security Risks to Computer Systems and Platforms" and filed on Jul. 27, 2012; and U.S. application Ser. No. 13/559,732, entitled "Systems and Methods for Network Flow Remediation Based on Risk Correlation" and filed on Jul. 27, 2012, which are herein incorporated by reference in their entirety. 3:56-4:8

- The rules database 110 and the policy database 112 may be configured in any type of suitable database configuration, such as a relational database, a structured query language (SQL) database, a distributed database, an object database, etc. Suitable configurations and database storage types will be apparent to persons having skill in the relevant art. The databases may each be a single database, or may comprise multiple databases which may be interfaced together, such as physically (e.g., internally via the communications infrastructure 110 or externally via the communications interface 108) or wirelessly (e.g., via Bluetooth), and may include one or more databases included within the computing system 100 and one or more databases external to the computing system 100, such as an external hard disk drive or storage accessed via a network (e.g., in the cloud). 4:28-42
- The communications interface 108 may be a modem, network interface card (e.g., an Ethernet card), a communications port, a Personal Computer Memory Card International Association (PCMCIA) card, or other type of communications interface suitable for performing the functions disclosed herein as will be apparent to persons having skill in the relevant art. Software and data transmitted to or from the computing system 100 may be in the form of signals, which may be electronic, electromagnetic, optical, etc. The

signals may travel via a communications path 114, which may be configured to carry the signals physically or wirelessly via a network. For example, the communications path 114 may carry signals from the communications interface 108 to a network such as a local area network (LAN), a wide area network (WAN), a wireless network (e.g., WiFi), a mobile communication network, a satellite network, the Internet, fiber optic, coaxial cable, infrared, radio frequency (RF), or any combination thereof. Other suitable network types and configurations will be apparent to persons having skill in the relevant art. 4:45-64

- For example, the communications interface 108 may include a plurality of universal serial bus (USB) connectors, an Ethernet connector, audio connectors, video connectors, etc. Suitable types of input devices that may be used with the computing system 100 for providing input will be apparent to persons having skill in the relevant art and may include a keyboard, mouse, touch screen, tablet, click wheel, trackball, microphone, camera, etc. 5:4-12
- The user action 302 may be any form of explicit input from a user of the computing system 100. For example, the user action may include a program launch, keyboard input, click of a mouse, touch input on a touch screen, moving of a mouse, scrolling or clicking on a click wheel, voice command into a microphone, gestures into a device configured to read human gestures, etc. 6:56-62

5. The'038 Patent & '918 Patent AAPA[30]

- Increasingly, office files, design documents, employee work products, company information, and most other important business information is being created and stored electronically on desktop computers, laptop computers, handheld computing devices

[30] These patents share a common specification and therefore only one citation will be given for these patents.

(collectively ‘personal computing device’ or ‘computing device’) and company networks. At work, employees access such networks, along with their associated corporate computing resources from their local computing device, on a daily basis in order to perform their jobs. Away from work, employees similarly access such networks and resources, typically through remote connections. Numerous types of electronic connections are ubiquitous in the industry and well known to the reader, for example: dial up connections, wireless connections, high-speed connections of various types, virtual private network connections, and others. ’038 Patent, Background.

- Endpoint access controls have followed an incremental, evolutionary path. Prior to the storage of sensitive data and the recognition of the security issues associated therewith, there were no endpoint access controls. However, security issues such as data theft, unauthorized access, fraud, etc., and the resulting concerns, created an industry-wide demand for security solutions. *Id*.
- The first generation of endpoint access control included operating system services that controlled user access to one or more system resources, such as applications, data files, configuration settings, etc. Users were permitted or denied access to these resources based on a variety of factors, such as their login ID (which was authenticated using a secret) and a secured profile of policy settings identifying permissions and/or restrictions. These permissions were generally static in that they were not context sensitive in any other dimension than the user ID. There was no consideration of environmental factors. This static nature of security services embedded into the operating system remains relatively unchanged in many environments, to the present day. *Id*.
- In the next step in the evolutionary path of endpoint security control, a series of point solutions were created that address point security concerns by providing point access control capabilities. Examples of these point solutions include: personal firewalls that

restrict inbound and/or outbound access to specified applications, ports, addresses and/or communication protocols; antivirus agents, anti-spyware agents and application white-list management agents that monitor, detect and/or restrict access to specific system resources such as memory, registry keys, etc.; software update agents that automatically update an application if it is not a specified version; data encryption agents that encrypt specific files, the complete contents of specific folders, etc.; and physical access control agents that restrict access to floppy drives, USB drives, CD-ROM drives, etc. These security agents are one-dimensional in that they look at a single aspect of the endpoint's security posture and make decisions on that basis. There is no integration of data across these security agents—all of these security solutions operate autonomously and completely independent of each other, with little or no communications between them or awareness of the state of other applications running on the endpoint. As with operating system security services, these point solutions are also static. The business logic and configurations of these point solutions are not context sensitive. They typically apply the same rules regardless of the user ID, user location, time of day, presence or absence of other security applications on the endpoint, configuration and state of other security or management applications on the endpoint, etc. While providing relatively stable and secure access control, such static endpoint controls remain inflexible and not adaptable to user and business needs. They are very much in use today in many environments. *Id*.

- In the most recent evolutionary step, context awareness has been introduced into the field of endpoint security control. Functional examples of context awareness capabilities on the market today include: if a named application is not running or is not of a specified minimum version, access to network connectivity or certain applications will be restricted or blocked altogether; if a user is in location X (as determined by an assigned IP address, reachability of a network host, or some other method of automated location

determination), the user is permitted outbound access using application X and Y to network servers on subnet Z, however if the user is in location Y (alternatively an unknown location), the user is permitted outbound access using application X and W to network servers on subnet V. In each of these examples, access to a resource (in the first case an application, in the second case the network and communications protocols) is context sensitive in the sense that the access privilege is conditional on the current state of the endpoint (in the first case a certain application running, in the second case the current location). However these solutions are limited in that they are only able to assess a limited set of inputs and affect a narrow set of access privileges. Additionally, once an access privilege has been granted, the decision is rarely revisited over the life of the user's connection or access session, i.e. they could come out of compliance subsequent to granting of access and will still retain access. *Id*.

- In accordance with the present invention and **conventional use**, host system 102 comprises a secure, access-controlled processing system where-to remote systems such as endpoint system 104 connect to access data, processing capacity and host-accessible resources. Policy management system 106 provides rules and policies concerning the connection of remote endpoint systems 104 to host system 102. Host system 102, endpoint system 104, and policy management system 106, are interconnected to communicate through a **conventional electronic network** 108, such as the Internet. 7:11-23
- Considering in detail host system 102, the system is seen to include, in a **conventional manner**, a processor and user & communications interface 102A, as well as conventional storage components 102B, operating systems and software (typically contained in storage and operated by the processor) and other conventional components. Further associated with host system 102 are a variety of resources, indicated at 102G, accessible directly or

indirectly through the host, including, for example: user data, user applications, physical ports, data storage devices, dial adaptors, network interfaces, and other resources as will be apparent to the reader. Further contained within host 102 are a plurality of conditions 102F. These conditions are monitored by agents 102E, the agents collecting and transmitting information to agent managers 102D for aggregation by agent monitor 102F. The various conditions, as well as the agent functions, are described in detail herein below. Host 102 may comprise, for example, a processing system of the type typically owned, managed and/or operated by a business to support the operation of its employees. It may comprise a server, enterprise system, personal computer, laptop, personal digital assistant, mobile communications device such as a 'smart' telephone, or any other type of remotely accessible system. In a **conventional manner**, host system 102 may include **conventional security features** for controlling access to the data and resources thereon. 7:24-49

- Continuing with reference to FIG. 1, endpoint system 104 comprises any processing system capable of interconnecting with host system 102, for example: a laptop computer, personal computer, server system, enterprise system, personal digital assistant, cellular telephone, 'smart' telephone or other personal device, or any other processing system capable of remotely accessing host system 102 for the purpose of accessing the resources available there on. Endpoint system 104 is seen to include, in a **conventional manner**, a processor and user & communications interface 104A, as well as conventional storage components 104B, operating systems and software (typically contained in storage and operated by the processor) and other **conventional components**. 7:52-64
- Considering now the details of policy management system 106, in the illustrated embodiment, the system comprises a **conventional processing system**, for example a server computer, enterprise computer, personal computer or a notebook computer.

Accordingly, the system is seen to include, in a **conventional manner**, a processor and user & communications interface 106A, as well as **conventional storage components 106B, operating systems and software (typically contained in storage and operated by the processor) and other conventional components**. In accordance with the present invention, policy management system 106 is seen to include a compliance analysis engine 106C as well as various policy information stored within storage system 106B. As will be seen from the description below, compliance analysis engine 106C, typically comprising software in data store 106B running on hardware 106A, functions to receive system condition information and process that condition information in accordance with the security policies, such as are stored within data storage 106B, in order to generate security rules. Analysis engine 106C can comprise a portion of the capacity of processor 106A and/or one or more dedicated and/or shared separate processor(s). 8:4-25

- Many computer hardware and software vendors are known to maintain a running list of known security vulnerabilities in their products. See for example vendor Web sites:
  - www.microsoft.com/technet/security/alerts/matrix.mspx
  - www.cisco.com/en/US/products/products_security_advisories_listing.html
- As used herein, references to software and software programs to describe a security vulnerability are to be interpreted in their broadest sense, including software such as application programs, operating systems and drivers, combinations of software and hardware and hardware.
- Because each vendor has their own terminology, definitions and subjective view of what constitutes a vulnerability and the degree of risk or exposure a given vulnerability represents (i.e. its severity), there are several industry initiatives to standardize vulnerability definitions and scores. See for example:

- www.kb.cert.org/vuls
- www.first.org/cvss/cvss-guide.html

- Depending on the source, information that may be published about each vulnerability includes information such as descriptive parameters that describe the hardware or software at risk (e.g. Intel-based hardware running Windows XP Service Pack 2), possible system impacts (e.g. memory buffer overflow, unauthorized remote control of the computer, etc), severity type, severity level, sources of more information, date vulnerability was first reported, etc.
- Vendors often use this information to prioritize their responses to vulnerabilities in their products. Responses typically take the form of customer notifications, often accompanied by specific interim remedial actions to take (e.g. disable a service, shut down a TCP port, etc.) and/or information on currently available patches that can be applied to eliminate the vulnerability.
- When there is no current software available to eliminate the vulnerability, the vendor will normally begin scheduling internal activities to develop a solution to the vulnerability and make the solution available to customers and product users as a 'patch' or 'update'. Once this becomes available, customers may receive notification, and/or find notification information on a vendor's web site.
- Information technology (IT) managers, also referred to herein as administrators, access vulnerability information by either receiving a notification from a vendor or industry group, going to the vendor or industry web site and querying the vulnerability database, or by establishing an electronic communications link with the remote database and electronically receiving vulnerability database updates on a periodic basis. IT managers typically use a combination of industry risk assessment and vendor risk assessment information to prioritize which vulnerabilities and patches to focus on first, and to

prioritize remediation activities relative to other routine IT operating activities and other IT projects. 21:49-22:34

- The interval of time between when a vulnerability is announced and when a vulnerable endpoint is patched commonly referred in the IT industry to as an ‘exploit window’, i.e. a window of time in which a security attack that specifically, opportunistically targets that publicized vulnerability can be created and used to probe endpoints to find vulnerable ones that can be attacked. During the exploit window, the endpoints remain exposed to a security attack unless some temporary securing action is taken to protect the endpoint. Attack exposure may be from the local machine only, from a remote machine, or both, depending on the nature of the vulnerability. The attack may utilize only the new exploit or more commonly utilize a combination of exploits to gain control of the system, gain reliable access to the system, take an action on the local system, or have the local system initiate a communications session with a remote computer of the hacker's choosing. 22:47-63
- The following sections describe some of the analytical models used by analysis engine 106C. Policy management system 106 is designed to allow analytical models operated by analysis engine 106C to be added in the future, individually upgraded or modified, or removed. **Conventional** software distribution methods are used to communicate new or modified analytical models and new versions of the analysis engine 106C. In accordance with the present invention, analysis engine 106C is also architected to allow the inputs and/or actions associated with a given policy to be modified or customized as required. **Conventional** software distribution methods are used to communicate new or modified policies or policy values. Policies incorporating combination rules are also supported through the logical combining of multiple individual rules using conventional logic clauses such as AND, OR, NOT, ELSE, IF,

WHEN, UNLESS, etc. 26:50-65

- One analytical model operable by analysis engine 106C involves treating endpoint condition state information 104F as a matrix of numeric values where as mentioned above and as implied in each of the subsequent analytical models described herein, the real time state information is converted to numerical values or risk weightings, e.g. 1-100. The standalone and business intelligence rules can be treated as a second matrix where rules are given relative importance ratings. By combining the two matrices using **conventional** matrix mathematics, the analysis engine 106C generates a third matrix as the result. This third matrix contains numerical compliance scores that can be converted to security compliance ratings for different enforcement actions.
- The security score thresholds, the input matrix elements, the input matrix security scores and the items to be included in the endpoint inputs list are all data values stored in the policy data store 106B and as such are configurable and extensible so as to allow tailoring to an individual user's need. Configuration is performed using a user interface 106A, from which new or revised matrix elements, thresholds, weightings and factors can be created and modified. When implemented in a distributed fashion, changes to these data values made in the policy management system 106 can be distributed to the software agent residing on the endpoint system 104 using **conventional** software distribution methods. 28:55-66
- The analysis engine 106C can be configured via policy settings to send a message to an administrator via a **conventional** data communications network and a commonly available data communications protocol, (e.g. via POP, SMTP, FTP, HTTP, etc.) when a specific policy event occurs, for example a specific noncompliance condition. 59:5-10

6. The ’948 Patent & ’616 Patent AAPA[31]

- Traditional legacy and currently available security technologies, such as next-generation anti-virus software, network firewalls, and intrusion detection/prevention systems, operate based on signatures, protocol anomalies or virtual execution in a sandbox to monitor threats and attacks. However, once a target of an attack or threat (i.e., a ‘target system’ or ‘target platform’) becomes infected or compromised, many of these technologies have been proven relatively ineffective in protecting systems and preventing data breaches or service disruptions. Emerging threats and attacks can exhibit a low and slow mode of operation and can be signature-less in order to evade traditional detection and defense technologies. Further, these technologies are often agnostic to the runtime operational integrity of computer workloads on the target (victim) systems and therefore often do not offer any level of remediation to service the affected target systems. As such, many traditional security technologies merely provide coarse-grained access controls limited to a simple allow or deny decision logic. ’948 Patent, Background
- A significant metric that relates to the lack of remediation controls in current security controls is the high rate of false positives and negatives in the detection of threats. False positives are unproductive and reduce operating capacity in data centers. False negatives lead to the bypass of security controls and compromise of the target systems and services. Existing signature-based approaches to security control are vulnerable to improvised attacks staged at targeted systems.. *Id.*
- The current predominant security paradigm is based on a hard edge and soft core architecture. Security appliances such as network firewalls and intrusion detection/prevention systems deploy at the edge (perimeter). Antivirus and network based

[31] These patents share a common specification and therefore only one citation will be given for these patents.

integrity measurement and verification services scan and audit the soft-core that comprises business critical systems, services and high value data silos. Once the edge is breached, defensive methods are largely ineffective in protecting vulnerable or compromised systems. In this paradigm, the hard edge does not generally provide any level of assurance of the runtime operational integrity of the soft core. *Id*.

- The heavy reliance on extensive application whitelists/blacklists (based on file hash digests and/or configuration), Internet Protocol (IP) address reputation lists, signatures, protocol anomalies and virtual execution in a sandbox, provides targeted and coordinated attacks a large staging surface to maneuver around with meticulously engineered evasion methods exploiting the gaps. Irrespective of the method of break-in, the post-infection behaviors during the window of exposure on a victim machine or environment are difficult to conceal, but may be obscured from timely detection and diagnosis by security administrators and virtual execution (sandbox) environments through evasion techniques. Various exemplary embodiments include an early warning system for threat identification and diagnosis with high forensic confidence and infection summaries for manual and/or automated intervention. *Id*.
- Current identity management (IdM) approaches are typically aimed at domain level authentication of users for the use of a security token in a transaction or business process wherein the issuance of the token is strictly based on proof of possession of credentials and not the reputation of the user. While the industry has adopted multi-factor authentication technologies for increased protection from malicious users whose intent is to steal credentials for the purpose of impersonation or delegation, these security controls do not provide a means for incorporating attribution of an authenticated user's risk posture, independent of provisioned entitlements (i.e., user account privileges and roles). Thus, existing security controls do not incorporate a complete view of a user's risk

posture as a component of an access policy, based on correlation of location and device agnostic global threat intelligence about the user. Accordingly, what is needed are systems and methods for runtime operational integrity monitoring of applications by providing dynamic operational integrity attestation of application security and user reputation at runtime that take into account a subject's (such as a device, application, or user) risk posture. What is further needed are systems and methods for performing post infection diagnoses, threat identification via signature-less anomalous behavior recognition using non-intrusive platform instrumentation without requiring 'hooks' into an operating system (OS) Kernel by security vendors that leverages OS vendor application programming interfaces (APIs). *Id*.

- The emerging cloud based application-hosting model requires a higher level of scrutiny of device and user security posture. Threats can be posed by both external and internal users, and therefore it has become harder to determine whether an authenticated user is a trustworthy operative. Application and network layer entitlements are largely static in nature. While role based access control (RBAC) systems are positioned to offer dynamic policies, these are not based on any form of aggregation of external threat intelligence in the context of the user in the transaction. *Id*.
- Tokenization of identities by Security Token Services (STS) intended to facilitate in Identity Federation and Single Sign On (SSO) for web and Enterprise applications have only aggravated the threat vectors by expanding the scope of resources that an authenticated user may access implicitly without granular context. Along with the coarse access policies offered by traditional security controls, such as firewalls and intrusion prevention systems, this increases the attack surface for application exploits by malicious users leveraging the excessive privileges granted by default to most users (and groups) today, in the absence of strict Separation of Duties (SoD) and Principle of Least Privilege

(PoLP) enforcement. *Id*.

- Current attestation systems are components of computer systems that permit reliable statements of evidence about those systems to be conveyed to remote parties, including other computers. Through evaluation of the identity and integrity of components of a system (i.e., a target system being evaluated), evidence is produced that the target system will not engage in defined classes of misbehaviors. As the users accessing a system and the requirements of applications running on a system generally cannot be known a priori, attestation systems and measurement systems alike must be flexible, providing for privacy, completeness of measurement, and trust in the basic collection and reporting mechanisms. *Id*.
- Existing attestation systems are often narrowly focused and generally aimed at specific use-cases limited to system components such as hardware and applications and therefore typically lack flexibility to dynamically address more general attestation problems. Further, existing definitions of attestation focus primarily on describing specific, narrow, and particular properties desirable in specific use-cases. Additionally, current attestation systems are created to work with one particular measurement system targeting one particular system of interest without considering the reputations of users or operators of the system of interest. *Id*.
- Exemplary systems and methods for dynamic attestation of application integrity are described in U.S. Non-Provisional patent application Ser. No. 13/399,065 entitled "System and Method for Application Attestation," filed Feb. 16, 2012, and U.S. Provisional Patent Application No. 61/443,854 entitled "System and Method for Application Attestation," filed Feb. 17, 2011, both of which are incorporated herein by reference in their entireties. 4:52-59
- As will be appreciated by persons skilled in the relevant art(s), a VM is a software

implementation of a machine such as a server, personal computer, mobile computing device, or other computing device that supports the execution of an operating system (OS) and executes applications as a physical machine running that OS would. A VM is a software implementation that duplicates the functionality of a physical machine implemented in hardware and software. Software applications and the OS running on a VM are limited to the resources and abstractions provided by the VM. Virtual machines (VMs) can be viewable within an overall virtual infrastructure. As will be appreciated by those skilled in the relevant art, a VMM or hypervisor can be used to start up, monitor, and manage VMs. Such hypervisors can be, but are not limited to VMMs such as the VMWARE™ Player, MICROSOFT™ VirtualPC, SUN™ VirtualBox, VMWARE™ ESX/ESXi, MICROSOFT™ Hyper-V, CITRIX™ XENServer, PARALLELS™ and others. As it would be apparent to one of skill in the art, other hypervisors and VMs/virtualization solutions can be used for VM 172 in network 100 as well. 8:47-67

- The method 300 can operate in conjunction with legacy security technologies 305 to fill in the security holes or gaps inherent in traditional security solutions. As illustrated in FIG. 3A, such legacy security technologies 305 can include anti-virus software, network firewalls, and intrusion detection/prevention systems (IDS/IPS). Unlike the early warning system 301 and the automated and rapid response 303, the legacy security technologies 305 are limited to using signatures, detecting protocol anomalies or virtual execution in a sandbox to monitor threats and attacks. As shown in FIG. 3A once a target of an attack or threat (i.e., a 'target system' or vulnerable system 307) becomes infected or compromised, these technologies 305 can be ineffective in protecting the compromised/infected systems 307 and preventing data breaches or service disruptions. 9:49-63
- Example limitations of legacy security technologies 305 are shown in FIG. 3A. For

example, as depicted in FIG. 3A, a traditional network firewall technology 305 using access control lists (ACLS) cannot reliably block every threat. This is because firewalls are typically limited to monitoring well-known, default open ports used as listening ports for web protocols such as the Transmission Control Protocol/Internet Protocol (TCP/IP), email protocols such as the Simple Mail Transfer Protocol (SMTP) and the Post Office Protocol (POP), and other protocols such as the File Transfer Protocol (FTP), the Secret File Transfer Protocol (SSH), Secure FTP (SFTP), telnet, and the Remote Desktop Protocol (RDP). IDS and IDP technologies using rulesets cannot detect every zero-day signature/anomaly attack and have issues with regarding false negatives and leaving platforms/systems vulnerable to backdoor attacks. Similarly, anti-virus software cannot detect zero-day, mutant, and low-and-slow infections. 9:64-10:13

- For example, an organization may periodically run security audit scans (i.e., on a weekly or bi-weekly basis). Such security audit scans may be National Institute of Standards and Technology (NIST)/Security Content Automation Protocol (SCAP) compliant checks for common vulnerability exposures to identify deviated and vulnerable systems 307. The results from these scans may be used as part of manual IT processes for ticketing (i.e., creating 'trouble tickets' for deviated/vulnerable systems 307) and security/compliance review boards. However, as shown in FIG. 3A these manual processes triggered by merely periodic scans can takes days or weeks to react to threats, thus leaving target systems open to a long window of exposure. 10:15-28
- With continued reference to FIG. 4, the temporal events 409 can be an annotation (e.g., an XML representation) of the configuration state of the system (including device, application and package) and include trust and severity scores as generated by third party security vendors based on integrity measurement and verification, in accordance with standards established by organizations. These organizations can include NIST, the

MITRE Corporation, and the United States Computer Emergency Readiness Team (US-CERT). The temporal events can be assertions about operational parameters expressed in the Open Vulnerability and Assessment Language (OVAL) to convey system details representing configuration information and a system/machine state including vulnerability, configuration, patch level, etc. Alternatively, the system details can be conveyed using the SCAP protocol. According to embodiments, the state representation includes schema attributes to measure configuration, vulnerability (exposures), compliance, and patch level based on severity. Compliance can be expressed in terms of the Extensible Configuration Checklist Description Format (XCCDF) using XCCDF checklists for devices attached to a system's Peripheral Component Interconnect (PCI) bus, and data security/system integrity compliance with the Health Insurance Portability and Accountability Act (HI PAA), Sarbanes-Oxley Act (SOX), and/or the Gramm-Leach-Bliley Act (GLBA). 11:59-12:17

- As shown in FIG. 4, the integrity reports 406 can be generated by performing one or more scans, including, but not limited to, a vulnerability scan 402, a configuration scan 403, a compliance scan 404, and a patch scan 405 of networked endpoints. In an embodiment, the trust broker 407 can be implemented as the Trust-as-a-Service (TaaS) Broker from TAASERA, Inc. In one exemplary embodiment, the trust orchestrator 430 can be implemented as the TaaS Orchestrator from TAASERA, Inc. 12:18-26
- In the embodiments of FIGS. 4-6, the integrity profile 420 represents an aggregation of system warnings (threats such as malware) identified based on the received temporal 409 and endpoint 520 events. In one embodiment, the format (schema) of the integrity profile 420 is a standard Extensible Markup Language (XML) notation. 12:39-44

- In embodiments, the device 560 depicted in FIG. 5 may be a desktop, server, laptop, a mobile computing device such as a smart phone or tablet, or other types of mobile computing platforms that comprise (a) an operating system (OS); (b) a user operating the device to access a service or another device 560 over a network; and (c) application packages, applications or application components that may be downloaded and installed either with or without user intervention for later execution. Non-limiting examples of the device 560 include a personal computer, server, laptop, or tablet device running a MICROSOFT™ WINDOWS® operating system (OS), a personal computer, server, or laptop running an OSX OS from Apple Inc., a personal computer, server, or laptop running a UNIX or Linux OS, a personal digital assistant (PDA), an iPhone™, an iPod™ touch, or iPad™ tablet device running an iOS from Apple Inc., a device operating the Android OS from Google Inc., device running the MICROSOFT™ WINDOWS® Mobile or Phone OS, a device running a Symbian OS, a device running a PALM OS®, a BLACKBERRY® device running a Blackberry OS from Research In Motion ("RIM"), a mobile phone, a portable game device, a gaming console, a hand held computer, a netbook computer, a palmtop computer, an ultra-mobile PC, or another similar type of computing device capable of processing instructions and receiving and transmitting data to and from users and other computing devices. 13:17-42
- In embodiments, the native machine instrumentation 511 can represent any event subscription, callback, notification mechanism provided by the supported operating system (OS) on the device 560 (e.g., MICROSOFT™ WINDOWS® Machine Instrumentation (WMI), Transport Filter Drivers, Application Layer Enforcement Callout Drivers, Linux/Unix Network Filter Drivers, etc.). 13:44-50
- With continued reference to FIG. 5, the application operational integrity system 500 can include a trust orchestrator 430 comprising a system event correlator 410 and a trust

broker 407. The application operational integrity system 500 can further include a malware analyzer 540. In an embodiment, the trust orchestrator 430 can be implemented as the TaaS Orchestrator from TAASERA, Inc. 14:32-38

- With continued reference to the exemplary embodiment provided in FIG. 6, the trust orchestrator 430 comprises a system event correlator 410 configured to use a risk correlation matrix 411 to generate an integrity profile 420 for the monitored system (i.e., the target system). In one embodiment, the system event correlator 410 can be implemented as a correlation engine. In another embodiment, the trust orchestrator 430 can be implemented as the TaaS Orchestrator from TAASERA, Inc. 15:29-37
- The directives are specific to the orchestration service and can include the use of vendor APIs. Examples of such vendor APIs include VMWARE™ vCloud APIs, BMC Atrium™ APIs for accessing a BMC Atrium™ configuration management database (CMDB) from BMC Software, Inc., Hewlett Packard Software Operations Orchestration (HP-OO) APIs, and standard protocols such as Open Flow. 18:9-14
- Examples of role management systems include commercial products available from Oracle, Computer Associates, IBM, and SAP for role based access controls to database objects, files, servers, directories/folders, web servers, applications, network domains/subdomains, printers, and other resources. In some of these systems, users are assigned one or more roles that aggregate a set of privileges and permissions. For example, roles in an Oracle or IBM relational database environment represent different tiers or levels of access to database objects. For example, a user account with a 'database administrator' (DBA) role may have privileges and permissions needed to delete (i.e., drop), create, and alter database objects such as tables, views, synonyms, stored procedures, triggers, user accounts, entire databases, and other roles. In contrast user

accounts with lesser roles, such as a developer or user role may only be able to insert and update data in database tables and views and revise existing stored procedure and trigger code. Yet other roles may be restricted to 'read only' access to select records from database tables and views as needed to generate reports and run queries. 21:32-51

- In accordance with embodiments, the identity management systems 903 shown in FIG. 9 can be configured to provide information pertinent to a user's organizational domain accounts and group level memberships (e.g., MICROSOFT™ Active Directory, Oracle Internet Identity). 21:53-57
- In embodiments, the log management systems 904 shown in FIG. 9 can be configured provide information pertinent to a user's activities at the endpoint device and at the network level of operations based on logs generated by endpoint and network elements and reported through standard system information and event management protocols (e.g., SYSLOG) and formats such as the WebTrends Enhanced Log file Format (WELF) or the Common Event Format (CEF) used for network security applications. 21:59-67
- In one embodiment, the queries 1121, 1130, 1122, 1128 issued by the trust broker 407 are preconfigured, but can be customized by the administrator from a graphical user interface (GUI) or dashboard, such as the exemplary GUI described below with reference to FIGS. 14-18. In embodiments, the management systems may include information systems hosted by social networking vendors such as FACEBOOK® operated by Facebook, Inc., Twitter operated by Twitter, Inc., and LinkedIn. 24:23-31
- The terms "console display," "display," "display screen," and "screen" are used interchangeably herein to refer broadly and inclusively to any type of display device or screen coupled to or integrated with a computing device for displaying content viewable by a user of the computing device, such as administrators for the systems describe herein or a user 1014 of a device 560. In an embodiment, the device 560 is a mobile computing

device 560. Such a display screen can include, for example and without limitation, a touch-screen liquid crystal display (LCD). In embodiments of the invention, the GUIs of a mobile device 560 is viewed on a display. In other embodiments, the GUIs shown in FIGS. 14-18 are viewed on a display of a server (i.e., a server console), a desktop computer (i.e., a PC monitor), or a laptop display. 26:47-60

- Examples of such computing device platforms include, but are not limited to, OSX server and workstation operating systems (OSs) from Apple, Inc.; WINDOWS® server and workstation OSs from the MICROSOFT™ Corporation; UNIX-based OSs, and Linux OSs, such as, but not limited to, Linux from RedHat™ Inc. 27:5-10
- In alternative embodiments, the GUIs of FIGS. 14-18 can be rendered on a display of a mobile computing device such as, but not limited to a personal digital assistant (PDA), an iPhone™, an iPod™ touch, or iPad™ tablet device, a device operating the Android operating system (OS) from Google Inc., a device running a MICROSOFT™ WINDOWS® Mobile or Phone OS, a device running a Symbian OS, a device running a PALM OS®, a BLACKBERRY® device running a Blackberry OS from Research In Motion ("RIM"), a smart phone, a hand held computer, a netbook computer, a palmtop computer, a laptop computer, an ultra-mobile PC, or another similar type of mobile device capable of processing instructions and receiving and transmitting data to and from users and other computing devices. 27:11-24
- As will be appreciated by persons skilled in the relevant art, WMI is a set of extensions to the MICROSOFT™ WINDOWS® Driver Model (WDM), which is a framework for device drivers that provides an operating system interface through which instrumented components (i.e., components of an instrumented platform) provide information and notification. WMI is a MICROSOFT™ implementation of the Distributed Management Task Force (DMTF) Web-Based Enterprise Management (WBEM) and the DMTF

Common Information Model (CIM) standards. WMI is preinstalled on some MICROSOFT™ WINDOWS® Operating Systems (OSs) and allows scripting languages like Visual Basic Scripting Edition (VBScript) or Windows PowerShell to manage computing devices and platforms running a MICROSOFT™ WINDOWS® OS. Such management can be performed locally (i.e., on the server or platform being managed) or remotely from a computing device external to the managed platform/device. 35:55-36:5

- Embodiments of the architecture 2500 can include packaging of a TaaS endpoint comprising an endpoint trust sensor with OS updates, such as, but not limited to, MICROSOFT™ WINDOWS®, UNIX, Linux, and Apple OSX updates (i.e., OS and application upgrades and security patches for browsers, business productivity suites, device drivers, and other platform components). By using the architecture 2500, an endpoint trust sensor can measure runtime operational integrity by evaluating risk based on 'actions' and leveraging the native machine instrumentation 610. 36:21-30
- The removable storage drive 2614 reads from and/or writes to a removable storage unit 2618 in a well-known manner. Removable storage unit 2618 may comprise a floppy disk, magnetic tape, optical disk, etc. which is read by and written to by removable storage drive 2614. As will be appreciated by persons skilled in the relevant art, removable storage unit 2618 includes a non-transitory computer usable storage medium having stored therein computer software and/or data. 37:43-50
- In alternative implementations, secondary memory 2610 may include other similar means for allowing computer programs or other instructions to be loaded into computer system 2600. Such means may include, for example, a removable storage unit 2622 and an interface 2620. Examples of such means may include a program cartridge and cartridge interface (such as that found in video game devices), a removable memory chip (such as an EPROM, or PROM) and associated socket, and other removable storage

units 2622 and interfaces 2620 which allow software and data to be transferred from the removable storage unit 2622 to computer system 2300. 37:51-61

- The computer system 2600 may also include a communications interface 2624. Communications interface 2624 allows software and data to be transferred between computer system 2600 and external devices. Communications interface 2624 may include a modem, a network interface (such as an Ethernet card), a communications port, a PCMCIA slot and card, or the like. Software and data transferred via communications interface 2624 may be in the form of signals, which may be electronic, electromagnetic, optical, or other signals capable of being received by communications interface 2624. These signals may be provided to communications interface 2324 via a communications path 2626. Communications path 2626 carries signals and may be implemented using wire or cable, fiber optics, a phone line, a cellular phone link, an RF link or other communications channels. 37:62-38:9
- Embodiments of the invention also may be directed to computer program products comprising software stored on any computer useable medium. Such software, when executed in one or more data processing device, causes a data processing device(s) to operate as described herein. Embodiments of the invention employ any computer useable or readable medium. Examples of computer useable mediums include, but are not limited to, primary storage devices (e.g., any type of random access memory), secondary storage devices (e.g., hard drives, floppy disks, CD ROMS, ZIP disks, tapes, magnetic storage devices, and optical storage devices, MEMS, nanotechnological storage device, etc.), and communication mediums (e.g., wired and wireless communications networks, local area networks, wide area networks, intranets, etc.). 38:45-59

7. The ’356 Patent AAPA

- Computer attacks are common today. Some examples of computer attacks are buffer

overflow attacks, malformed URL attacks, brute force attacks, viruses and worms. Most attacks are malicious in intent. Computer attacks are typically received via a network intranet or Internet interface targeted at the operating system or an installed service. While computer firewalls can prevent some types of malicious attacks they should not be considered a complete solution for stopping a malicious hacker from penetrating a computer on a network. ’356 Patent, Background.

- A computer virus is a computer program that is normally harmful in nature to a computer user. Computer viruses are received via several media, such as a computer diskette, e-mail or vulnerable program. Once a virus is received by a user, it remains “dormant” until it is executed by the user (or other program). The main difference of a virus versus a worm is the need for the user or program to execute the virus program for it to spread and infect others. *Id.*
- A computer worm is a computer program similar to a computer virus, except that a computer worm does not require action by a person to become active. A computer worm exploits some vulnerability in a system to gain access to that system. Once the worm has infected a particular system, it replicates by executing itself. Normally, worms execute themselves and spawn a process that searches for other computers on nearby networks. If a vulnerable computer is found, the worm infects this computer and the cycle continues. *Id.*
- Most computer attacks have a characteristic “signature” by which the attack can be identified. The signature can take various forms depending on the nature of the attack, but typically comprises several consecutive lines of plain text or executable code that are distinctive and appear in the attack. Once a signature is determined for a new computer attack, intrusion detection or intrusion prevention software can be created and distributed to customers. The intrusion detection or intrusion prevention software detects the attack

from a network interface card (NIC) or when the attack attempts to pass through a firewall. The detection is by a “key word” search for the signature of the attack. The intrusion prevention or intrusion detection software will then thwart the attack by deleting it or preventing its execution by appropriate command to the operating system. *Id.*

- It is important to identify new computer attacks (and their signatures), as soon as possible after the new attack is released. Then, its signatures can be identified and the intrusion prevention or intrusion detection software can be created and distributed to customers. *Id.*
- Likewise, it is important to detect a manual attempt to “hack” a victim's server or workstation, whereby a (hacker) person at a remote workstation attempts in real time to gain access to the victim's server or workstation. This typically begins by the hacker entering many combinations of userIDs and passwords, hoping that one such combination will gain access to sensitive software or data in the server or workstation. Hacking can also be facilitated if there is an improper configuration to a server which allows unknown third parties to gain administrative authority to a program or data base. After a hacking, there will usually be some residual evidence in log files or as binary executable code, as deleted or modified system files, etc. *Id.*
- A hacker may also transmit exploitation code to the victim's server or workstation, which code automatically exploits vulnerabilities in a victim's server, as would a hacker do manually. For example, a buffer overflow attack exploitation program exploits a vulnerability, typically caused by programmer error, that allows for arbitrary code execution on the target system. As another example, an attacker can inject special machine code into a program variable (usually input by a user) to cause arbitrary code execution in a program. This special code, once given to the program to execute, is placed in the correct area of computer memory, such that the executing program is unaware of the malicious intent of the injected code. There are several classes of buffer overflow, including format

string, remote and local. It is important to thwart hackers (as well as viruses and worms). *Id.*

- An Intrusion Detection System ("IDS") is currently known and has a known (i.e. "used") address to detect known computer attacks by matching key aspects of that attack to a known "signature". The IDS is associated with an enterprise, and has a list of known signatures of known viruses and worms, and other common attacks. The IDS searches each packet it receives for the known signatures, and thereby detects when the enterprise is being "attacked" by virus, worm or any other attack which has a known signature. When this occurs, the IDS notifies a security operations center ("SOC"), and the SOC will check that the proper anti-virus, anti-worm or other intrusion protection software is currently installed in the enterprise or customer network. While the IDS is effective in safeguarding an enterprise against known "exploits" (for example, computer viruses, worms and exploitation code), it does not identify or safeguard against new exploits for which the signatures are not yet known. *Id.*
- A "honeypot" is currently known to collect suspicious Internet message packets. The honeypot is a device such as a server, workstation or embedded device (for example, an old workstation, Single Board Computer (SBC) or de-commissioned server) that has an IP address on the Internet or company intranet, but the IP address is unused, i.e. the device has no function that requires input or service from any other server or workstation, the IP address is not registered with a domain name service, the IP address is not sent or broadcast to any other server or workstation, and the honeypot is not serving any useful function to the enterprise or network (other than gathering information). So, all packets sent to the honeypot are unsolicited and suspect. It is known for a human analyst to analyze all of the packets received by the honeypot to determine their type and whether they represent a known or unknown computer attack. For example, the analyst will determine which

packets are harmless broadcast traffic, network administration, or web crawler requests. The analyst will also look for harmful known viruses, worms, and exploitation code contained in the packets. The analyst will also look at residual evidence of hacking in the honeypot (for example, changes to data bases, software, system files, etc.). The analyst will also identify new computer attacks by filtering through network packets (logged by the honeypot) for known attacks. Once known attacks are filtered, the analyst has a smaller set of data to analyze. This smaller set of data is scrutinized for anything suggesting a new attack. Packets must have a purpose or be explained before they are discounted as known or harmless. While the foregoing human analysis of the honeypot process is effective, it is time consuming, requires a computer savvy human to make the analysis and is prone to error. Also, the shear number of packets received by the honeypot delays the detection of new computer attacks, viruses, computer worms and exploitation code. *Id.*

- FIG. 3 illustrates in more detail step 100 of FIG. 2 (i.e. determining if the packet contains a portion of a known exploit). In the illustrated embodiment of the present invention, a known intrusion detection system ("IDS") 22 is connected to the intranet 14 and to honeypot 12. The IDS 22 has a current list of signatures of known viruses, worms, exploitation programs and other exploits. The IDS performs a key word search of the packets it receives, searching for these known signatures. When the IDS detects such an exploit or a portion thereof in a packet it receives based on the presence of the key words in the packet, the IDS sends the packet in an alert or identifies the packet in an alert to program 30 in the honeypot 12. The identification can be in the form of a TCP sequence number of the current packet or the sequence of packets which includes the current packet. The TCP sequence is a sequence of packets that together form one packet. (TCP packets have a sequence number because the packets often arrive at their destination out of order. This is because the packets may take different routes to the destination. The TCP sequence

numbers are used to reassemble the packets at their destination.) (Other types of messages or packets can be identified in the alert by other means appropriate to the type of message or packet.) *Id.*, 6:33-55.

D. Prior Art Under 35 U.S.C. § 102(f)

Discovery is ongoing, and Defendant reserves the right to assert any of the other preceding items of prior art under 35 U.S.C. § 102(f), especially with respect to items of prior art where further investigation reveals that the purported inventors of the Patents-in-Suit derived the purported invention from the prior art and/or the author(s), maker(s), or inventor(s) of the prior art.

E. Prior Art Under 35 U.S.C. § 102(g)

The patents, publications, systems, and other prior art to the Patents-in-Suit identified in these contentions were conceived by another and diligently reduced to practice before the conception and reduction to practice of the Asserted Claims of the Patents-in-Suit by the named inventors of those patents, and Defendant alleges that such prior art by the individuals named as authors and inventors on the prior art, as well by the companies for which they worked, invalidate the Plaintiff's Patent-in-Suit under 35 U.S.C. § 102(g).

IV. PATENT L.R. 3-3(B)

Pursuant to P.R. 3-3, Defendant contends that certain prior art references below anticipate one or more Asserted Claims and that to the extent the identified prior art references do not anticipate the Asserted Claims, those claims are invalid as obvious under 35 U.S.C. § 103. Each anticipatory prior art reference, either alone or in combination with other prior art, also renders the Asserted Claims invalid as obvious. In particular, each anticipatory prior art reference may be combined with (1) information generally known to persons skilled in the art at the time of the alleged invention, and/or (2) any of the other anticipatory prior art references. To the extent that Plaintiff contends that any of the anticipatory prior art fails to disclose one or more limitations of

the Asserted Claims, Defendant contends that any difference between the reference and the corresponding patent claims would have been obvious to one of ordinary skill in the art. Thus, all statements regarding anticipation should be interpreted as both reflecting anticipation by the reference as well as invalidity due to single reference obviousness, to the extent that Plaintiff contends that any limitation is missing.

A. <u>Anticipation</u>

As stated above, Defendant incorporates by reference all other invalidity contentions related to the Asserted Patents served on or otherwise provided to Plaintiff, whether past or future. In accordance with P.R. 3-3(b) and (c), prior art references anticipating the Asserted Claims are provided below.

1. '796 Patent

Subject to Defendant's reservations of rights and in accordance with Patent L.R. 3-3(b) and 3-3(c), Defendant contends that each of the following prior art references anticipates one or more Asserted Claims of the '796 Patent in accordance with the claim charts attached hereto as Exhibits A.

- Friedl
- Budzinski
- Kanno
- Richardson
- Sheth
- Madnick
- Hu
- Nandhra

Additionally, Defendant incorporates by reference each of the grounds of invalidity presented in the petition for the following *inter partes* review proceeding(s) (IPRs) and any additional petitions for IPRs that may be filed challenging the patentability of the ’796 patent:

- IPR2023-00443

2. ’137 Patent

Subject to Defendant’s reservations of rights and in accordance with Patent L.R. 3-3(b) and 3-3(c), Defendant contend that each of the following prior art references anticipates one or more Asserted Claims of the ’137 Patent in accordance with the claim charts attached hereto as Exhibits B.

- Raduchel
- Van der Made
- Zwienenberg
- Edery
- Fischer
- ChakraVyuha and/or associated publications
- Dan
- Dresevic
- Agarwal
- Attanasio
- Lambert
- Angelo

Additionally, Defendant incorporates by reference each of the grounds of invalidity presented any petitions for IPRs that may be filed challenging the patentability of the ’137 patent:

3. ’441 Patent

Subject to Defendant’s reservations of rights and in accordance with Patent L.R. 3-3(b) and 3-3(c), Defendant contend that each of the following prior art references anticipates one or more Asserted Claims of the ’441 Patent in accordance with the claim charts attached hereto as Exhibits C.

- Starnes
- Lockhart
- Friedrichs
- Tuvell
- Greene
- Cha
- Guzman
- Leone
- Reissman
- Long Luo
- Matsumoto
- Bose
- OfficeScan 10.5 and/or associated publications
- Kuppusamy
- Kil
- Munetoh
- Rajan

Additionally, Defendant incorporates by reference each of the grounds of invalidity presented in the petition for the following *inter partes* review proceeding(s) (IPRs) and any additional petitions for IPRs that may be filed challenging the patentability of the ’441 patent:

- IPR2023-00801

4. ’517 Patent

Subject to Defendant’s reservations of rights and in accordance with Patent L.R. 3-3(b) and 3-3(c), Defendant contends that each of the following prior art references anticipates one or more Asserted Claims of the ’517 Patent in accordance with the claim charts attached hereto as Exhibits D.

- Dalcher
- VIDES and/or associated publications
- ThunderBYTE and/or associated publications
- Cuckoo Sandbox and/or associated publications
- Satish
- Bettini
- Nicodemus
- Yeh
- Chien
- Alperovitch
- OfficeScan 10.5 and/or associated publications

Additionally, Defendant incorporates by reference each of the grounds of invalidity presented in the petition for the following *inter partes* review proceeding(s) (IPRs) and any additional petitions for IPRs that may be filed challenging the patentability of the ’517 patent:

- IPR2023-00704
- IPR2023-00798

5. ’038 Patent & ’918 Patent

Subject to Defendant’s reservations of rights and in accordance with Patent L.R. 3-3(b) and 3-3(c), Defendant contends that each of the following prior art references anticipates one or more

Asserted Claims of the ’038 Patent in accordance with the claim charts attached hereto as Exhibits E. Subject to Defendant’s reservations of rights and in accordance with Patent L.R. 3-3(b) and 3-3(c), Defendant contends that each of the following prior art references anticipates one or more Asserted Claims of the ’918 Patent in accordance with the claim charts attached hereto as Exhibits I.

- Cheng
- Chickering
- Farmer
- Freund
- Herrmann I
- Herrmann II
- Sobel
- Wright
- Freund II
- Bluestone
- Corley
- Couillard
- Kuo
- CyberGatekeeper and/or associated publications, including each of Behr, Demaria, and InforExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity
- Elemental Security and/or associated publications, including Elemental Web
- Check Point R60 and/or Check Point’s NGX system, including its Management Server (SmartCenter), Virtual Private Network gateways, corresponding

SecuRemote/SecureClient remote access clients (SRSC or "RAVS") using the Secure Configuration Verification (SCV) feature as well as Integrity Client (also known as Integrity Agent and Integrity Flex) and Integrity Server (also known as Integrity Advanced Server or IAS) (together, the Integrity Advanced Server and Client system, or "IASC"), and associated publications (collectively, "Check Point R60")

- IPass and/or associated publications, including iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief

Additionally, Defendant incorporates by reference each of the grounds of invalidity presented in the petition for the following *inter partes* review proceeding(s) (IPRs) and any additional petitions for IPRs that may be filed challenging the patentability of the '038 and '918 patents, respectively:

- IPR2023-00573 ('038 patent)
- IPR2023-00574 ('918 patent)

6. '948 Patent

Subject to Defendant's reservations of rights and in accordance with Patent L.R. 3-3(b) and 3-3(c), Defendant contends that each of the following prior art references anticipates one or more Asserted Claims of the '948 Patent in accordance with the claim charts attached hereto as Exhibits F.

- Datta Ray
- Farley
- Gill
- Helman
- Kramer
- Lo

- Marvasti
- Njemanze
- Uner
- Wilson
- Bose
- Das
- Overcash II
- Sreenivas
- Leone
- Pan
- Williams
- OfficeScan 10.5 and/or associated publications

Additionally, Defendant incorporates by reference each of the grounds of invalidity presented in the petition for the following *inter partes* review proceeding(s) (IPRs) and any additional petitions for IPRs that may be filed challenging the patentability of the ’948 patent:

- IPR2023-00705
- IPR2023-00800

7. ’616 Patent

Subject to Defendant’s reservations of rights and in accordance with Patent L.R. 3-3(b) and 3-3(c), Defendant contends that each of the following prior art references anticipates one or more Asserted Claims of the ’616 Patent in accordance with the claim charts attached hereto as Exhibits G.

- Cooper
- Datta Ray
- DiFalco

- Dowd
- Ghai
- Jaeger
- Kelekar
- Overcash
- Overcash II
- Patrick
- Thielamay
- Wootton
- Das
- Sreenivas
- Leone Schrecker
- Singla
- OfficeScan 10.5 and/or associated publications

Additionally, Defendant incorporates by reference each of the grounds of invalidity presented in the petition for the following *inter partes* review proceeding(s) (IPRs) and any additional petitions for IPRs that may be filed challenging the patentability of the ’616 patent:

- IPR2023-00706
- IPR2023-00799

8. ’356 Patent

Subject to Defendant’s reservations of rights and in accordance with Patent L.R. 3-3(b) and 3-3(c), Defendant contends that each of the following prior art references anticipates one or more Asserted Claims of the ’356 Patent in accordance with the claim charts attached hereto as Exhibits P.

- Afek

- Apap
- Carpenter
- Cooper
- Copeland
- Desai
- Eichstaedt
- Fretwell
- Gassen
- Gluck
- Gupta
- Jerrim
- Mualem
- Roux
- Stute
- Triulzi
- Wang

Additionally, Defendant incorporates by reference each of the grounds of invalidity presented in any additional petitions for IPRs that may be filed challenging the patentability of the ’356 patent:

B. <u>Obviousness</u>

As stated above, Defendant incorporates by reference all other invalidity contentions related to the Asserted Patents served on or otherwise provided by Plaintiff, whether past or in the future. In addition to certain claims being anticipated under Plaintiff’s apparent claim constructions as indicated above, the Asserted Claims are also invalid as obvious over the same teachings identified for anticipation. Further, the Asserted Claims are obvious over various combinations of the references shown in the claim charts accompanying or incorporated by reference into this

disclosure. No Asserted Claim goes beyond combining familiar elements in known methods to achieve predictable results or does more than choose between clear alternatives known to those of skill in the art. Thus, to the extent that an Asserted Claim is not anticipated, it is nevertheless invalid as obvious. Specifically, Defendant asserts that any charted or incorporated reference in combination with one or more other charted or incorporated references renders the Asserted Claims obvious.

1. Motivation to Combine

Motivations to combine, as well as the general state of the art, may be found in a variety of places including in the references defined above, and the specification of the Asserted Patents. A person of ordinary skill in the art at the time of the alleged invention would have been motivated to combine any one piece of identified prior art with any other identified piece of prior art. For at least these reasons, it would have been obvious to a person of skill in the art at the time of the alleged invention of the Asserted Claims to combine the various references cited herein so as to practice the Asserted Claims and there is a motivation in the art to make such a combination.

Motivations to combine various prior art references are present in the references themselves, the common knowledge of one of ordinary skill in the art, the prior art as a whole, or the nature of the problems allegedly addressed by the Asserted Patents. Further reasons to combine the references identified in these charts include common sense, the nature of the problem being solved, the express, implied and inherent teachings of the prior art, the knowledge of persons of ordinary skill in the art, the fact that the prior art is generally directed towards methods and systems in the same field, that such combinations would have yielded predictable results, and that such combinations would have represented known alternatives to a person of ordinary skill in the art.

In *KSR International Co. v. Teleflex, Inc.*, the United States Supreme Court held that, among other things, "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." 127 S. Ct. 1727, 1739 (2007);

see also id. at 1731 ("[A] court must ask whether the improvement is more than the predictable use of prior art elements according to their established functions."). In particular, a patent is obvious where "the content of the prior art, the scope of the patent claim, and the level of ordinary skill are not in material dispute, and the obviousness of the claim is apparent in light of these factors." *Id.* at 1745–46. The Supreme Court found that "if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill." *Id.* at 1731.

Moreover, the Court recognized that market pressures will motivate a person of ordinary skill to survey known art for solutions to problems. *Id*. at 1732 ("When there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill in the art has good reason to pursue the known options within his or her technical grasp."). When a person of ordinary skill uses an identified, predictable solution to solve a problem, "it is likely the product not of innovation but of ordinary skill and common sense." *Id*.

In addition, when a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. *Id*. at 1740. If a person of ordinary skill can implement a predictable variation, U.S.C. § 103 bars its patentability. *Id*. The rationale to combine or modify prior art references is significantly stronger when references seek to solve similar problems, come from the same field, and correspond well. *In re Inland Steel Co.*, 265 F.3d 1354, 1362 (Fed. Cir. 2001).

Although the law does not require evidence of motivation to combine, motivation exists to combine one or more of the references disclosed herein with each other. In addition exemplary motivations identified herein, motivation to modify a particular reference or to combine any two or more of the identified references comes from (a) the nature of the problem being solved, (b) the

teachings of the prior art, (c) the knowledge of persons of ordinary skill in the art, (d) the fact that all of the references teach systems, apparatuses, and methods related to the subject matter and address the same technical issues described in the Asserted Patents, and (e) one would be motivated by considerations of efficiency, effectiveness, convenience, cost-savings, and accessibility, to combine the various teachings. Additionally, one would be motivated to address the alleged problems or achieve the purported objectives identified in the Background sections of the Asserted Patents.

To the extent not anticipated, the Asserted Claims represent no more than the result of ordinary implementation of the prior art. Moreover, no showing of a specific motivation to combine prior art is required to combine the references disclosed above and in the attached charts, as each combination of art would have no unexpected results, and at most would simply represent an alternative to one of skill in the art. *See KSR*, 127 S.Ct. at 1739-40 (rejecting the Federal Circuit's "rigid" application of the teaching, suggestion, or motivation to combine test, instead espousing an "expansive and flexible" approach). Indeed, the Supreme Court held that a person of ordinary skill in the art is "a person of ordinary creativity, not an automaton" and "in many cases a person of ordinary skill in the art will be able to fit the teachings of multiple patents together like pieces of a puzzle." *Id.* at 1742. Nevertheless, in keeping with the Local Patent Rules, and in addition to the information contained in the section immediately above and elsewhere in these contentions, additional motivation and reason to combine the cited art are identified. A person having ordinary skill in any or all of these fields would be aware of all prior art in those fields, including but not limited to the identified prior art references and systems, and would have been motivated to combine the teachings of prior art with the field.

In sum, motivations to modify or combine the identified references including the references listed above can be found via, for example, discussions in the cited references, the state of the art discussed in the references, and the knowledge of one of ordinary skill in the art. One of ordinary

skill in the art would have been motivated to combine these references, because these references relate to common objectives and subject matter. The references share commonalities in terms of their general subject matter as well as the types of equipment, products, systems, modules, software, and/or methods used. Further, the prior art references explicitly or implicitly reference other prior art references, share common authors or inventor, were published in the same journals, were compiled by a common author of a compilation or reference book, were presented at the same conferences, and/or were developed at common companies, schools, or organizations which would motivate one of skill in the art to combine them. These references are within the field of the Asserted Patents and are directed to similar subject matter within the field. Additionally, the references, and any products, devices, or processes described in the references, existed and/or were invented in the same time period providing further motivation for combination. These disclosures were provided without prejudice to any arguments or objections concerning the relevance of motivation to combine in connection with any invalidity contentions.

Defendant reserved the right to further specify the motivations to combine the prior art in response to positions that Plaintiff may take later in this case and as discovery, including third party discovery, proceeds. Defendant may rely on any and all portions of the prior art, other documents, and expert testimony to establish that a person of ordinary skill in the art would have been motivated to modify or combine the prior art so as to render the claims invalid as obvious.

While they reserve the right to rely on any combination of the references reflected in their charts or incorporated herein by reference, Defendant provided the following exemplary and non-exhaustive list of references and/or combinations evidencing invalidity of the claims of the Asserted Patents.

For at least the reasons described in these contentions, it would have been obvious to one of ordinary skill in the art to combine any of a number of prior art references, including any combination of those prior art references identified above along with the knowledge of one of

ordinary skill in the art to meet the limitations of the Asserted Claims. Moreover, as mentioned above, Defendant has not yet completed their search or discovery concerning additional prior art. As such, Defendant's inclusion of exemplary combinations does not preclude them from identifying other invalidating combinations as appropriate, and Defendant reserves the right to identify additional specific combinations as well as to detail and explain such combinations.

To the extent not anticipated, the Asserted Claims represent no more than the result of ordinary variations of the prior art. Defendant further believes that no showing of a specific motivation to combine prior art is required to combine the references disclosed above and in the attached charts, as each combination of art would have no unexpected results, and at most would simply represent a known alternative to one of skill in the art. *See KSR Int'l Co. v. Teleflex, Inc.*, 550 U.S. 398, 415-16 (2007) (rejecting the Federal Circuit's "rigid" application of the teaching, suggestion, or motivation to combine test, instead espousing an "expansive and flexible" approach). Indeed, the Supreme Court held that a person of ordinary skill in the art is "a person of ordinary creativity, not an automaton" and "in many cases a person of ordinary skill in the art will be able to fit the teachings of multiple patents together like pieces of a puzzle." *Id.* at 420-21. Nevertheless, in addition to the information contained elsewhere in these contentions, Defendant identifies motivations and reasons to combine the cited art.

One or more combinations of the prior art references identified above would have been obvious because these references would have been combined using: known methods to yield predictable results; known techniques in the same way; a simple substitution of one known, equivalent element for another to obtain predictable results; and/or a teaching, suggestion, or motivation in the prior art generally. In addition, it would have been obvious to try combining the prior art references identified above because there were only a finite number of predictable solutions and/or because known work in one field of endeavor prompted variations based on predictable design incentives and/or market forces either in the same field or a different one.

Further, the combinations of the prior art references identified above would have been obvious because the combinations represent known potential options with a reasonable expectation of success.

Additional evidence that there would have been a motivation to combine the prior art references identified above includes the interrelated teachings of multiple prior art references; common authorship; the effects of demands known to the design community or present in the marketplace; the existence of a known problem for which there was an obvious solution encompassed by the Asserted Claims; the existence of a known need or problem in the field of the endeavor at the time of the alleged invention(s); and the background knowledge that would have been possessed by a person having ordinary skill in the art.

Thus, the motivation to combine the teachings of the prior art references disclosed above are found in the references themselves and also in: (1) the nature of the problem being solved; (2) the express, implied and inherent teachings of the prior art; (3) the knowledge of persons of ordinary skill in the art; (4) the predictable results obtained in combining the different elements of the prior art; (5) the predictable results obtained in simple substitution of one known element for another; (6) the use of a known technique to improve similar devices, methods, or products in the same way; (7) the predictable results obtained in applying a known technique to a known device, method, or product ready for improvement; (8) the finite number of identified predictable solutions that had a reasonable expectation of success; and (9) known work in various technological fields that could be applied to the same or different technological fields based on design incentives or other market forces.

Additionally, it would be obvious to one of skill in the art to consult and/or combine any of the prior art listed above because all of these references relate to the same area of technology and/or are from analogous art. The Asserted Claims only unite old elements, well known in the field, with no change in their respective function. Given the interrelated teachings of the prior art,

the effects of demands known to the design community or present in the marketplace, and the background knowledge possessed by a person having ordinary skill in the art, it would have been obvious for one of ordinary skill in the art to combine these familiar elements, disclosed and/or embodied in the prior art listed above to practice the Asserted Claims.

In accordance with these advances, the prior art could have been combined according to methods known to those of ordinary skill within the fields of information extraction and network security to yield predictable results. The substitution of one element in these fields could have been predictably achieved by one of ordinary skill at the time of the alleged inventions. One of ordinary skill in the art would have been aware of these various methods and apparatuses, and would have been able to select appropriate attributes of one for inclusion in another. Those of ordinary skill in the art could have employed known techniques to improve similar prior art devices in the same way as claimed in the Asserted Patents.

In addition, the prior art also provided sets of finite, identified, predictable solutions for known problems that would have been obvious to those of ordinary skill to try with a reasonable expectation of success. In addition, the prior art provided multiple teachings, suggestions, and motivations that would have led one of ordinary skill to modify or to combine prior art teachings to arrive at systems and methods purportedly covered by the asserted claims, as can be found in the references themselves.

The prior art references provide motivations to combine because they describe the field of the Asserted Patents, teach improvements, explain desired features, and even expressly state that one of skill in the art would be able to apply their teachings to related systems or methods.

Any reference or combination of references that anticipates or makes obvious an asserted independent claim also makes obvious any asserted claim dependent on that independent claim, as the element of each dependent Asserted Claim was known by a person of ordinary skill at the time of the alleged invention, and it would have been obvious to combine those known elements with

the independent claims at least as a matter of common sense and routine innovation. Accordingly, Defendant contends that each Asserted Claim would have been obvious not only by the combinations described in these contentions, but also by any combination of references that renders obvious an Asserted Claim.

In addition to the examples of motivation provided below and/or that are set out in the Exhibits A-P, Defendant reserves the right to rely on the disclosures of the references listed above for additional motivation to combine. The below-identified examples of combinations are given merely to illustrate various motivations to combine and are not intended to provide an exhaustive list of every possible combination to which the motivation may apply. Defendant reserves the right to contend that the above and below described motivations to combine apply to other combinations at the appropriate time, i.e., in expert reports regarding invalidity.

C. <u>Obviousness Combinations and Exemplary Motivations to Combine</u>

1. '796 Patent

To the extent that Plaintiff contends that any of the references identified as anticipatory references fail to disclose one or more elements, those anticipatory references render the Asserted Claims obvious, either alone or in combination with other references. Defendant contends that one of skill in the art, at the time the alleged inventions were made, would have been motivated to combine the references disclosed herein in such a way as to reach the alleged inventions, as described in further detail below. Defendant presently believes a reasonable basis exists that each of the claims asserted against it would have been obvious to one of ordinary skill in the art at the time of the alleged invention.

Below, Defendant identifies their current bases for obviousness. As seen below, Defendant presently asserts certain combinations of prior art reference(s) with or without the knowledge of one of ordinary skill i. Defendant offers these combinations without conceding that the reference(s) standing alone are insufficient to invalidate the relevant claims. Defendant specifically reserves their rights to combine any of its anticipatory references or any combinations

of references with the knowledge of ordinary skill in the art to rebut a contention of Plaintiff that such reference(s) standing alone do not disclose every element of the Asserted Claims.

Defendant lists below prior art references and combinations of prior art references now known to Defendant, which Defendant contends render at least obvious the ’796 Asserted Claims. The corresponding charts in Exhibit A for each prior art reference identifies where and how each alleged item of prior art meets one or more limitations of the Asserted Claims. Pursuant to Patent Local Rule 3-6, Defendant expressly reserves the right to supplement these charts to provide additional details about obviousness should the Court’s claim construction require, or should Plaintiff serve adequate Infringement Contentions that better serve to notify Defendant which elements of its products correspond with the claim elements. Defendant’s contention that the references in this section in various combinations render obvious the ’796 Asserted Claims under 35 U.S.C. § 103 is in no way an admission or suggestion that each reference does not independently anticipate or render obvious the ’796 Asserted Claims under 35 U.S.C. § 102. These combinations would have been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field.

It would been obvious to combine each of the references discussed below because they are analogous art generally directed to methods, apparatuses, and techniques for providing automatic information extraction from data streams such as text and spoken documents. The technology described in the ’796 Patent was widely known before the alleged priority date of the ’796 Patent and provided numerous motivations to combine the prior art references and systems disclosed herein. Methods of extracting information from data streams, were well known in the art, as admitted in the ’796 AAPA, (*see* Section III.A above and incorporated by reference herein) for example speech recognition techniques, other known transcription techniques and character

recognition techniques. Further, programming languages for identifying regular expressions were well known the in the art as admitted in the '796 AAPA.

> "In accordance with an illustrative embodiment of the invention, the term "regular expression" is taken to mean any form of pattern that is matchable in the flex, lex, or perl programming languages, although the invention is not limited thereto. The programming languages flex and lex are described in J. R. Levine *et al.*, "Lex & Yacc," 1992, the disclosure of which is incorporated by reference herein. The programming language perl is described in L. Wall *et al.*, "Programming Perl," 1996, the disclosure of which is incorporated by reference herein. Similarly, in accordance with such an illustrative embodiment, the term "matching" refers to the processes by which these programs match their input."

'796 Patent, 3:17-34.

The '796 Patent combines these components according to their known, predictable and intended uses, just as they were and/or would have been combined by a POSA in the prior art. None of the combinations presents an unexpected result, a counterintuitive combination, or a technological challenge; and no such description exists in the common specification of the asserted patents.

Accordingly, the prior art cited herein would have been readily combined according to methods known to those of ordinary skill within the field of automatic information extraction from data streams to yield predictable results. The substitution of one element in this system could have been predictably achieved by one of ordinary skill at the time of the alleged invention. One of ordinary skill in the art would have been able to select appropriate attributes of one for inclusion in another. Those of ordinary skill in the art could have employed known techniques to improve similar prior art devices in the same way as claimed in the '796 Patent. Thus, the combinations all have well defined, predictable, expectation of success.

- For the Asserted Claims of the '796 patent, to the extent the references charted in any of the charts in Exhibit A do not expressly disclose "identifying at least one regularly identifiable expression in the input sequence of data symbols, wherein the at least one regularly identifiable expression represents a pattern that is matchable in accordance with

a programming language that supports such a regularly identifiable expression", or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the AAPA and with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any other charted references and/or of those references identified in Section III.A, including, for example: Friedl, Sheth, Richardson, Madnick, Nandhra, Yun, Hu. For example, Friedl discloses identifying certain information (such as repeated (double) words in edited documents, lines in mailbox files beginning with "Subject:" or "From:", etc.) within input text (i.e., input sequences of data symbols), and that such information represents a pattern that is matchable in accordance with a programming language that supports the use of regularly identifiable expressions (e.g., by disclosing the user of Perl and Python for processing text and identifying and extracting information, wherein both languages (and others) have such regular expression support built-in). A person of skill in the art would be motivated to combine these references because they each disclose methods, apparatuses, and techniques for providing automatic information extraction from data streams such as text and spoken documents. It is the nature of those of skill in the art of automatic information extraction from data streams to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems automatically extracted information from data streams and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how, and when, other solutions in this field identified regularly identifiable expressions though the use of a programming language that supports regularly identifiable expressions, particularly because such programming languages are commonly used and provide the basis for many different solutions for identifying and extracting information from data streams. These combinations would have also been obvious to try given that there were a

finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’796 patent, to the extent the references charted in any of the charts in Exhibit A do not expressly disclose “identifying at least a portion of information associated with the at least one regularly identifiable expression”, or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the AAPA and with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any other charted references and/or of those references identified in Section III.A, including, for example: Friedl, Sheth, Richardson, Madnick, Nandhra, Yun, Hu. For example, Friedl discloses identifying portions of the information within input text based on the use of regular expression pattern matching (i.e., identifying at least a portion of information associated with the at least one regularly identifiable expression), such as by identifying a double (repeated) word within a received text, or certain information which is associated with a regularly identifiable expression (e.g., text) in a mailbox file, such as the subjects, dates, or reply addresses that follow “Subject:”, “Date:”, and “Reply-To” prefixes, respectively, via the use of regular expressions. A person of skill in the art would be motivated to combine these references because they each disclose methods, apparatuses, and techniques for providing automatic information extraction from data streams such as text and spoken documents. It is the nature of those of skill in the art of automatic information extraction from data streams to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products

and systems automatically extracted information from data streams and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how, and when, other solutions in this field identified regularly identifiable expressions and information associated with such expressions. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’796 patent, to the extent the references charted in any of the charts in Exhibit A do not expressly disclose “extracting the portion of information”, or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the AAPA and with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any other charted references and/or of those references identified in Section III.A, including, for example: Friedl, Sheth, Richardson, Madnick, Nandhra, Yun, Hu. For example, Friedl discloses extracting information associated with a regularly identifiable expression, such as the information following the prefixes “Subject:”, “Date:”, and “Reply-To” and saving that extracted information in a local variable ($subject, $date, $reply_address). A person of skill in the art would be motivated to combine these references because they each disclose methods, apparatuses, and techniques for providing automatic information extraction from data streams such as text and spoken documents. It is the nature of those of skill in the art of automatic information extraction from data streams to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems automatically

extracted information from data streams and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how, and when, other solutions in this field identified regularly identifiable expressions, and how others extracted identified information or information associated with identified regularly identifiable expressions. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’796 patent, to the extent the references charted in any of the charts in Exhibit A do not expressly disclose “wherein the regularly identifiable expression identifying step comprises comparing the input sequence of data symbols to one or more previously-stored regularly identifiable expressions to determine if there is a match between a portion of the input sequence and at least one of the previously-stored regularly identifiable expressions”, or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the AAPA and with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any other charted references and/or of those references identified in Section III.A, including, for example: Friedl, Sheth, Richardson, Madnick, Nandhra, Yun, Hu . For example, Friedl discloses comparing input text (i.e., the input sequence of data symbols) to previously-stored information (i.e., regularly identifiable expressions) to determine if there is a match between a portion of the input text (i.e., the input sequence) and the previously-stored information (i.e., regularly identifiable expressions). A person of skill in the art would be motivated to combine these references because they each disclose

methods, apparatuses, and techniques for providing automatic information extraction from data streams such as text and spoken documents. It is the nature of those of skill in the art of automatic information extraction from data streams to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems automatically extracted information from data streams and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how, and when, other solutions in this field identified regularly identifiable expressions, and what use cases others found for the technology (i.e., how others used pattern matching to compare values to previously stored values). These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’796 patent, to the extent the references charted in any of the charts in Exhibit A do not expressly disclose “wherein the extracted portion of information is used to take a specified action”, or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the AAPA and with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any other charted references and/or of those references identified in Section III.A, including, for example: Friedl, Sheth, Richardson, Madnick, Nandhra, Yun, Hu. For example, Friedl discloses taking certain actions based on extracted text (i.e., the extracted portion of information), such as by performing computations on extracted information (e.g., determining temperature format (is it Celsius?), converting Fahrenheit

to Celsius, or displaying the result of the extraction and/or subsequent calculation. A person of skill in the art would be motivated to combine these references because they each disclose methods, apparatuses, and techniques for providing automatic information extraction from data streams such as text and spoken documents. It is the nature of those of skill in the art of automatic information extraction from data streams to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems automatically extracted information from data streams and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how, and when, other solutions in this field identified regularly identifiable expressions, and what use cases others found for the technology (i.e., what actions others took based on the extracted information). though the use of a programming language that supports regularly identifiable expressions, particularly because such programming languages are commonly used and provide the basis for many different solutions for identifying and extracting information from data streams. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’796 patent, to the extent the references charted in any of the charts in Exhibit A do not expressly disclose “a data capture device for obtaining the input sequence of data symbols”, or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the AAPA and with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the

teachings of any other charted references and/or of those references identified in Section III.A, including, for example: Friedl, Sheth, Richardson, Madnick, Nandhra, Yun, Hu, and Groner. For example, Friedl discloses computer which captures from the user input text (i.e., an input sequence of data symbols), via a keyboard or similar (i.e., a data capture device). A person of skill in the art would be motivated to combine these references because they each disclose methods, apparatuses, and techniques for providing automatic information extraction from data streams such as text and spoken documents. It is the nature of those of skill in the art of automatic information extraction from data streams to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems automatically extracted information from data streams and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how, and when, other solutions in this field identified classes of data in an input data stream via a data capture device of some kind like a keyboard. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’796 patent, to the extent the references charted in any of the charts in Exhibit A do not expressly disclose “a data output device, operatively coupled to the at least one processor, for presenting the extracted portion of information to a user”, or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the AAPA and with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any other charted

references and/or of those references identified in Section III.A, including, for example: Friedl, Sheth, Richardson, Madnick, Nandhra, Yun, Hu, and Groner. For example, Friedl discloses a computer which displays identified and extracted text (i.e., presents the extracted portion of information) to a user via a display or similar (i.e., a data output device), and that Python and/or Perl operations performed by the processor to identify and extract text, as well as display such text, indicating the display (i.e., data output device) is operatively coupled to the processor. A person of skill in the art would be motivated to combine these references because they each disclose methods, apparatuses, and techniques for providing automatic information extraction from data streams such as text and spoken documents. It is the nature of those of skill in the art of automatic information extraction from data streams to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems automatically extracted information from data streams and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how, and when, other solutions in this field identified classes of data in an input data stream and how others output extracted information to a user (i.e., via a data output device like a display). These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’796 patent, to the extent the references charted in any of the charts in Exhibit A do not expressly disclose fundamental computer components such as, for example, “at least one processor” and/or “memory, operatively coupled to the at

least one processor", or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the AAPA and with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any other charted references and/or of those references identified in Section III.A, including, for example: Friedl, Sheth, Richardson, Madnick, Nandhra, Yun, Hu, and Groner. For example, Friedl discloses the use of a CPU (i.e., a processor) to perform regular expression matching, and the use of Perl and Python, which require the use of a processor to execute code written in such languages, and further discloses the use of memory (RAM or other memory, such as persistent memory) operatively coupled to the processor for storing the results of the identifying and extracting operations. A person of skill in the art would be motivated to combine these references because they each disclose methods, apparatuses, and techniques for providing automatic information extraction from data streams such as text and spoken documents. That computers, and other devices including computers, use processors and memory was well-known to a POSA in July 2001, and the '796 Patent does not suggest that the claimed "processor" or "memory" is structurally different from any conventional "processor" or "memory" used in a conventional computer. A POSA would have been aware of numerous prior art devices that included a processor and a memory, including memory that was utilized to store information, and processors that were utilized to execute programming instructions, to perform automatic information extraction from data streams. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

Consistent with the exemplary motivations to combine each of the references discussed above, the below table provides a summary of the exemplary combinations of references rendering obvious to a POSA, with or without a POSA's knowledge (including the AAPA and general state of the art), skill and//or creativity, the Asserted Claims of the '796 Patent.

Reference	**Reference Alone or in Combination With One or More of the Following**
Budzinkski	Kanno, Friedl, Sheth, Richardson, Madnick, Nandhra, Hu, Yun, Groner
Kanno	Budzinkski, Friedl, Sheth, Richardson, Madnick, Nandhra, Hu, Yun, Groner
Friedl	Budzinkski, Kanno, Sheth, Richardson, Madnick, Nandhra, Hu, Yun, Groner
Sheth	Budzinkski, Kanno, Friedl, Richardson, Madnick, Nandhra, Hu, Yun, Groner
Richardson	Budzinkski, Kanno, Friedl, Sheth, Madnick, Nandhra, Hu, Yun, Groner
Madnick	Budzinkski, Kanno, Friedl, Sheth, Richardson, Nandhra, Hu, Yun, Groner
Nandhra	Budzinkski, Kanno, Friedl, Sheth, Richardson, Madnick, Hu, Yun, Groner
Hu	Budzinkski, Kanno, Friedl, Sheth, Richardson, Madnick, Nandhra, Yun, Groner
Yun	Budzinkski, Kanno, Friedl, Sheth, Richardson, Madnick, Nandhra, Hu, Groner

Additionally, Defendant incorporates by reference each of the obviousness combinations and motivations to combine presented in the petition for the following *inter partes* review proceeding(s) (IPRs) and any additional petitions for IPRs that may be filed challenging the patentability of the '796 patent:

- IPR2023-00443

2. '137 Patent

To the extent that Plaintiff contends that any of the references identified as anticipatory references fail to disclose one or more elements, those anticipatory references render the Asserted Claims obvious, either alone or in combination with other references. Defendant contends that one of skill in the art, at the time the alleged inventions were made, would have been motivated to

combine the references disclosed herein in such a way as to reach the alleged inventions, as described in further detail below. Defendant presently believes a reasonable basis exists that each of the claims asserted against it would have been obvious to one of ordinary skill in the art at the time of the alleged invention.

Below, Defendant identifies their current bases for obviousness. As seen below, Defendant presently asserts certain combinations of prior art reference(s) with or without the knowledge of one of ordinary skill in the art. Defendant offers these combinations without conceding that the reference(s) standing alone are insufficient to invalidate the relevant claims. Defendant specifically reserves their rights to combine any of its anticipatory references or any combinations of references with the knowledge of ordinary skill in the art to rebut a contention of Plaintiff that such reference(s) standing alone do not disclose every element of the Asserted Claims.

Defendant lists below prior art references and combinations of prior art references now known to Defendant, which Defendant contends render at least obvious the '137 Asserted Claims. The corresponding charts in Exhibit B for each prior art reference identifies where and how each alleged item of prior art meets one or more limitations of the Asserted Claims. Pursuant to Patent Local Rule 3-6, Defendant expressly reserves the right to supplement these charts to provide additional details about obviousness should the Court's claim construction require, or should Plaintiff serve adequate Infringement Contentions that better serve to notify Defendant which elements of their products allegedly correspond with the claim elements. Defendant's contention that the references in this section in various combinations render obvious the '137 Asserted Claims under 35 U.S.C. § 103 is in no way an admission or suggestion that each reference does not independently anticipate or render obvious the '137 Asserted Claims under 35 U.S.C. § 102. These combinations would have been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field.

It would been obvious to combine each of the references discussed below because they are analogous art generally directed to managed security control of processes on a computer system, application validation, and/or application monitoring. The technology described in the '137 Patent was widely known before the alleged priority date of the '137 Patent and provided numerous motivations to combine the prior art references and systems disclosed herein. Methods, and defense-in-depth systems, for managed security control of processes on a computer system, including providing kernel-level protection of a computer system from rogue or malicious computer programs, were well known in the art, as admitted in the '137 AAPA, (*see* Section III.B above and incorporated by reference herein). For example, as admitted in the '137 AAPA, "many commercially available products that detect and exclude viruses from computing devices and systems" were accessible to those skilled in the art. Additionally, for example, and as admitted in the '137 AAPA, various methods and systems existed for malware detection and exclusion, including those involving (1) "virtual execution of a computer code to attempt to identify harmful code before it is actually allowed to execute" and (2) "controlling and monitoring a computing device in real time while it is actually running a program, and attempting to anticipate any harmful activity the program may try to initiate." Further, the '137 AAPA acknowledges that the following characteristics and determinations were well known with prior art, real-time malware detection and exclusion techniques:

> "By intercepting process loading and unloading and keeping a list of currently-active processes, each client process can be checked for various characteristics, including checking executable names, version numbers, executable file checksums, version header details, configuration settings, and the like. With this information, a determination can be made whether a particular process in question should have access to the Internet and what kind of access (i.e., protocols, Internet addresses, time limitations, and the like) is permissible for the given specific user."

'137 Patent, 2:49-58.

The '137 Patent combines these components according to their known, predictable and intended uses, just as they were and/or would have been combined by a POSA in the prior art. None of the combinations presents an unexpected result, a counterintuitive combination, or a

technological challenge; and no such description exists in the common specification of the asserted patents.

Accordingly, the prior art cited herein would have been readily combined according to methods known to those of ordinary skill within the field of managed security control of processes on a computer system to yield predictable results. The substitution of one element in this system could have been predictably achieved by one of ordinary skill at the time of the alleged invention. One of ordinary skill in the art would have been able to select appropriate attributes of one for inclusion in another. Those of ordinary skill in the art could have employed known techniques to improve similar prior art devices in the same way as claimed in the ’137 Patent. Thus, the combinations all have well defined, predictable, expectation of success.

- For the Asserted Claims of the ’137 patent, to the extent the references charted in any of the charts in Exhibit B do not expressly disclose “receiving notice from the computing device’s operating system”, “intercepting a signal from the computing device’s operating system”, or similar claim limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Van der Made, Raduchel, Edery, Fischer, ChakraVyuha, Dan, Angelo, Dresevic, Agarwal, Attanasio, Lambert, Zwienenberg. A person of skill in the art would be motivated to combine these references because they each disclose control and monitoring of processes/programs on a computing device in the same general field. It is the nature of those of skill in the art of managed security control of processes on a computer system to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how and from where other products and systems receive notice and intercept signals within a computing

device and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how and from where other solutions in this field receive notice and intercept signals within a computing device. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results. A person of ordinary skill would have also been motivated because a person of ordinary skill would have known that programs are loaded and executed via operating system functions, that those functions can be intercepted and interrupted, and that when validating a program to determine whether it is trustworthy or potentially malicious, it would be preferable to halt the program from executing so that validation can occur before the program has an opportunity to load, execute, and potentially perform malicious activities.

- For the Asserted Claims of the ’137 patent, to the extent the references charted in any of the charts in Exhibit B do not expressly disclose “monitoring, at the operating system kernel of the computing device”, “wherein the step of … is performed at the operating system kernel of the computing device”, or similar claim limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, ChakraVyuha, Dan, Agarwal, Lambert, Dresevic, Angelo. A person of skill in the art would be motivated to combine these references because they each disclose control and monitoring of processes/programs on a computing device in the same general field. It is the nature of those of skill in the art of managed

security control of processes on a computer system to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how and from where other products and systems receive notice, and monitor programs/processes, within a computing device and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how and from where other products and systems receive notice, and monitor programs/processes, within a computing device. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results. A person of ordinary skill would also been motivated to monitor at the operating system kernel of the computing device because it is a space where various operating system functions that enable opening, loading, and executing files, creating processes, installations, and the like occur that may be associated with malicious activity. A person of ordinary skill would also have been motivated because it was well known that memory is generally divided into kernel space and user space, such that there are a finite number of options of which space to monitor. Furthermore, to the extent it is contended that the prior discloses interrupting the loading and execution of a program and validating it, but not monitoring the program, it would have been obvious to subsequently monitor the program and a person of ordinary skill would have been motivated to do so to determine if the program exhibited suspicious activity. Likewise, to the extent it is contended that the prior art discloses monitoring the program but not interrupting the loading and execution of the program and validating it, a

person of ordinary skill would have been motivated to ascertain the trustworthiness of the program.

- For the Asserted Claims of the ’137 patent, to the extent the references charted in any of the charts in Exhibit B do not expressly disclose “validating the new program comprises comparing a checksum for the new program with a previously determined checksum” or “computing a checksum for the new program, and comparing the new program’s checksum to a checksum for the allowed program”, or similar claim limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, V Van der Made, Raduchel, Edery, Fischer, ChakraVyuha, Dan, Angelo, Dresevic, Agarwal, Attanasio, Lambert, Zwienenberg. A person of skill in the art would be motivated to combine these references because they each disclose control and monitoring of processes/programs on a computing device in the same general field. It is the nature of those of skill in the art of managed security control of processes on a computer system to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems validate new programs or processes and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how other solutions in this field validate new programs or processes. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable

results. A person of ordinary skill in the art also would have been motivated because it was well known that trusted programs can be validated with a hash or checksum value of the program for comparison with a known valid hash or checksum value of a desired program. *E.g.*, Lambert at Abstract, 11:40-53, Ex. B-Lambert. The '137 Patent admits that the validation module was prior art: "The validation module 108 can be any one of a variety of pieces of software that are used to verify that a program has not been tampered with. For example, the validation module can represent the MD5 software module commonly known to those in the art. The MD5 software module calculates a checksum for each allowed program and that checksum is stored for later comparison. '137 Patent, 8:7-19. It was also known in the art to allow or restrict execution of a program based on file system or network path, its URL zone, the sender of an attachment, identification of information provided by a trusted application, file extension, and file name. *E.g.*, Lambert at 11:40-53, 13:4-14:29. Furthermore, by validating a program, resources can be saved that would otherwise be devoted to monitoring the program's execution for suspicious activity, and resources can be devoted to monitoring programs that cannot be validated.

- For the Asserted Claims of the '137 patent, to the extent the references charted in any of the charts in Exhibit B do not expressly disclose "if the new program is not validated, monitoring the new program while it loads and executes in connection with the computing device", "if the new program is not validated … controlling the files the new program attempts to access during execution of the new program", or "if the new program is not the same as the allowed program, monitoring the new program while allowing it to execute on the computing device", "if the new program is not the same as the allowed program … controlling the files the program attempts to access during execution of the new program on the computing device", "if the new program is not the same as the allowed program … controlling the network activity of the new program", "if the new program is not valid,

monitoring the new program while allowing it to execute on the computing device", or similar claim limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Van der Made, Raduchel, Edery, Fischer, ChakraVyuha, Dan, Angelo, Dresevic, Agarwal, Attanasio, Lambert, Zwienenberg. A person of skill in the art would be motivated to combine these references because they each disclose control and monitoring of processes/programs on a computing device in the same general field. It is the nature of those of skill in the art of managed security control of processes on a computer system to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems conduct real-timing monitoring of new, unknown, and/or suspicious programs or processes and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how other solutions in this field conduct real-timing monitoring of new, unknown, and/or suspicious programs or processes. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the '137 patent, to the extent the references charted in any of the charts in Exhibit B do not expressly disclose "suspending loading of the [new] program onto the computing device", "suspending the execution of the new program on the

computing device", "interrupting the loading of a new program for operation with the computing device", "suspending the loading of the new program", or similar claim limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Van der Made, Raduchel, Edery, Fischer, ChakraVyuha, Dan, Angelo, Dresevic, Agarwal, Attanasio, Lambert, Zwienenberg. A person of skill in the art would be motivated to combine these references because they each disclose control and monitoring of processes/programs on a computing device in the same general field. It is the nature of those of skill in the art of managed security control of processes on a computer system to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems conduct real-timing monitoring of new, unknown, and/or suspicious programs or processes and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how other solutions in this field conduct real-timing monitoring of new, unknown, and/or suspicious programs or processes. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results. A person of ordinary skill would have also been motivated because a person of ordinary skill would have known that programs are loaded and executed via operating system functions, that those functions can be intercepted and interrupted, and that when validating a program to determine whether it

is trustworthy or potentially malicious, it would be preferable to halt the program from executing so that validation can occur before the program has an opportunity to load, execute, and potentially perform malicious activities.

- For the Asserted Claims of the ’137 patent, to the extent the references charted in any of the charts in Exhibit B do not expressly disclose “deciding whether to terminate the trigger intercepted by the detection module”, “terminating the execution of the new program”, or “taking remedial measures to protect the security of other computing device components”, or similar claim limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Van der Made, Raduchel, Edery, Fischer, ChakraVyuha, Dan, Angelo, Dresevic, Agarwal, Attanasio, Lambert, Zwienenberg. A person of skill in the art would be motivated to combine these references because they each disclose control and monitoring of processes/programs on a computing device in the same general field. It is the nature of those of skill in the art of managed security control of processes on a computer system to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems implement remedial, and/or protection, measures upon detection of suspicious/known virus or malware and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how other solutions in this field implement remedial, and/or protection, measures upon detection of suspicious/known virus or malware. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with

concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results. A person of ordinary skill would have also been motivated because a person of ordinary skill would have known that one way to prevent potential malicious activity from continuing is to terminate or restrict the cause of the malicious activity.

- For the Asserted Claims of the ’137 patent, to the extent the references charted in any of the charts in Exhibit B do not expressly disclose any of “a pre-execution module,” “a validation module,” “a detection module,” “a pre-execution monitor,” or an “execution module” coupled together as claimed, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Van der Made, Raduchel, Edery, Fischer, ChakraVyuha, Dan, Angelo, Dresevic, Agarwal, Attanasio, Lambert, Zwienenberg. A person of skill in the art would be motivated to combine these references because they each disclose these elements in the same general field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems operate and mimic those same techniques. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results. Furthermore, the

"coupling" of these various modules (or monitor) reflects not any invention but rather routine, obvious organization of functional software components. The specification states that "a skilled programmer would be able to write such a computer program to implement the disclosed invention based on the flow charts and associated description in the application text." '137 Patent at 5:38-41. The prior art disclosed that software components could be separated or combined as desired. *E.g.*, Lambert at 7:50-8:9. Put another way, the "modules" (or monitor) recited by the claim describe software delineated by functional terms that is in some way "coupled" together to software that performs related functions, and it was obvious to organize software in that way. It was obvious that the 'pre-execution module" that is "operable for receiving notice from the computing device's operating system that a new program is being loaded onto the computing device" would be coupled to a validation module "operable for determining whether the program is valid," particularly as the program's execution could be halted so it could then be validated, the "detection module" then coupled to the "pre-execution monitor" (or, according to Tassera, pre-execution module) "operable for intercepting a trigger from the computing device's operating system," and "an execution module" that is "coupled to the detection module and operable for monitoring, at the operating system kernel of the computing device, the program in response to the trigger intercepted by the detection module," particularly as the execution module is to be operating in response to behavior of the detection module.

- For the Asserted Claims of the '137 patent, to the extent the references charted in any of the charts in Exhibit B do not expressly disclose "wherein the step of validating the new program comprises analyzing characteristics of the new program," and to the extent the term characteristics is alleged to concern something other than hashes, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of

any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Van der Made, Raduchel, Edery, Fischer, ChakraVyuha, Dan, Angelo, Dresevic, Agarwal, Attanasio, Lambert, Zwienenberg. A person of skill in the art would be motivated to combine these references because they each disclose these elements in the same field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems operate and mimic those same techniques. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results. A person of ordinary skill in the art would also have been motivated because he or she would have known that the prior art disclosed identifying and classifying files based on their file extensions, file names, file or network path, URL zone, sender of attachment, and the like, and that these were all possible criteria for treating a program as trustworthy or not. *E.g.*, Ex. B-Lambert.

- For the Asserted Claims of the ’137 patent, to the extent the references charted in any of the charts in Exhibit B do not expressly disclose “wherein the step of monitoring the new program comprises controlling the files the new program attempts to access during execution of the new program,” it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Van

der Made, Raduchel, Edery, Fischer, ChakraVyuha, Dan, Angelo, Dresevic, Agarwal, Attanasio, Lambert, Zwienenberg. A person of skill in the art would be motivated to combine these references because they each disclose these elements in the same field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems operate and mimic those same techniques. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results. A person of ordinary skill in the art would also have been motivated because he or she would have known that a program can create security problems by trying to access files it should not, and that this can be prevented by monitoring and/or curtailing the program's access rights. *E.g.*, Lambert at 2:60-64; Ex. B-Lambert. Moreover, to the extent it is contended that this element should be construed to mean that the monitoring controls the "files" themselves, then a person of ordinary skill would have been motivated to apply the same process of validating and monitoring files as was applied to "the new program" to provide the same security procedures for all files.

- For the Asserted Claims of the '137 patent, to the extent the references charted in any of the charts in Exhibit B do not expressly disclose "wherein the step of monitoring the new program while allowing it to execute comprises controlling the network activity of the new program" (claim 20), it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of

ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Van der Made, Raduchel, Edery, Fischer, ChakraVyuha, Dan, Angelo, Dresevic, Agarwal, Attanasio, Lambert, Zwienenberg. A person of skill in the art would be motivated to combine these references because they each disclose these elements in the same field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems operate and mimic those same techniques. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results. A person of ordinary skill in the art would also have been motivated because he or she would have known that a program can create security problems by trying to perform network activity it should not, and that this can be prevented by monitoring and/or curtailing the program's access rights. *E.g.*, Lambert at 2:60-64, 17:42-51; Ex. B-Lambert.

- For the Asserted Claims of the '137 patent, to the extent the references charted in any of the charts in Exhibit B do not expressly disclose any of "a validation module" that is "operable for determining whether the program is valid" (claim 1), "validating the new program" (claim 6), "the step of validating the new program comprises determining whether the new program corresponds with an approved program" (claim 9), "validating the new program comprises comparing a checksum for the new program with a previously determined checksum" (claim 10), "validating the new program comprises analyzing

characteristics of the new program" (claim 11), "determining whether the new program is the same as the allowed program" (claim 14), "wherein the step of determining whether the new program is the same as the allowed program comprises: computing a checksum for the new program, and comparing the new program's checksum to a checksum for the allowed program" (claim 16), "the step of determining whether the new program is the same as the allowed program comprises determining whether the new program has been modified" (claim 17), "comparing the new program to the allowed program; and determining whether the new program is valid" (claim 25), "comparing the new program to the allowed program comprises comparing a checksum of the new program to a checksum of the allowed program" (claim 27), "determining whether the new program is valid comprises determining whether a checksum of the new program corresponds to a checksum of the allowed program" (claim 28) each would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Van der Made, Raduchel, Edery, Fischer, ChakraVyuha, Dan, Angelo, Dresevic, Agarwal, Attanasio, Lambert, Zwienenberg. A person of skill in the art would be motivated to combine these references because they each disclose these elements in the same field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems operate and mimic those same techniques. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover,

these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results. A person of ordinary skill in the art also would have been motivated because it was well known that trusted programs can be validated with a hash or checksum value of the program for comparison with a known valid hash or checksum value of a desired program. *E.g.*, Lambert at Abstract, 11:40-53, Ex. B-Lambert. The '137 Patent admits that the validation module was prior art: "The validation module 108 can be any one of a variety of pieces of software that are used to verify that a program has not been tampered with. For example, the validation module can represent the MD5 software module commonly known to those in the art. The MD5 software module calculates a checksum for each allowed program and that checksum is stored for later comparison. '137 Patent, 8:7-19. It was also known in the art to allow or restrict execution of a program based on file system or network path, its URL zone, the sender of an attachment, identification of information provided by a trusted application, file extension, and file name. *E.g.*, Lambert at 11:40-53, 13:4-14:29. Furthermore, by validating a program, resources can be saved that would otherwise be devoted to monitoring the program's execution for suspicious activity, and resources can be devoted to monitoring programs that cannot be validated, which would have motivated a person of ordinary skill.

- For the Asserted Claims of the '137 patent, to the extent the references charted in any of the charts in Exhibit B do not expressly disclose "wherein the execution module is further operable for deciding whether to terminate the trigger intercepted by the detection module" (claim 4), "the step of taking remedial measures to protect the security of other computing device components if the new program is not the same as the allowed program" (claim 14), "the step of terminating the execution of the new program if the program performs suspicious activities" (claim 22), it would have been obvious to combine the teachings of

any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Van der Made, Raduchel, Edery, Fischer, ChakraVyuha, Dan, Angelo, Dresevic, Agarwal, Attanasio, Lambert, Zwienenberg. A person of skill in the art would be motivated to combine these references because they each disclose these elements in the same field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems operate and mimic those same techniques. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results. A person of ordinary skill in the art would also have been motivated because he or she would have known that untrustworthy programs may be malicious and that one option to prevent malicious activity is to terminate the program or deny the suspicious action when potentially malicious activity is observed. As shown by the B Exhibits, the prior art taught validating programs based on hash values and restricting their execution either wholly or completely, and it would have been obvious to terminate a program when it attempted to perform a restricted action.

- For the Asserted Claims of the ’137 patent, to the extent the references charted in any of the charts in Exhibit B do not expressly disclose “monitoring, at the operating system kernel of the computing device,” or “wherein the step of monitoring the new program while it executes is performed at the operating system kernel of the computing device,” each

would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Van der Made, Raduchel, Edery, Fischer, ChakraVyuha, Dan, Angelo, Dresevic, Agarwal, Attanasio, Lambert, Zwienenberg. A person of skill in the art would be motivated to combine these references because they each disclose these elements in the same field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems operate and mimic those same techniques. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results. A person of ordinary skill in the art also would have been motivated because it was well known that security models could be implemented in user mode or kernel mode. Lambert at 7:50-8:8. A person of ordinary skill in the art would also have been motivated to monitor at the operating system kernel because it was known that this allowed monitoring of operating system functions such as those that open files, load files, enable files to be executed, create processes, load library functions such as DLL functions, run scripts, and perform software installation functions, and to prevent those functions from being improperly used. Lambert at 14:46-15:6, 16:25-55.

- For the Asserted Claims of the ’137 patent, to the extent the references charted in any of the charts in Exhibit B do not expressly disclose “monitoring, at the operating system

kernel of the computing device," "wherein the step of monitoring the new program while it executes is performed at the operating system kernel of the computing device," or "wherein the step of monitoring the new program comprises intercepting a signal from the computing device's operating system," each would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Van der Made, Raduchel, Edery, Fischer, ChakraVyuha, Dan, Angelo, Dresevic, Agarwal, Attanasio, Lambert, Zwienenberg. A person of skill in the art would be motivated to combine these references because they each disclose these elements in the same field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems operate and mimic those same techniques. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results. A person of ordinary skill in the art also would have been motivated because it was well known that security models could be implemented in user mode or kernel mode. Lambert at 7:50-8:8. A person of ordinary skill in the art would also have been motivated to monitor at the operating system kernel because it was known that this allowed monitoring of operating system functions such as those that open files, load files, enable files to be executed, create processes, load library functions such as DLL functions, run scripts, and perform software

installation functions, and to prevent those functions from being improperly used. Lambert at 14:46-15:6, 16:25-55.

- For the Asserted Claims of the ’137 patent, to the extent there is any requirement that monitoring be reduced or eliminated for valid programs and the references charted in any of the charts in Exhibit B do not expressly disclose that feature, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Van der Made, Raduchel, Edery, Fischer, ChakraVyuha, Dan, Angelo, Dresevic, Agarwal, Attanasio, Lambert, Zwienenberg, as well as the common sense and knowledge of a person of ordinary skill in the art. A person of skill in the art would be motivated to combine these references because they each disclose these elements in the same general field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems operate and mimic those same techniques. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results. A person of ordinary skill in the art also would have been motivated because monitoring the execution of a program takes up limited system resources. A person of skill in the art would have recognized that it is not necessary to monitor a validated program in the same way as a program that cannot be validated, and that a program that cannot be validated

presents greater security concerns than a program that is known to be trustworthy. *E.g.*, Lambert at Background.

Consistent with the exemplary motivations to combine each of the references discussed above, the below table provides a summary of the exemplary combinations of references rendering obvious to a POSA, with or without a POSA's knowledge (including the AAPA and general state of the art), skill and/or creativity, the Asserted Claims of the '137 Patent.

Reference	**Reference Alone or in Combination With One or More of the Following**
Raduchel	Van der Made, Edery, Fischer, ChakraVyuha, Dan, Angelo, Dresevic, Agarwal, Attanasio, Lambert, Zwienenberg
Van der Made	Raduchel, Edery, Fischer, ChakraVyuha, Dan, Angelo, Dresevic, Agarwal, Attanasio, Lambert, Zwienenberg
Edery	Raduchel, Van der Made, Fischer, ChakraVyuha, Dan, Angelo, Dresevic, Agarwal, Attanasio, Lambert, Zwienenberg
Fischer	Raduchel, Van der Made, Edery, ChakraVyuha, Dan, Angelo, Dresevic, Agarwal, Attanasio, Lambert, Zwienenberg
ChakraVyuha	Raduchel, Van der Made, Edery, Fischer, Dan, Angelo, Dresevic, Agarwal, Attanasio, Lambert, Zwienenberg
Dan	Raduchel, Van der Made, Edery, Fischer, ChakraVyuha, Angelo, Dresevic, Agarwal, Attanasio, Lambert, Zwienenberg
Angelo	Raduchel, Van der Made, Edery, Fischer, ChakraVyuha, Dan, Dresevic, Agarwal, Attanasio, Lambert, Zwienenberg
Dresevic	Raduchel, Van der Made, Edery, Fischer, ChakraVyuha, Dan, Angelo, Agarwal, Attanasio, Lambert, Zwienenberg
Agarwal	Raduchel, Van der Made, Edery, Fischer, ChakraVyuha, Dan, Angelo, Dresevic, Attanasio, Lambert, Zwienenberg
Attanasio	Raduchel, Van der Made, Edery, Fischer, ChakraVyuha, Dan, Angelo, Dresevic, Agarwal, Lambert, Zwienenberg
Lambert	Raduchel, Van der Made, Edery, Fischer, ChakraVyuha, Dan, Angelo, Dresevic, Agarwal, Attanasio, Zwienenberg
Zwienenberg	Raduchel, Van der Made, Edery, Fischer, ChakraVyuha, Dan, Angelo, Dresevic, Agarwal, Attanasio, Lambert

Additionally, Defendant incorporates by reference each of the obviousness combinations and motivations to combine that may be presented in any petitions for IPRs that may be filed challenging the patentability of the ’137 patent:

3. ’441 Patent

To the extent that Plaintiff contends that any of the references identified as anticipatory references fail to disclose one or more elements, those anticipatory references render the Asserted Claims obvious, either alone or in combination with other references. Defendant contends that one of skill in the art, at the time the alleged inventions were made, would have been motivated to combine the references disclosed herein in such a way as to reach the alleged inventions, as described in further detail below. Defendant presently believes a reasonable basis exists that each of the claims asserted against it would have been obvious to one of ordinary skill in the art at the time of the alleged invention.

Below, Defendant identifies their current bases for obviousness. As seen below, Defendant presently asserts certain combinations of prior art reference(s) with or without the knowledge of one of ordinary skill in the art. Defendant offers these combinations without conceding that the reference(s) standing alone are insufficient to invalidate the relevant claims. Defendant specifically reserves their rights to combine any of its anticipatory references or any combinations of references with the knowledge of ordinary skill in the art to rebut a contention of Plaintiff that such reference(s) standing alone do not disclose every element of the Asserted Claims.

Defendant lists below prior art references and combinations of prior art references now known to Defendant, which Defendant contends render at least obvious the ’441 Asserted Claims. The corresponding charts in Exhibit C for each prior art reference identifies where and how each alleged item of prior art meets one or more limitations of the Asserted Claims. Pursuant to Patent Local Rule 3-6, Defendant expressly reserves the right to supplement these charts to provide additional details about obviousness should the Court’s claim construction require, or should

Plaintiff serve adequate Infringement Contentions that better serve to notify Defendant which elements of their products allegedly correspond with the claim elements. Defendant's contention that the references in this section in various combinations render obvious the '441 Asserted Claims under 35 U.S.C. § 103 is in no way an admission or suggestion that each reference does not independently anticipate or render obvious the '441 Asserted Claims under 35 U.S.C. § 102. These combinations would have been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field.

It would been obvious to combine each of the references discussed below because they are analogous art generally directed to managed security control of processes on a computer system. The technology described in the '441 Patent was widely known before the alleged priority date of the '441 Patent and provided numerous motivations to combine the prior art references and systems disclosed herein. Methods, and defense-in-depth systems, for managed security control of processes on a computer system, including providing attestation for applications at runtime, were well known in the art, as admitted in the '441 AAPA, (*see* Section III.C above and incorporated by reference herein). For example, as admitted in the '441 AAPA, "many enterprises are moving there IT infrastructures to cloud computing environments (e.g., in which third parties may provide shared computing resources and applications running on those resources are offered as services to a plurality of customers)" and thus such techniques were accessible to those skilled in the art.

The '441 Patent combines these components according to their known, predictable and intended uses, just as they were and/or would have been combined by a POSA in the prior art. None of the combinations presents an unexpected result, a counterintuitive combination, or a technological challenge; and no such description exists in the common specification of the asserted patents.

Accordingly, the prior art cited herein would have been readily combined according to methods known to those of ordinary skill within the field of managed security control of processes on a computer system to yield predictable results. The substitution of one element in this system could have been predictably achieved by one of ordinary skill at the time of the alleged invention. One of ordinary skill in the art would have been able to select appropriate attributes of one for inclusion in another. Those of ordinary skill in the art could have employed known techniques to improve similar prior art devices in the same way as claimed in the ’441 Patent. Thus, the combinations all have well defined, predictable, expectation of success.

- For the Asserted Claims of the ’441 patent, to the extent the references charted in any of the charts in Exhibit C do not expressly disclose “a runtime execution context indicating attributes of the application at runtime” and “wherein the attributes comprise one or more executable file binaries of the application and loaded components of the application” or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Starnes, Lockhart, Friedrichs, Greene, Tuvell, Cha, Guzman, Leone, Reissman, Long Luo, Matsumoto, Bose, Leone, Munetoh, Rajan, and/or OfficeScan which disclose a runtime execution context (e.g. a system state) indicating attributes of the application at runtime, wherein the attributes comprise one or more executable file binaries of the application and loaded components of the application. As an example, Lockhart discloses a system and method in which an attestation server receives continuous application assessments, even as the application operates (i.e., a runtime execution context indicating attributes of the application at runtime), including “source code” or “binary code” (i.e., executable file binaries) and “one or more components of a software application from a remote site” (i.e.,

loaded components of the application). Similarly, Starnes discloses receiving by an attestation server a runtime application profile, including information regarding properties of all application package components including dynamically loadable modules, as well as and scripts and intermediate code elements. Likewise, Greene discloses receiving at an attestation server "application architecture information for a software application," including code for the application and "components that constitute[] a software application." Friedrichs discloses receiving at an attestation server metadata about an application, contextual information, and the application itself (i.e., executable file binaries and loaded components). Matsumoto discloses a certification server (i.e., attestation server) receiving certification information about an application. Reissman discloses a computer (i.e., attestation server) receiving states, guards, and transitions (i.e., execution and security contexts). A person of skill in the art would be motivated to combine these references because they each disclose methods and systems for providing attestation of a computing environment and thus are in the same general field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems provide attestation services and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how other solutions in this field obtained runtime information from monitored systems. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable

results. A person of ordinary skill would also have been motivated because these are just data points that can be evaluated when assessing the security risks of an application.

- For the Asserted Claims of the ’441 patent, to the extent the references charted in any of the charts in Exhibit C do not expressly disclose “a security context providing security information about the application, wherein the security information comprises an execution analysis of the one or more executable file binaries and the loaded components,” “an introspective security context” and/or “static, dynamic, or virtual” execution analysis, or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Starnes, Lockhart, Friedrichs, Greene, Tuvell, Cha, Guzman, Leone, Reissman, Long Luo, Matsumoto, Bose, Leone, Munetoh, Rajan, and/or OfficeScan which disclose a security context (e.g. integrity information) providing security information about the application, wherein the security information comprises an execution analysis of the one or more executable file binaries and the loaded components. As an example, Lockhart discloses receiving at the attestation server business context information, generated “though various testing techniques such as dynamic application testing, static binary testing, automated and manual pen testing, as well as manual code review” (i.e., an execution analysis of the executable file binaries and loaded components), including “dynamic” execution analysis, as well as analysis that is “automated, manual, static, dynamic, etc.”, and that testing can be “combined into a ‘custom’ or ‘application-specific’ workflow, and the results of each test may be correlated with other results to identify a wide range of potential vulnerabilities and/or faults that are detected by different tests,” “correlated to facilitate their review within the context and/or portion of the application in which flaws or vulnerabilities were

found," and "repeated as indications of new threats or updates to the application are received" (i.e., introspective). Starnes discloses receiving at an attestation server information regarding application process integrity and security posture, including through the generation of a trust score via interrogation of the target running application, and that the security context is based on trust evaluation over time and in different contexts, including recent measurements and verifications (i.e., introspective), and that analysis can be static or dynamic. Greene discloses receiving at an attestation server an execution analysis of the code for the software application and the components and running security tests on the code of the software application based on the application architecture information, as well as that the security context includes historical security information that is combined with specific details of the application architecture information (i.e., introspective), and discloses that the execution analysis may include use of a "a static source code analysis tool to identify security vulnerabilities". Friedrichs discloses computing a signature for an application and determining whether the signature matches known malicious software, whitelists, or blacklists, as well as monitoring what rights the application is looking to obtain, the state of the application, etc., that the security context includes historical information, geographical information, recent infections, including analysis and information collected over a period of time (i.e., introspective). Matsumoto discloses a certification server (i.e., attestation server) receiving certification information about an application. Reissman discloses a computer (i.e., attestation server) receiving states, guards, and transitions (i.e., execution and security contexts). A person of skill in the art would be motivated to combine these references because they each disclose methods and systems for providing attestation of a computing environment and thus are in the same general field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A

person of skill would have been motivated to look at how other products and systems provide attestation services and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how other solutions in this field obtained runtime information from monitored systems. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results. A person of ordinary skill would also have been motivated because these are just data points that can be evaluated when assessing the security risks of an application.

- For the Asserted Claims of the ’441 patent, to the extent the references charted in any of the charts in Exhibit C do not expressly disclose “generating” or “sending” “an application artifact” for tracking changes in execution context, or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Starnes, Lockhart, Friedrichs, Greene, Tuvell, Cha, Guzman, Leone, Reissman, Long Luo, Matsumoto, Bose, Leone, Munetoh, Rajan, and/or OfficeScan. For example, Leone discloses generating, by the attestation server, an application artifact (e.g. an dynamic security level) as a reference for changes in a subsequent execution context. Starnes discloses generating and sending an application token for the running application as a reference for subsequent real time notification of application state changes (i.e., as a reference for subsequent changes in execution context) and discloses that “any state changes in the running application trigger the interactions to

refresh the application token" (i.e., subsequent changes are tracked based on the artifact). Lockhart discloses "the infrastructure and methods for continuous, periodic or event-triggered application assessments, even as the application operates in a secure production environment" and "continuous application assurance provides for automatic re-analysis of an application" including where "[r]e-analysis is triggered by changes in the external application environment [such as] detected attacks," which a person of ordinary skill would understand to encompass the use of an application artifact for tracking changes in execution context. Friedrichs discloses "continuously gather[ing] contextual information," for a security determination, "deriving a model based on the determination, the model encoding rules to be utilized in making future determinations when a second set of contextual information is similar to the first set of contextual information," including over different time periods, and through use of "a record of transactions and actions made on a given system," or "log," which a person of ordinary skill would understand to encompass the use of an application artifact for tracking changes in execution context. Matsumoto discloses generating and sending certification information (i.e., a report) at the request of the user. Reissman discloses generating and sending a report of the results of the testing and validation of the application (i.e., a report indicating security risks). Bose discloses investigating parent-child process relationships as attributes that are monitored. A person of skill in the art would be motivated to combine these references because they each disclose methods and systems for providing attestation of a computing environment and thus are in the same general field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems provide attestation services and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how other solutions in this field

obtained runtime information from monitored systems. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’441 patent, to the extent the references charted in any of the charts in Exhibit C do not expressly disclose “digitally signing” attestation results, or wherein “attributes further comprise parent-child process associations of the application”, or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Starnes, Lockhart, Friedrichs, Greene, Tuvell, Cha, Guzman, Leone, Reissman, Long Luo, Matsumoto, Bose, Leone, Munetoh, Rajan, and/or OfficeScan. For example, Starnes discloses that the disclosed “application token may include a digitally signed token” and that the attributes comprising the runtime security context include “properties of all application package components,” which a person of ordinary skill would understand to include parent-child process associations. Lockhart discloses the use of various security measures to keep secure and confidential attestation results, such as DRM, secure containerizing, the use of APIs to transform structured data, etc., all of which a person of ordinary skill would understand to include digitally signing the results, and further discloses that attributes comprising the runtime security context include information regarding which applications including third party applications that interact with the application and the security environment in which the application will operate, all of which

a person of ordinary skill would understand to include parent-child process associations. Greene discloses that the attributes comprising the runtime security context include information regarding application characteristics including clustering, how the application connects to other systems, package installation environment, etc., all of which a person of ordinary skill would understand to include parent-child process associations. Friedrichs discloses the use of "digital signatures" as a "standard technology" for verification, and that the attributes comprising the runtime security context include information such as DLLs referenced by the application, services stopped or started by the application, behavioral attributes of the application, registry keys and modifications to the registry, as well as API calls made by the application, all of which a person of ordinary skill would understand to include parent-child process associations. A person of skill in the art would be motivated to combine these references because they each disclose methods and systems for providing verified attestation of a computing environment and thus are in the same general field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems provide verified attestation services and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how other solutions in this field verified attestation results and the scope of attributes monitored by such monitored systems. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’441 patent, to the extent the references charted in any of the charts in Exhibit C do not expressly disclose “authenticating the application using a plurality of collaboration services”, or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Starnes, Lockhart, Friedrichs, Greene, Tuvell, Cha, Guzman, Leone, Reissman, Long Luo, Matsumoto, Bose, Leone, Munetoh, Rajan, and/or OfficeScan. For example, Starnes discloses authenticating an application with a plurality of collaboration services, including, e.g., whitelisting services. Likewise, Lockhart discloses authenticating an application with a plurality of collaboration services, including, e.g., the owner of the application, a licensee, or a third party, and may include an enterprise-wide collection of applications, or industry-wide data, as well as cross-application reporting and peer benchmarking. Friedrichs discloses authenticating an application with a plurality of collaboration services, including, e.g., whitelisting and blacklisting services. A person of skill in the art would be motivated to combine these references because they each disclose methods and systems for providing application authentication and attestation of a computing environment and thus are in the same general field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems provide application authentication and attestation services and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how other solutions in this field authenticate applications and integrate, e.g., existing whitelisting and blacklisting services. These combinations would have also been obvious to try given that there were a finite number of

identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’441 patent, to the extent the references charted in any of the charts in Exhibit C do not expressly disclose “controlling a user’s transaction with” or “network access to the application by applying a set of authorization rules in accordance with the attestation results” and the use of “confidence metrics in the attestation results indicating a level of security risk by different classifications”, or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Starnes, Lockhart, Friedrichs, Greene, Tuvell, Cha, Guzman, Leone, Reissman, Long Luo, Matsumoto, Bose, Leone, Munetoh, Rajan, and/or OfficeScan. For example, Starnes discloses controlling user access, including network access, to an application based on the disclosed attestation system including via “access controls” and further discloses the use of weighed trust scores indicating security risks and controlling access based on these trust scores. Likewise, Lockhart discloses controlling user access, including network access, to an application through the use of access tokens based on attestation results, including the use of “assessment” and “assurance” levels which constitute confidence metrics in the attestation results indicating a level of security risk by different classifications, and which are used to control access to the applications. Greene discloses controlling user access, including network access, to an application based on the disclosed attestation system such that if an application fails a

security test a user will be required to address it before the application may be used or accessed, as well as the use of "low, medium or high risk-rankings" and controlling access based on such risks. Friedrichs discloses controlling user access, including network access, to an application based on the disclosed attestation system including by blocking access or deleting an application should the attestation result so require, and further discloses the use of confidence scores that reflect confidence in the classification of a program's risk, and controlling access based on these confidence levels in certain circumstances. Luo teaches a method whereby a user's access is controlled by a set trigger authentication policy on an application server in accordance with attestation results which can control a user's transaction with the application. A person of skill in the art would be motivated to combine these references because they each disclose methods and systems for providing application authentication and attestation of a computing environment and thus are in the same general field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems provide application authentication and attestation services and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how other solutions in this field control access to applications subject to attestation testing and analysis. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

Consistent with the exemplary motivations to combine each of the references discussed above, the below table provides a summary of the exemplary combinations of references rendering obvious to a POSA, with or without a POSA's knowledge (including the AAPA and general state of the art), skill and/or creativity, the Asserted Claims of the '441 Patent.

Reference	**Reference Alone or in Combination With One or More of the Following**
Starnes	Lockhart, Friedrichs, Greene, Tuvell, Cha, Guzman, Leone, Reissman, Long Luo, Matsumoto, Bose, OfficeScan, Kil, Kuppusamy, Munetoh, Rajan
Lockhart	Starnes, Friedrichs, Greene, Tuvell, Cha, Guzman, Leone, Reissman, Long Luo, Matsumoto, Bose, OfficeScan, Kil, Kuppusamy, Munetoh, Rajan
Friedrichs	Starnes, Lockhart, Greene, Tuvell, Cha, Guzman, Leone, Reissman, Long Luo, Matsumoto, Bose, OfficeScan, Kil, Kuppusamy, Munetoh, Rajan
Greene	Starnes, Lockhart, Friedrichs, Tuvell, Cha, Guzman, Leone, Reissman, Long Luo, Matsumoto, Bose, OfficeScan, Kil, Kuppusamy, Munetoh, Rajan
Tuvell	Starnes, Lockhart, Friedrichs, Greene, Cha, Guzman, Leone, Reissman, Long Luo, Matsumoto, Bose, OfficeScan, Kil, Kuppusamy, Munetoh, Rajan
Cha	Starnes, Lockhart, Friedrichs, Greene, Tuvell, Guzman, Leone, Reissman, Long Luo, Matsumoto, Bose, OfficeScan, Kil, Kuppusamy, Munetoh, Rajan
Guzman	Starnes, Lockhart, Friedrichs, Greene, Tuvell, Cha, Leone, Reissman, Long Luo, Matsumoto, Bose, OfficeScan, Kil, Kuppusamy, Munetoh, Rajan
Leone	Starnes, Lockhart, Friedrichs, Greene, Tuvell, Cha, Guzman, Reissman, Long Luo, Matsumoto, Bose, OfficeScan, Kil, Kuppusamy, Munetoh, Rajan
Reissman	Reissman, Starnes, Lockhart, Friedrichs, Greene, Tuvell, Cha, Guzman, Leone, Long Luo, Matsumoto, Bose, OfficeScan, Kil, Kuppusamy, Munetoh, Rajan
Long Luo	Starnes, Lockhart, Friedrichs, Greene, Tuvell, Cha, Guzman, Leone, Reissman, Matsumoto, Bose, OfficeScan, Kil, Kuppusamy, Munetoh, Rajan
Matsumoto	Starnes, Lockhart, Friedrichs, Greene, Tuvell, Cha, Guzman, Leone, Reissman, Long Luo, Bose, OfficeScan, Kil, Kuppusamy, Munetoh, Rajan
Bose	Starnes, Lockhart, Friedrichs, Greene, Tuvell, Cha, Guzman, Leone, Reissman, Long Luo, Matsumoto, OfficeScan, Kil, Kuppusamy, Munetoh, Rajan
OfficeScan	Starnes, Lockhart, Friedrichs, Greene, Tuvell, Cha, Guzman, Leone, Reissman, Long Luo, Matsumoto, Bose, Kil, Kuppusamy, Munetoh, Rajan

Reference	Reference Alone or in Combination With One or More of the Following
Kil	Starnes, Lockhart, Friedrichs, Greene, Tuvell, Cha, Guzman, Leone, Reissman, Long Luo, Matsumoto, Bose, OfficeScan, Kuppusamy, Munetoh, Rajan
Kuppusamy	Starnes, Lockhart, Friedrichs, Greene, Tuvell, Cha, Guzman, Leone, Reissman, Long Luo, Matsumoto, Bose, OfficeScan, Kil, Munetoh, Rajan
Munetoh	Starnes, Lockhart, Friedrichs, Greene, Tuvell, Cha, Guzman, Leone, Reissman, Long Luo, Matsumoto, Bose, OfficeScan, Kil, Kuppusamy, Rajan
Rajan	Starnes, Lockhart, Friedrichs, Greene, Tuvell, Cha, Guzman, Leone, Reissman, Long Luo, Matsumoto, Bose, OfficeScan, Kil, Kuppusamy, Munetoh

Additionally, Defendant incorporates by reference each of the obviousness combinations and motivations to combine presented in the petition for the following *inter partes* review proceeding(s) (IPRs) and any additional petitions for IPRs that may be filed challenging the patentability of the ’441 patent:

- IPR2023-00801

4. ’517 Patent

To the extent that Plaintiff contends that any of the references identified as anticipatory references fail to disclose one or more elements, those anticipatory references render the Asserted Claims obvious, either alone or in combination with other references. Defendant contends that one of skill in the art, at the time the alleged inventions were made, would have been motivated to combine the references disclosed herein in such a way as to reach the alleged inventions, as described in further detail below. Defendant presently believes a reasonable basis exists that each of the claims asserted against it would have been obvious to one of ordinary skill in the art at the time of the alleged invention.

Below, Defendant identifies their current bases for obviousness. As seen below, Defendant presently asserts certain combinations of prior art reference(s) with or without the knowledge of one of ordinary skill in the art. Defendant offers these combinations without conceding that the

reference(s) standing alone are insufficient to invalidate the relevant claims. Defendant specifically reserves their rights to combine any of its anticipatory references or any combinations of references with the knowledge of ordinary skill in the art to rebut a contention of Plaintiff that such reference(s) standing alone do not disclose every element of the Asserted Claims.

Defendant lists below prior art references and combinations of prior art references now known to Defendant, which Defendant contends render at least obvious the '517 Asserted Claims. The corresponding charts in Exhibit D for each prior art reference identifies where and how each alleged item of prior art meets one or more limitations of the Asserted Claims. Pursuant to Patent Local Rule 3-6, Defendant expressly reserves the right to supplement these charts to provide additional details about obviousness should the Court's claim construction require, or should Plaintiff serve adequate Infringement Contentions that better serve to notify Defendant which elements of their products allegedly correspond with the claim elements. Defendant's contention that the references in this section in various combinations render obvious the '517 Asserted Claims under 35 U.S.C. § 103 is in no way an admission or suggestion that each reference does not independently anticipate or render obvious the '517 Asserted Claims under 35 U.S.C. § 102. These combinations would have been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field.

It would been obvious to combine each of the references discussed below because they are analogous art generally directed to assessing the runtime risk behavior of an application or device in a computer system. The technology described in the '517 Patent was widely known before the alleged priority date of the '517 Patent and provided numerous motivations to combine the prior art references and systems disclosed herein. As admitted in the '517 AAPA, (*see* Section III.C above and incorporated by reference herein), "antivirus programs, firewalls, and intrusion detection/prevention systems (IDSs/IPSs)", and other "programs and approaches … to provide

security and protect end users from harmful threats", were well known in the art. The cited art demonstrates that, at the time of the alleged priority date of the '517 Patent (January 2013), solutions to detect and prevent threats by advanced persistent threats—the problem that the '517 Patent states it intended to solve ('517 Patent, 1:24-39)—including methods and systems that assessed the runtime risk behavior of an application running on a computing device were well known in the art. The '517 AAPA also expressly acknowledges that its purported solution to solve this problem does not require any specialized technology, and that "other configurations and systems for performing the functions disclosed [in the '517 Patent specification] may be suitable technology." *See* '517 Patent at 2:43-54, 2:58-62, 2:66-3:3, 3:8-12, 3:16-20, 3:35-37, 4:28-32, 4:45-51, 4:56-64, 5:7-11, 6:56-62. Accordingly, the '517 Patent solution admits that it combines these conventional components according to their known, predictable and intended uses, just as they were and/or would have been combined by a POSA in the prior art. None of the combinations presents an unexpected result, a counterintuitive combination, or a technological challenge; and no such description exists in the common specification of the asserted patents.

Likewise, the prior art cited herein would have been readily combined according to methods known to those of ordinary skill within the field of managed security control of processes on a computer system to yield predictable results. The substitution of one element in this system could have been predictably achieved by one of ordinary skill at the time of the alleged invention. One of ordinary skill in the art would have been able to select appropriate attributes of one for inclusion in another. Those of ordinary skill in the art could have employed known techniques to improve similar prior art devices in the same way as claimed in the '517 Patent. Thus, the combinations all have well defined, predictable, expectation of success.

- For the Asserted Claims of the '517 patent, to the extent the references charted in any of the charts in Exhibit D do not expressly disclose "storing, in a policy database, a plurality of assessment policies" and/or "storing in a rules database a plurality of rules", or similar

limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Bettini, which expressly discloses storing rules in a rules database and policies in a policy database, Dalcher, VIDES, ThunderBYTE, Cuckoo Sandbox, Satish, Bettini, Nicodemus, Yeh, Chien, Alperovitch, Bose, and/or OfficeScan. *See, e.g.,* Bettini at Fig. 3, Dalcher at Figs. 3, 5, ThunderBYTE at ANTI-VIR.DAT files, ANTI-VIR.DAT records, VIDES at pp. 79, 85, OfficeScan at 5-62, 7-6, Alperovitch (application manifest database 32, policies database), Yeh at 6:40-45 (rules database 130), Chien at 1:56-59 (malware behavior database). A person of skill in the art would be motivated to combine these references because they each disclose assessing the runtime risk of an application executing on a computer through the application of rules and policies in the same general field. It is the nature of those of skill in the art of computer and application security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems store and manage rules and policies and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how other solutions in this field maintained databases of rules and policies. The concept of storing information in databases, including policies and rules, was also well known to a person of ordinary skill in the art, and a person of ordinary skill in the art would have considered it a matter of design choice whether to store information in one database or another. A person of ordinary skill would also have been motivated because storing policies and rules in separate databases could preserve one or the other in the event of a database failure or compromise. These combinations would have also been obvious to try given that there were a finite number of identified, predictable

solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’517 patent, to the extent the references charted in any of the charts in Exhibit D do not expressly disclose an “action sequence”, including a sequence of at least two performed actions of a “user action,” an “application action,” and/or a “system action,” or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Satish (which teaches an “action sequence” (e.g. event sequencing), as well as a “user action” (e.g. keystrokes), an “application action” (e.g. an OS activity performed by a process) and a “system action” (e.g. registry modify event types)), Dalcher (which discloses an “an action sequence” as well as a “user action” (e.g., selecting the launch of a web browser), an “application action” (e.g., browser application launching action), and a “system action” (e.g., the system monitoring the browser application in a sandbox environment), Bose (which discloses an “action sequence” as well as a “user action” (e.g., granting permission to install), an “application action” (e.g., installing a payload, starting a process), and a “system action” (e.g., creating/rebuilding or placing/saving a file), OfficeScan (which discloses “an action sequence” as well as a “user action” (e.g., user opens an attachment or clicks on a phishing email), an “application action” (e.g., downloading malware and/or searching for files to encrypt), and a “system action” (e.g., encrypting files)), Alperovitch (which discloses “an action sequence” as well as a “user action” (e.g., user choosing to download and install an application and a user providing

permission to the reputation engine to access the mobile device), an "application action" (e.g., requests for mobile device resources, such as access to files and pictures stored on the mobile device, access to cameras and microphones), and a "system action" (e.g., mobile device's processor turning on the mobile device microphone and camera, initiating a network connection with a rogue IP address, starting a virtual game of cards)), and/or Chien (which discloses an "action sequence," i.e., a sequence of at least two performed actions, as a behavior list, which is a sequence or stream of events). A person of skill in the art would be motivated to combine these references because they each disclose assessing the runtime risk of an application executing on a computer through the use of action sequences and rules in the same general field. It is the nature of those of skill in the art of computer and application security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems monitor applications through the application of rules and events and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how other solutions in this field analyze sequences of actions. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results. Indeed, as acknowledged by the specification of the '517 patent (which states that "the user action … may be *any form* of explicit input from a user of the computing system"), there is no technological difference between, nor any specific technology changes to accommodate, one well known "user action" over another. Likewise, as acknowledged by the specification of the '517 patent (which states that "the

application action … may be *any activity* performed by the application initiated programmatically by a task in execution of the computing system"), there is no technological difference between, nor any specific technology changes to accommodate, one well known "application action" over another. Similarly, as acknowledged by the specification of the '517 patent (which states that "the system action … may be a*ny operation* performed by the computing system ... that changes the state of the computing system"), there is no technological difference between, nor any specific technology changes to accommodate, one "system action" over another. '517 patent, 9:17-24 (emphasis added). Moreover, while the specification of the '517 patent teaches that a system action … may be a*ny operation* performed by the computing system ... *on behalf or, or as a consequence of* a user action ... or application action", the claims are drafted such that, an action sequence can be of any of such actions being performed in any order. *see, e.g.*, claims 1, 5-9, 13, 17-21. Furthermore, as acknowledged by the specification of the '517 patent, there is no technological difference between a system action being triggered by, or being performed on behalf of, a user action over an application action, nor any specific technology changes to accommodate a system action being triggered by, or being performed on behalf of, a user action over an application action, *Id*., 9:33-50. Thus, it is simply a design choice to select an action sequence from a finite number and order of well-known user actions, application actions, and system actions to yield predictable results because doing so would involve nothing more than applying a known technique to a known method that was ready for improvement.

- For the Asserted Claims of the '517 patent, to the extent the references charted in any of the charts in Exhibit D do not expressly disclose "a plurality of rules, wherein each rule identifies an action sequence", or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common

sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Bettini, Dalcher, VIDES, ThunderBYTE, Satish, Yeh, Chien, Alperovitch, Bose, and/or OfficeScan. *See, e.g.,* Bettini at Fig. 3; Dalcher at Figs. 3, 5; ThunderBYTE at ANTI-VIR.DAT files, ANTI-VIR.DAT records; VIDES at pp. 79, 85; OfficeScan at 5-62, 7-6; Satish at 1:50-67, 6:29-42 ("a set of rules (e.g., if-then-else statements)" that each identify an "event sequence"); Alperovitch at [0055] (application manifest database that aggregates application behaviors of a mobile device for use in determining the reputation score of an application); Yeh at 2:4-7 ("anomaly monitoring software on the user computer detects [suspect] activities that … are compared against a rules database on the user computer"); Chien at 3:62-65 ("a behavior list is a sequence or stream of events"), 4:8-13. A person of skill in the art would be motivated to combine these references because they each disclose assessing the runtime risk of an application executing on a computer through the application of rules and policies in the same general field. It is the nature of those of skill in the art of computer and application security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems identify action sequences indicative of risks or threats using rules and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how other solutions in this field identified action sequences indicative of risks or threats using rules. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was

ready for improvement to yield predictable results.

- For the Asserted Claims of the ’517 patent, to the extent the references charted in any of the charts in Exhibit D do not expressly disclose a method that includes an action sequence of a user action performed before an application action, an action sequence of an application action performed before a system action, an action sequence of a user action performed before a system action, and/or an action sequence including “three performed actions, a first action of the three performed actions is a user action, a second action of the three performed actions is an application action, a third action of the three performed actions is a system action, the second action is performed temporally after the first action, and the third action is performed temporally after the second action”, or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Satish which teaches an “action sequence” (e.g. event sequencing), as well as a “user action” (e.g. keystrokes), an “application action” (e.g. an OS activity performed by a process) and a “system action” (e.g. registry modify event types), Dalcher, VIDES, ThunderBYTE AV, Cuckoo Sandbox, Satish, Bettini, Nicodemus, Yeh, Chien, Alperovitch, Bose, and/or OfficeScan. *See also, e.g.*, Dalcher (disclosing a first action of a user launching a web browser, which triggers the browser application action, which triggers the system to monitor the browser application in a sandbox environment); Bose (disclosing the first action is a user action (e.g., granting permission to install), and the second action is an application action (e.g., installing a payload, starting a process) occurring at a time after the first action, and the third action is a system action (e.g. creating/rebuilding or placing/saving a file) occurring at a time after the second action); OfficeScan (disclosing first action is a user action (e.g.,

user opens an attachment or clicks on a phishing email), the second action is an application action (e.g., downloading malware and/or searching for files to encrypt) occurring at a time after the first action, and the third action is a system action (e.g. encrypting files) occurring at a time after the second action); Alperovitch at [0041]; [0054]. A person of skill in the art would be motivated to combine these references because they each disclose assessing the runtime risk of an application executing on a computer through the use of action sequences and rules in the same general field. It is the nature of those of skill in the art of computer and application security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems monitor applications through the application of rules and events and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how other solutions in this field analyze sequences of actions. The concept of sequencing actions on a computing environment, including user, application, and system actions, was also well known to a person of ordinary skill in the art, and a person of ordinary skill in the art would have considered it as simply a matter of design choice (as acknowledged by the ’517 patent specification). These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’517 patent, to the extent the references charted in any of the charts in Exhibit D do not expressly disclose a “forensic chain of events” described by an action sequence of at least two performed “user action(s)”, “application action(s)” or “system action(s)”, or similar limitations, it would have been obvious to combine the

teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Satish which teaches an using at least ordered sequences (e.g. event sequencing), proximity analysis (e.g. timing information), and to assess whether the at least two performed actions of the action sequence describe a forensic chain of events (e.g. specific event pair based on sequencing information and timing), ThunderBYTE AV (which discloses using heuristics analysis to detect instruction sequences common to viruses), and/or Alperovitch (which discloses the reputation engine and agent are assessing and investigating the actions of the action sequence as a chain of events, such as turning on a camera and a microphone (or other recording device) of a mobile device or running a purported game application and sending unauthorized SMS messages), also including Dalcher, VIDES, Cuckoo Sandbox, Satish, Bettini, Nicodemus, Yeh, Chien, Bose, and/or OfficeScan. A person of skill in the art would be motivated to combine these references because they each disclose assessing the runtime risk of an application executing on a computer through the use of action sequences and rules in the same general field. It is the nature of those of skill in the art of computer and application security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems monitor applications through the application of rules and events and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how other solutions in this field analyze sequences of actions. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field.

Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’517 patent, to the extent the references charted in any of the charts in Exhibit D do not expressly disclose “identifying … a behavior score for the application program that executes on the device based on the identified runtime risk”, or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, ThunderBYTE AV which discloses a heuristic flag that denotes a score, and where the sum of scores for each triggered flag is compared to a predefined limit to identify a virus, Bose (which discloses degree of certainty or confidence level, class label of +1 or -1) for the application program that executes on the device based on the identified runtime risk), OfficeScan (which discloses current number of infected computers from predictive, preventive detection, reputation scores) for the application program that executes on the device based on the identified runtime risk), and/or Alperovitch (which discloses a reputation engine calculating reputation scores after analyzing application behavior, and where the calculated reputation score may indicate a probability that an application is a malicious software, e.g., classified into a risk category denoted by color), also including Dalcher, VIDES, Cuckoo Sandbox, Satish, Bettini, Nicodemus, Yeh, Chien, and/or Satish . A person of skill in the art would be motivated to combine these references because they each disclose assessing the runtime risk of an application executing on a computer through the application of rules and policies in the same general field. It is the nature of those of skill in the art of computer and application security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have

been motivated to look at how other products and systems indicate the risk or threat of an application program that executes on a device to the device, a computing system including the device, or another device connected to the computing system, and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how other solutions in this field indicated the risk or threat of an application program that executes on a device to the device, a computing system including the device, or another device connected to the computing system. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’517 patent, to the extent the references charted in any of the charts in Exhibit D do not expressly disclose “the runtime risk is identified based on matching rules and probabilistic weights and scores of performed actions on the device”, or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Alperovitch which discloses “[a]n overall reputation score may be determined based upon the calculated probabilities”, where the calculated probabilities include “aggregated behaviors of the application that may also be evaluated to determine a reputation score,” and “[a]s more information about an application is reported or otherwise made available to reputation engine, a statistical confidence level of the reputation score may be higher,” also including Dalcher, VIDES, ThunderBYTE, Cuckoo Sandbox, Satish, Bettini, Nicodemus, Yeh,

Chien, Bose, and/or OfficeScan. A person of skill in the art would be motivated to combine these references because they each disclose assessing the runtime risk of an application executing on a computer through the application of rules and policies in the same general field. It is the nature of those of skill in the art of computer and application security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems identify the runtime risk of an application program that executes on a device to the device, a computing system including the device, or another device connected to the computing system, and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how other solutions in this field identified the runtime risk of an application program that executes on a device to the device, a computing system including the device, or another device connected to the computing system. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. The '517 patent admits as much as it identifies using "probabilistic weights and scores" as one of a finite number of alternatives for identifying runtime risk. *See, e.g.*, '517 patent at 8:1-9, 9:5-13, 9:25-32. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results including higher accuracy, and increased statistical confidence level of the calculated runtime risk.

- For the Asserted Claims of the '517 patent, to the extent the references charted in any of the charts in Exhibit D do not expressly disclose "the user action is any form of explicit input from a user of a computing system", or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge,

common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Satish which teaches an "action sequence" (e.g. event sequencing), as well as a variety of actions, including, a "user action" that "is a[] form of explicit input from a user of a computing system" (e.g. keystrokes), , Dalcher (e.g., selecting the launch of a web browser), Bose (e.g., granting permission to install an application), OfficeScan (e.g., user opens an attachment or clicks on a phishing email), and/or Alperovitch (e.g., user choosing to download and install an application and a user providing permission to the reputation engine to access the mobile device), also including, VIDES, ThunderBYTE, Cuckoo Sandbox, Bettini, Nicodemus, Yeh, Chien, Bose. A person of skill in the art would be motivated to combine these references because they each disclose assessing the runtime risk of an application executing on a computer through the use of action sequences and rules in the same general field. It is the nature of those of skill in the art of computer and application security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems identify the runtime risk of an application program that executes on a device to the device, a computing system including the device, or another device connected to the computing system, and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how other solutions in this field identified the runtime risk of an application program that executes on a device to the device, a computing system including the device, or another device connected to the computing system based upon observed actions. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior

art in the same field. Moreover, these combinations would have been the result of a mere design choice from a finite number and order of well-known user actions, application actions, and system actions to yield predictable results because doing so would simply involve applying known techniques to a known method that was ready for improvement.

- For the Asserted Claims of the ’517 patent, to the extent the references charted in any of the charts in Exhibit D do not expressly disclose “the application action is any activity performed by the application initiated programmatically by a task in execution of a computing system”, or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Satish which teaches an “action sequence” (e.g. event sequencing), as well as a variety of actions, including, an “application action” that is “an activity performed by the application initiated programmatically by a task in execution of a computing system” (e.g. an OS activity performed by a process) , Dalcher (e.g., browser application launching action), Bose (e.g., installing a payload, starting a process), OfficeScan (e.g., downloading malware and/or searching for files to encrypt), and/or Alperovitch (e.g., requests for mobile device resources, such as access to files and pictures stored on the mobile device, access to cameras and microphones), as well as VIDES, ThunderBYTE, Cuckoo Sandbox, , Bettini, Nicodemus, Yeh, Chien. A person of skill in the art would be motivated to combine these references because they each disclose assessing the runtime risk of an application executing on a computer through the use of action sequences and rules in the same general field. It is the nature of those of skill in the art of computer and application security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems

identify the runtime risk of an application program that executes on a device to the device, a computing system including the device, or another device connected to the computing system, and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how other solutions in this field identified the runtime risk of an application program that executes on a device to the device, a computing system including the device, or another device connected to the computing system based upon observed actions. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of a mere design choice from a finite number and order of well-known user actions, application actions, and system actions to yield predictable results because doing so would simply involve applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’517 patent, to the extent the references charted in any of the charts in Exhibit D do not expressly disclose “the system action is any operation performed by a computing system on behalf of, or as a consequence of, a user action or application action that changes the state of the computing system”, or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Satish which teaches an “action sequence” (e.g. event sequencing), as well as a variety of actions, including, and a “system action” that is an “operation performed by a computing system on behalf of, or as a consequence of, a user action or application action that changes the state of the computing

system" (e.g. registry modify event types performed on behalf of, or as a consequence of, a "user action" (e.g. keystrokes) and an "application action" (e.g. an OS activity performed by a process)), Dalcher (a "system action" (e.g., the system monitoring the browser application in a sandbox environment) performed on behalf of, or as a consequence of, a user launching a web browser and a browser application action) and an "application action" (e.g., installing a payload, starting a process)), Bose (a "system action" (e.g., creating/rebuilding or placing/saving a file) performed on behalf of, or as a consequence of, a user action (e.g., granting permission to install), and an application action (e.g., installing a payload, starting a process)user action (e.g., granting permission to install)), OfficeScan (a "system action" (e.g., encrypting files) performed on behalf of, or as a consequence of, a "user action" (e.g., user opens an attachment or clicks on a phishing email) and an "application action" (e.g., downloading malware and/or searching for files to encrypt), and/or Alperovitch (a "system action" (e.g., mobile device's processor turning on the mobile device microphone and camera, initiating a network connection with a rogue IP address, starting a virtual game of cards) performed on behalf of, or as a consequence of, a "user action" (e.g., user choosing to download and install an application and a user providing permission to the reputation engine to access the mobile device) and an "application action" (e.g., requests for mobile device resources, such as access to files and pictures stored on the mobile device, access to cameras and microphones)). A person of skill in the art would be motivated to combine these references because they each disclose assessing the runtime risk of an application executing on a computer through the use of action sequences and rules in the same general field. It is the nature of those of skill in the art of computer and application security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems identify the runtime risk of an

application program that executes on a device to the device, a computing system including the device, or another device connected to the computing system, and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how other solutions in this field identified the runtime risk of an application program that executes on a device to the device, a computing system including the device, or another device connected to the computing system based upon observed actions. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of a mere design choice from a finite number and order of well-known user actions, application actions, and system actions to yield predictable results because doing so would simply involve applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’517 patent, to the extent the references charted in any of the charts in Exhibit D do not expressly disclose “a runtime monitor including a processing device”, or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Satish which discloses a runtime monitor including a processing device (e.g. a security module), as well as Dalcher, VIDES, ThunderBYTE, Cuckoo Sandbox, Bettini, Nicodemus, Yeh, Chien, Alperovitch, Bose, and/or OfficeScan. A person of skill in the art would be motivated to combine these references because they each disclose assessing the runtime risk of an application executing on a computer through the application of rules and policies

in the same general field. It is the nature of those of skill in the art of computer and application security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems identify the runtime risk of an application program that executes on a device to the device, a computing system including the device, or another device connected to the computing system, and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how other solutions in this field identified the runtime risk of an application program that executes on a device to the device, a computing system including the device, or another device connected to the computing system. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’517 patent, to the extent the references charted in any of the charts in Exhibit D do not expressly disclose “identifying, using at least one assessment policy, a runtime risk for an application program that executes on a device, wherein the identified runtime risk indicates a risk or threat of the identified action sequence of the application”, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Satish, Dalcher, ThunderBYTE, Alperovitch, Bettini, Bose, Chien, OfficeScan, and/or Yeh. A person of skill in the art would be motivated to combine these references because they each disclose

assessing the runtime risk of an application executing on a computer through the application of rules and policies in the same general field. It is the nature of those of skill in the art of computer and application security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems identify the runtime risk of an application program that executes on a device to the device, a computing system including the device, or another device connected to the computing system, and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how other solutions in this field identified the runtime risk of an application program that executes on a device, and where the identified runtime risk indicated a risk or threat of the identified action sequence of the application executing on the device. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

Consistent with the exemplary motivations to combine each of the references discussed above, the below table provides a summary of the exemplary combinations of references rendering obvious to a POSA, with or without a POSA's knowledge (including the AAPA and general state of the art), skill and/or creativity, the Asserted Claims of the '517 Patent.

Reference	**Reference Alone or in Combination With One or More of the Following**
Dalcher	VIDES, Thunderbyte AV, Cuckoo Sandbox, Satish, Bettini, Nicodemus, Yeh, Chien, Alperovitch, Bose, OfficeScan
VIDES	Dalcher, Thunderbyte AV, Cuckoo Sandbox, Satish, Bettini, Nicodemus, Yeh, Chien, Alperovitch, Bose, OfficeScan
Thunderbyte AV	Dalcher, VIDES, Cuckoo Sandbox, Satish, Bettini, Nicodemus, Yeh, Chien, Alperovitch, Bose, OfficeScan

Reference	Reference Alone or in Combination With One or More of the Following
Cuckoo Sandbox	Dalcher, VIDES, Thunderbyte AV, Satish, Bettini, Nicodemus, Yeh, Chien, Alperovitch, Bose, OfficeScan
Satish	Dalcher, VIDES, Thunderbyte AV, Cuckoo Sandbox, Bettini, Nicodemus, Yeh, Chien, Alperovitch, Bose, OfficeScan
Bettini	Dalcher, VIDES, Thunderbyte AV, Cuckoo Sandbox, Satish, Nicodemus, Yeh, Chien, Alperovitch, Bose, OfficeScan
Nicodemus	Dalcher, VIDES, Thunderbyte AV, Cuckoo Sandbox, Satish, Bettini, Yeh, Chien, Alperovitch, Bose, OfficeScan
Yeh	Dalcher, VIDES, Thunderbyte AV, Cuckoo Sandbox, Satish, Bettini, Nicodemus, Chien, Alperovitch, Bose, OfficeScan
Chien	Dalcher, VIDES, Thunderbyte AV, Cuckoo Sandbox, Satish, Bettini, Nicodemus, Yeh, Alperovitch, Bose, OfficeScan
Alperovitch	Dalcher, VIDES, Thunderbyte AV, Cuckoo Sandbox, Satish, Bettini, Nicodemus, Yeh, Chien, Bose, OfficeScan
Bose	Dalcher, VIDES, Thunderbyte AV, Cuckoo Sandbox, Satish, Bettini, Nicodemus, Yeh, Chien, Alperovitch, OfficeScan
OfficeScan	Dalcher, VIDES, Thunderbyte AV, Cuckoo Sandbox, Satish, Bettini, Nicodemus, Yeh, Chien, Alperovitch, Bose

Additionally, Defendant incorporates by reference each of the obviousness combinations and motivations to combine presented in the petition for the following *inter partes* review proceeding(s) (IPRs) and any additional petitions for IPRs that may be filed challenging the patentability of the ’517 patent:

- IPR2023-00704
- IPR2023-00798

5. ’038 Patent & ’918 Patent

To the extent that Plaintiff contends that any of the references identified as anticipatory references fail to disclose one or more elements, those anticipatory references render the Asserted Claims obvious, either alone or in combination with other references. Defendant contends that one of skill in the art, at the time the alleged inventions were made, would have been motivated to

combine the references disclosed herein in such a way as to reach the alleged inventions, as described in further detail below. Defendant presently believes a reasonable basis exists that each of the claims asserted against it would have been obvious to one of ordinary skill in the art at the time of the alleged invention.

Below, Defendant identifies their current bases for obviousness. As seen below, Defendant presently asserts certain combinations of prior art reference(s) with or without the knowledge of one of ordinary skill in the art. Defendant offers these combinations without conceding that the reference(s) standing alone are insufficient to invalidate the relevant claims. Defendant specifically reserves their rights to combine any of its anticipatory references or any combinations of references with the knowledge of ordinary skill in the art to rebut a contention of Plaintiff that such reference(s) standing alone do not disclose every element of the Asserted Claims.

Defendant lists below prior art references and combinations of prior art references now known to Defendant, which Defendant contends render at least obvious the ’038 Asserted Claims and ’918 Asserted Claims. The corresponding charts in Exhibits E, H and I for each prior art reference identifies where and how each alleged item of prior art meets one or more limitations of the Asserted Claims. Pursuant to Patent Local Rule 3-6, Defendant expressly reserves the right to supplement these charts to provide additional details about obviousness should the Court’s claim construction require, or should Plaintiff serve adequate Infringement Contentions that better serve to notify Defendant which elements of their products allegedly correspond with the claim elements. Defendant’s contention that the references in this section in various combinations render obvious the ’038 Asserted Claims and ’918 Asserted Claims under 35 U.S.C. § 103 is in no way an admission or suggestion that each reference does not independently anticipate or render obvious the ’038 Asserted Claims and ’918 Asserted Claims under 35 U.S.C. § 102. These combinations would have been obvious to try given that there were a finite number of identified, predictable

solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field.

It would been obvious to combine each of the references discussed below because they are analogous art generally directed to methods and systems for controlling access to computing resources based on known computing security vulnerabilities. The technology described in the ’038 and ’918 Patents was widely known before the alleged priority date of these patents (December 21, 2005) and provided numerous motivations to combine the prior art references and systems disclosed herein. The common specification of the ’038 and ’918 Patents states that it addresses security concerns for electronic networks that allow, for example, employees at work or away from work to access “office files, design documents, employee work products, company information, and most other important business information [] being created and stored electronically on desktop computers, laptop computers, handheld computing devices (collectively ‘personal computing device’ or ‘computing device’) and company networks.” ’038 Patent, 1:30-47. The common specification of the ’038 and ’918 Patents also states that it strives to prevent “[i]nappropriate and/or unauthorized access to” electronic networks and computing resources by employing flexible endpoint access control. ’038 Patent, 1:47-55._

As admitted in the ’038 and ’918 AAPA (*see* Section III.C above and incorporated by reference herein), the field of endpoint access control followed an evolutionary path in the prior art, starting with “operating system services that controlled user access to one or more system resources” and “permitted or denied access to these resources based on a variety of factors, such as their login ID (which was authenticated using a secret) and a secured profile of policy settings identifying permissions and/or restrictions.” ’038 Patent, 1:63-2:3. The common specification of the ’038 and ’918 Patents also acknowledges that these initial approaches to endpoint access control were followed by “point solutions,” such as personal firewalls, antivirus agents, software update agents, restricting access to specific resources, etc., and then by “context awareness,” such

as restricting access if a named application is not running or is not of a specified minimum version or if a user is in a certain location, etc. ’038 Patent, 2:9-30, 2:42-61. The common specification of the ’038 and ’918 Patents identifies that the problems it intends to solve are that, as of the alleged priority date of these patents (December 21, 2005), the conventional access control solutions did not (1) provide a higher-level, holistic view of overall endpoint state, (2) collect and/or integrate endpoint state information from multiple sources, (3) revisit the decision to grant particular to a user’s endpoint over the course of the user’s connection or access session, (4) permit human administrators to define, via configuration settings, parameter values for different users and user groups, and (5) have the capability to optionally and selectively notify an end user when access control restrictions are being enforced on their endpoint. ’038 Patent, 2:23-41, 2:61-67, 3:1-26.

The cited art demonstrates that, at the time of the alleged priority date of the ’038 and ’918 Patents (December 21, 2005), methods and systems for controlling access to computing resources based on known computing security vulnerabilities were well known to solve these problems in the same manner disclosed in the common specification of these patents. The ’038 and ’918 AAPA also expressly acknowledges that its purported solution to solve these problems does not require any specialized technology, and that “[t]here are a number of alternative implementations for how the invention can be instantiated and operated” and, while several “examples of implementation methods” are disclosed, “[t]his list is exemplary and not exhaustive” and “others will now be apparent to the reader.” *See, e.g.*, ’038 Patent at 54:22-26, 7:52–64 (“endpoint system” can be “any processing system” made up of “conventional components”), 1:30-43 (network connections “are ubiquitous in the industry and well known to the reader”), 7:24-29 (using, “in a conventional manner, a processor” and “conventional storage components”), 8:4-13 (same), 9:6-9 (using “the accepted definition in the art” for software agents), 59:6–7 (“conventional data communications network”), 21:49–51 (listing “Web sites” that “[m]any computer hardware and software vendors are known to maintain”), 7:42-46 (using known “mobile communications device” and servers),

8:4-8 ("server computer" is "conventional processing system"), 1:63-65 (admitting "first generation of endpoint access control included operating system services"), 2:15-16 (exemplary known software agents include "antivirus agents[] [and] anti-spyware agents"). Accordingly, the common specification of the '038 and '918 Patents admits that it combines these conventional components according to their known, predictable and intended uses, just as they were and/or would have been combined by a POSA in the prior art. None of the combinations presents an unexpected result, a counterintuitive combination, or a technological challenge; and no such description exists in the common specification of the '038 and '918 Patents.

Likewise, the prior art cited herein would have been readily combined according to methods known to those of ordinary skill within the field of controlling access to computing resources based on known computing security vulnerabilities. The substitution of one element in this system could have been predictably achieved by one of ordinary skill at the time of the alleged invention. One of ordinary skill in the art would have been able to select appropriate attributes of one for inclusion in another. Those of ordinary skill in the art could have employed known techniques to improve similar prior art devices in the same way as claimed in the '038 and '918 Patents. Thus, the combinations all have well defined, predictable, expectation of success.

- For the Asserted Claims of the '038 and '918 patents, to the extent the references charted in any of the charts in Exhibits E, H and I do not expressly disclose "providing a user interface, at a computing system remote from the end point, configured to allow configuration of a plurality of policies," "providing a user interface, at a computing system remote from the end point, configured to allow configuration of a plurality of policies," including a "web page" user interface, and similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A,

including, for example, Cheng, Chickering, Farmer, Freund, Hermann I, Hermann II (and the incorporated Freund II), Sobel, Corley, Couillard, Kuo, Bluestone, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief. For example, Wright discloses a user interface (provided by a user interface module) at a computing system (i.e., the server system) remote from the endpoint mobile devices, which is configured to allow configuration of a plurality of policies (via policy management module 236). Likewise, Chickering discloses a user interface that an administrator can use to configure a plurality of policies (via controller 8), and that the user interface is at a computing system remote from the endpoint devices. Similarly, Freund discloses a user interface, including as a web page, on the computing system (router and sandbox server) that is remote from the endpoint computers and which allows the configuration of security policies via the administration interface. Herrmann II (and the incorporated Freund II) disclose a computing system (various servers) remote from endpoints, including an interface for configuring "centrally-maintained access rules," (i.e., to allow configuration of a plurality of policies). Farmer discloses a computing system (server) remote from the endpoints (hosts), including a user interface, including a "web-based interface", from which policies can be created and "configured to implement any type of policy rule or related enforcement action." Check Point R60 discloses computing systems (IASC, RAVS) remote from endpoint devices that include a web-based user interface (e.g., the Administration Console or SmartDashboard) from which policies can be created and configured (e.g., in the Policy Studio or SCVEditor). Corley teaches a

network that connects client computational device 716 (i.e., endpoint) with the security compliance server (i.e., a computing system having a user interface) placed remotely from each other. The security compliance server is integrated with an application that assists in the administrative management to provide centralized control for supporting administrative actions with respect to physical security operations and computational security operations. Further, the management application contains operator interface module 704 supports a user interface for an administrative user. Corley also references the user interface adapter 148 which connects various user devices within a typical computer architecture of a data processing system. Couillard teaches the CAS (Compliance Access Server) contains data storage in the form of Random Access Memory as well as an administrative management interface which allows configuration of compliance rules and policies within the computer system. Cheng teaches a web-based application, accessible through any browser, to access an access manager to configure policy rule sets for a client machine. A person of skill in the art would be motivated to combine these references because they each disclose methods and systems for implementing security policies at a system remote from an endpoint and were thus in the same general field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill in the art would have been motivated to specifically provide a user interface for the configuration of policies, to avoid waste of resources associated with the inability to easily configure policies. For example, it would have been obvious to a person of skill to look at how other solutions in this field accomplished user interfaces for configuring policies. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these

combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results. A person of ordinary skill would further have been motivated because a user interface can simplify user interactions relative to command line interfaces. A person of ordinary skill would further have been motivated because web page-based interfaces allow an interface to be served over the internet through a browser.

- For the Asserted Claims of the ’038 and ’918 patents, to the extent the references charted in any of the charts in Exhibits E, H and I do not expressly disclose “identifying, from the plurality of policies, a plurality of operating conditions on the endpoint to monitor,” and similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, _Cheng, Chickering, Farmer, Freund, Hermann I, Hermann II (and the incorporated Freund II), Sobel, Corley, Couillard, Kuo, Bluestone, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief.. For example, CyberGatekeeper discloses identifying, from the plurality of policies, a plurality of operating conditions on the endpoint to evaluate (e.g., conditions specified in the policies such as running process/modules, registry settings, file attributes, configuration file settings, OS version, registry key, antivirus .dat files). Likewise, Wright discloses that “aspects of a policy include[] . . . defining or identifying security features to be monitored, ports to be

monitored, network services to be monitored, [and] applications to be monitored" (i.e., identifying operating conditions on the endpoint to monitor from the plurality of policies). Similarly, Herrmann identifies, from the plurality of policies, a plurality of operating conditions to monitor, including the engine versions of anti-virus measures, data file versions, registry settings, file installations, currently executing processes, a particular checksum, and whether a file integrity policy is in force. Chickering discloses operating conditions to monitor on the endpoint (i.e., "health indicators") which are identified from the plurality of policies. Herrmann II (and the incorporated Freund II) disclose the "configuration of rules which govern use of the protocols monitored by the system," including monitoring currently active processes, including checking executable name, version numbers, executable file checksums, version header details, configuration settings, etc. Check Point R60 discloses identifying from the plurality of policies various operating conditions on the endpoint to monitor, including Integrity Enforcement Rules (e.g., registry keys and registry values, existence of files, files properties such as version numbers and that certain program be running or not on the endpoint to monitor), as well as SCV checks, each of which comprise an operating condition on the endpoint to monitor and which are maintained in a single file. Corley teaches a network that connects client computational device 716 (i.e., endpoint) with the security compliance server (i.e., a computing system) placed remotely from each other. The security compliance server 702 accesses policy database 712 which contains various types of policies that are configurable to control the operation of multiple aspects of the components in a data processing system. Further, the security compliance server 702 configures the client 716 by sending network security monitoring agent 718 a copy of filtering policy 720 which contains rules and/or conditions that need to be checked against the network-related activities of the client 716. Couillard discloses the installation and configuration of an agent software on the endpoint device to

provide a list of compliance rules to verify any number of operating conditions on the endpoint device. Kuo teaches unified security management for endpoint protection that comprises software security sub 101 (i.e., software agents) which includes unified agent 109 that includes a function like monitoring host activities (i.e., operating conditions). The sub-agent of unified agent 109 uses a data collector to periodically examine the host's configuration database, software version, patch signature information, and security profile data and transmit it to the Unified Agent Manager 601 of the management server 103 (i.e., a computing system). Cheng teaches assessing the posture of a client machine based on policy rule sets by client agent software on the client machine. A person of skill in the art would be motivated to combine these references because they each disclose methods and systems for monitoring operating conditions and were thus in the same general field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill in the art would have been motivated to specifically monitor the operating conditions specified in the policy, to avoid waste of resources monitoring operating conditions that were irrelevant to policies. For example, it would have been obvious to a person of skill to look at how other solutions in this field accomplished monitoring operating conditions, such as through policies. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’038 and ’918 patents, to the extent the references charted in any of the charts in Exhibits E, H and I do not expressly disclose “configuring one or

more software agents on the endpoint to monitor the plurality of operating conditions," and similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Cheng, Chickering, Farmer, Freund, Hermann I, Hermann II (and the incorporated Freund II), Sobel, Corley, Couillard, Kuo, Bluestone, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief. For example CyberGatekeeper, discloses configuring one or more software services provided by an operating system on the endpoint (*e.g.*, CyberGatekeeper agent, services provided by the operating system that provide details to the agent) to monitor the plurality of operating conditions. Likewise, Wright discloses configuring software agents (e.g., location detection module 208, security features determination module 210, client diagnostics module 246, policy rule engine 704, driver 706, file filter 824, TDI filter 822, and/or packet filter engine 818) to monitor the plurality of operating conditions, which are distributed via distribution module 234. Herrmann discloses software agents such as client security module 355 to monitor the plurality of operating conditions, including by using a "client monitoring protocol (CMP)" and through the use of heartbeat challenges. Herrmann II (and the incorporated Freund II) disclose software agents on the endpoint (*i.e.*, anti-virus plug-in 326, TrueVector service 320, and client module 324, and client data monitor 311/data acquisitions module 440) to monitor operating conditions. Farmer discloses configuring software agents on the

endpoint (i.e., data gathering engine 222 and policy engine 220 on host 110) to monitor the plurality of operating conditions (e.g., "data gathering engine [] is configured to collect host data related to the host."). Check Point R60 discloses Integrity Clients and/or RAVS (software agents) on the endpoint, which are configured to monitor operating conditions set out in the Integrity Policy and the SCV policy. Corley teaches a data processing system that comprises client computational device 716 (i.e., endpoint) which comprises network security monitoring agent 718 that consists a copy of filtering policy 720 which contains rules and/or conditions that need to be checked against the network-related activities of client 716. Couillard discloses the installation and configuration of an agent software on the endpoint device to provide a list of compliance rules to verify any number of operating conditions on the endpoint device. Kuo teaches unified security management for endpoint protection that comprises software security sub 101 which includes unified agent 109 that includes a function like monitoring host activities. The sub-agent of unified agent 109 uses a data collector to periodically examine the host's configuration database, software version, patch signature information, and security profile data. Cheng teaches client agent software that is configured on a client machine to apply policy rule sets on the client machine. A person of skill in the art would be motivated to combine these references because they each disclose methods and systems for monitoring operating conditions and were thus in the same general field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill in the art would have been motivated to specifically monitor the operating conditions specified in the policy, to avoid unnecessary waste of resources monitoring operating conditions that were irrelevant to policies. For example, it would have been obvious to a person of skill to look at how other solutions in this field accomplished monitoring operating conditions, such as through policies. These combinations would

have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results. A person of skill in the art would further have been motivated to monitor the operating conditions in the policy at the endpoint to avoid waste of resources monitoring operating conditions that were irrelevant to policies, and because the endpoint is where the conditions occur.

- For the Asserted Claims of the ’038 and ’918 patents, to the extent the references charted in any of the charts in Exhibits E, H and I do not expressly disclose “ configuring one or more software services provided by an operating system on the endpoint to monitor the plurality of operating conditions,” “one or more software services provided by an operating system on the endpoint configured to evaluate a plurality of operating conditions identified in the plurality of policies,” and similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Cheng, Chickering, Farmer, Freund, Hermann I, Hermann II (and the incorporated Freund II), Sobel, Corley, Couillard, Kuo, Bluestone, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief.. For example, Elemental Security, discloses configuring one or more software

services provided by an operating system on the endpoint to monitor the plurality of operating conditions (e.g. monitoring the endpoints). CyberGatekeeper, discloses configuring one or more software services provided by an operating system on the endpoint (*e.g.*, CyberGatekeeper agent, services provided by the operating system that provide details to the agent) to monitor the plurality of operating conditions. Likewise, Wright discloses configuring software services provided by an operating system on the endpoint (*i.e.*, operating system services are configured by location detection module 208, security features determination module 210, client diagnostics module 246, management tools 816, application / transport driver interface (TDI) filter 822, packet filter engine 818, NDIS intermediate driver 834, file filter 824, file system 826, and/or windows socket 828), including, for example, windows socket filter 828 (Winsock), which can be configured to monitor and provide which applications are accessing a network and the networking services the applications will be using. Herrmann II (and the incorporated Freund II) disclose that anti-virus plug-in 326 may read Windows registries to determine where information about anti-virus applications can be found, and Herrmann II discloses specific code implementations to read those registries in Windows NT, Windows 95, and Windows 98, as well as use of the Windows driver "Winsock", while Freund II discloses client monitor 311 is "preferably implemented as a Windows VxD driver," and which comprises various operating system services, including: "Winsock Hook 501 includes functionality for connecting or 'hooking' into the Winsock communication driver. File Hook 503, in a similar manner, includes functionality allowing the driver 440 to hook into the file subsystem provided by the underlying operating system. Process Hook 505, another subcomponent which hooks into the underlying operating system, tracks all currently-executing applications or processes. All of these hooks, as implemented in a Windows VxD, are capable of executing at ring 0—that is, execution at the highest privileged level

of the operating system." Farmer discloses the use of a data gathering engine use software services provided by the operating system on the endpoint to monitor to "host configuration information, such as host operating system information and installed hardware and software information, file system information, such as file names and sizes, and information about currently running applications," and to "detect[] a version of a program (using meta-data, MD5 signatures and the like)". Check Point R60 discloses Integrity Client and RAVS which are comprised of several Windows drivers and Windows services, provided by the operating system, and which are used to monitor operating conditions. Corley teaches a data processing system that comprises client computational device 716 (i.e., endpoint) which comprises network security monitoring agent 718 that consists a copy of filtering policy 720 which contains rules and/or conditions that need to be checked against the network-related activities of client 716. Couillard discloses the installation and configuration of software services on the endpoint device to provide a list of compliance rules to verify any number of operating conditions on the endpoint device. Kuo teaches unified security management for endpoint protection that comprises software security sub 101 which includes unified agent 109 that includes a function like monitoring host activities. The sub-agent of unified agent 109 uses a data collector to periodically examine the host's configuration database, software version, patch signature information, and security profile data. Cheng teaches client agent software that is configured on a client machine to apply policy rule sets on the client machine. A person of skill in the art would be motivated to combine these references because they each disclose methods and systems for monitoring operating conditions and were thus in the same general field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill in the art would have been motivated to specifically monitor the operating conditions specified in

the policy, to avoid unnecessary waste of resources monitoring operating conditions that were irrelevant to policies. A person of ordinary skill in the art would have also found it obvious, and been motivated, to configure software agents or software services provided by an operating system on the endpoint to monitor operating conditions that determine policy compliance because the operating conditions of the endpoint have to be monitored to know if the policy that depends on those operating conditions is being complied with. Furthermore, it was known that software services provided by an operating system could be used to monitor endpoint conditions. For example, it would have been obvious to a person of skill to look at how other solutions in this field accomplished monitoring operating conditions, such as through policies. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results. A person of ordinary skill in the art would also have been motivated to integrate the monitoring or evaluation functionality with the operating system to provide an integrated product.

- For the Asserted Claims of the ’038 and ’918 patents, to the extent the references charted in any of the charts in Exhibits E, H and I do not expressly disclose “receiving, across a network, at the computing system, status information about the plurality of operating conditions on the endpoint gathered by the one or more software services on the endpoint, and user information that identifies a user of the endpoint,” and similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references

identified in Section III.A, including, for example, Cheng, Chickering, Farmer, Freund, Hermann I, Hermann II (and the incorporated Freund II), Sobel, Corley, Couillard, Kuo, Bluestone, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief.. For example, Wright discloses receiving status information about monitored operating conditions (comprising location indicators, activity statuses of one or more security features, network parameters, and the like may be received at monitoring module 226, policy setting module 238, policy enforcement module 244, and/or authorization module 232), as well as user information that identifies a user (e.g., authorization module 232 receives a username and password and user information). Chickering discloses that the computing system receives status information (e.g., whether a particular application is currently executing or whether particular software is installed) and user information identifying a user (e.g., a username and a password, a biometric, or a combination thereof that identifies a user 18 individually or by role). Herrmann discloses receiving "heartbeat" challenges from client security module 255, including status information about operating conditions, and discloses receiving user information such as a username and password at ADS 480 for authentication. Farmer discloses receiving "host data" (including, e.g., "user information, such as password and user name information, network information . . . host configuration information . . . file system information . . . and information about currently running applications") at the central server from the various endpoints (hosts). Check Point R60 discloses receiving at the Integrity server and the gateway, respectively, logs with status information about the plurality of operating

conditions on the endpoint gathered by the one or more software services, and SCV status information, and that in both cases the well-known IKE protocol is used to authenticate users, which includes the receipt of user information identifying users of endpoints. Corley teaches a data processing system that comprises client computational device 716 (i.e., endpoint) which comprises network security monitoring agent 718 that consists a copy of filtering policy 720 which contains rules and/or conditions that need to be checked against the network-related activities of client 716. The network security monitoring agent 718 (i.e., software agents) performs checking and records the potentially problematic activity within the cached activity log 722 (i.e., status information) and periodically sent it to a dynamic security level adjustment module 742 of the security compliance server 740 (i.e., a computing system). Couillard teaches that in the CAS, which is installed on the endpoint device, a compliance decision maker receives information about any number of operating conditions on the endpoint device received by the Compliance Security Agent (CSA) in the computing system across a network. Kuo teaches unified security management for endpoint protection that comprises software security sub 101 (i.e., software agents) which includes unified agent 109 that includes a function like monitoring host activities (i.e., operating conditions). The sub-agent of unified agent 109 uses a data collector to periodically examine the host's configuration database, software version, patch signature information, and security profile data and transmit it to the Unified Agent Manager 601 of the management server 103 (i.e., a computing system). Cheng teaches a client agent on a client machine that performs checks on the client machine, as defined by elements of policy rule sets, with the results reported to the access manager. A person of skill in the art would be motivated to combine these references because they each disclose methods and systems for monitoring operating conditions and were thus in the same general field. It is the nature of those of skill in the art of computer security to monitor developments in competing

products and combine relevant features and functionalities. A person of skill in the art would have been motivated to specifically monitor the operating conditions at the endpoint and send status information about the conditions to the computing system, to facilitate an updated view of the operating conditions at the computing system. For example, it would have been obvious to a person of skill to look at how other solutions in this field accomplished monitoring operating conditions. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results. A person of ordinary skill would also have been motivated because user information was one of various criteria that could be used to set permissions.

- For the Asserted Claims of the ’038 and ’918 patents, to the extent the references charted in any of the charts in Exhibits E, H and I do not expressly disclose “determining, by the computing system, a compliance state of the endpoint based on [the user information and] the status information and a plurality of compliance policies in the data store,” and similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Cheng, Chickering, Farmer, Freund, Hermann I, Hermann II (and the incorporated Freund II), Sobel, Corley, Couillard, Kuo, Bluestone, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network

Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief.. For example, CyberGatekeeper discloses determining, by the computing system, a compliance state of the endpoint based on the user information and status information, and a plurality of compliance policies in the data store. Similarly, Wright discloses determining compliance (e.g., by policy enforcement module 244) based on status information and policies (e.g., whether specified applications are running, specified protocols are enabled, a specified VPN is running, per the compliance policies). Likewise, Herrmann, "determin[es] whether the client is in compliance with [a] policy based, at least in part, on attributes of the client" by reference to the plurality of policies on the policy server, and further discloses that user information is used to determine the particular policy against which attributes should be checked. Chickering discloses determining a compliance state based on user information and status information, compared against the health policies in the health policy database, and/or in protection device policy table, including by first determining compliance based on user information and then by comparing status information against policies applicable to that user. Check Point R60 discloses generating reports including Enforcement Details reports, which require determination of the compliance state of endpoints, and the process for which involves comparing user and status information (e.g. SCV status) against policies for a particular user. Corley teaches a security compliance server 740 (i.e., a server within the computing system) that comprises dynamic security level adjustment module 742 which receives the record logs of the potentially problematic activity of client computing device 716 (i.e., endpoint). Couillard teaches verification of compliance for the endpoint device for each applicable rule or policy based on the Compliancy Appliance Server's analysis of status information on the endpoint device using a set of Corporate Compliancy

Rules (CCR) provided by the Compliancy Security Agent (CSA). Kuo teaches unified security management for endpoint protection that comprises software security sub 101 (i.e., software agents) which includes unified agent 109 that includes a function like monitoring host activities (i.e., operating conditions). The sub-agent of unified agent 109 uses a data collector to periodically examine the host's configuration database, software version, patch signature information, and security profile data and transmit it to the Unified Agent Manager 601 of the management server 103 (i.e., a computing system). Cheng teaches an access manager that determines whether there is adequate protection on a client machine based on the results of checks, as defined by elements of policy rule sets in a policy repository, by a client agent. A person of skill in the art would be motivated to combine these references because they each disclose determining compliance states of computing systems and thus are in the same general field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and make determinations about compliance state by the computing system, as opposed to the endpoint and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how other solutions in this field determine compliance. A person of ordinary skill would have been motivated to determine compliance on the computing system to reduce the load on the endpoint and it was well known that tasks could be divided between client and server. Further, a person of skill in the art would have been motivated to make compliance determinations on the computing system, as opposed to the endpoint, as a way to avoid having compliance determinations made by a potentially compromised endpoint, and because it would support compliance determinations to be made at a centralized location for potentially many endpoints with attendant simplification in the administration of the compliance determination. These

combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results. A person of ordinary skill would also have been motivated because user information was one of various criteria that could be used to set permissions.

- For the Asserted Claims of the ’038 and ’918 patents, to the extent the references charted in any of the charts in Exhibits E, H and I do not expressly disclose “initiating, by the computing system, based on the compliance state, an action identified in at least one rule in the data store, wherein the action is carried out by a processor on the endpoint,” “such that the computing system remotely ensures endpoint compliance with the plurality of compliance policies in the data store of the computing system,” and similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Cheng, Chickering, Farmer, Freund, Hermann I, Hermann II (and the incorporated Freund II), Sobel, Corley, Couillard, Kuo, Bluestone, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief. For example, Wright discloses, based on the compliance state, initiating an action identified by a rule in the data store (e.g., policy enforcement module

sends instructions to processor on the endpoint to perform an action from the rules, including, e.g., locking the endpoint, blocking access to a file, or blocking access to an application). Similarly, Chickering discloses, based on the compliance state (e.g., if an endpoint is misconfigured and thus not in compliance), initiating an action in a rule from the data store (e.g., denying access to a server per a protection policy), wherein the endpoint carries out the action, such as by enforcing a set of access rights at the endpoint firewall that were obtained from the server. Herrmann II (and the incorporated Freund II) disclose initiating at the computing system, based on the compliance state an action identified in a rule at the data store (e.g., in the policies in policy store 371), to be carried out by the processor at the endpoint, such as where the integrity server sends permission to the endpoint to access protected resources, subsequent to which the endpoint accesses the resource, or, by contrast, where the sandbox server initiates an action by indicating noncompliance to the endpoint, which then performs the action of displaying a message indicating noncompliance and lack of access. Check Point R60 discloses that once a protected computer is out of compliance for a certain number of heartbeats, the client executes whatever action is specified in an enforcement rule, such as observing (logging endpoint use), warning (logging and sending a warning to the endpoint), or restricting (restricting the endpoint according to whatever restriction rules are configured) computers that are out of compliance. Corley teaches a security compliance server 740 (i.e., a computing system) that comprises dynamic security level adjustment module 742 which receives the record logs of the potentially problematic activity of client computing device 716 (i.e., endpoint). Further, analysis of record logs is performed in accordance with various configurable conditions and/or rules (i.e., compliance policies). Each of the various implemented policies – which are confirmed by the compliance procedures in Corley – include certain actions/rules/conditions which are implemented on the various user devices

and which can be different for each individual user and/or device. Couillard teaches the transmission of a compliance state message form the Compliancy Server Agent to the Compliancy Appliance Server which initiates an action whereby the results of compliancy analysis are presented to a user of the device through a local web page, or the device can be deemed non-compliant and be blocked from logging on to the computer system. Kuo teaches unified security management for endpoint protection that comprises software security sub 101 (i.e., software agents) which includes unified agent 109 that includes a function like monitoring host activities (i.e., operating conditions). The sub-agent of unified agent 109 uses a data collector to periodically transmit the user client device information to the Unified Agent Manager 601 of the management server 103 (i.e., a computing system). The Unified Agent Manager 601 determines by using function whether the received information about the endpoint is out of compliance or not (i.e., compliance state). If an out of compliance event occurred, the Unified Agent Manager 601 coordinates automatically with sub-agent 523 of the endpoint to download an update or take other proper measures (i.e., action) through Action Enforcer 530 (i.e., processor) of the endpoint device. Cheng teaches that e.g., remediation action, such as implementing up-to-date virus checking options, as identified by a rule requiring the latest antivirus version be installed on the client machine, may be taken on the client machine based on the compliance state. A person of skill in the art would be motivated to combine these references because they each disclose methods and systems for monitoring operating conditions and enforcing policy compliance and were thus in the same general field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill in the art would have been motivated to specifically monitor the operating conditions specified in the policy and enforce compliance to ensure network security. For example, it would have been obvious

to a person of skill to look at how other solutions in this field accomplished monitoring operating conditions and enforcing compliance, such as through policies. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results. A person of ordinary skill would also have been motivated because the computing system could operate as a centralized location to initiate remedial actions on a potentially compromised endpoint. A person of ordinary skill would also have been motivated because if an endpoint were to be out of compliance with a policy, it would be desirable to enforce compliance with the policy.

- For the Asserted Claims of the ’038 and ’918 patents, to the extent the references charted in any of the charts in Exhibits E, H and I do not expressly disclose “authorizing access by the endpoint to a computing resource on the network, authorization being determined by the remote computing system in response to the compliance state,” and similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Cheng, Chickering, Farmer, Freund, Hermann I, Hermann II (and the incorporated Freund II), Sobel, Corley, Couillard, Kuo, Bluestone, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass

Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief. For example, Wright discloses, based on the compliance state, authorizing access to a computing resource on a network (e.g., policy enforcement module sends instructions to processor on the endpoint to, e.g., lock the endpoint, block access to a file, or block access to an application). Similarly, Chickering discloses, based on the compliance state (e.g., if an endpoint is misconfigured and thus not in compliance), authorizing access to a computing resource on a network (e.g., denying access to a server), wherein the endpoint carries out the action, such as by enforcing a set of access rights at the endpoint firewall that were obtained from the server. Herrmann II (and the incorporated Freund II) disclose, based on the compliance state, authorizing access to a computing resource on a network, to be carried out by the processor at the endpoint, such as where the integrity server sends permission to the endpoint to access protected resources, subsequent to which the endpoint accesses the resource, or, by contrast, where the sandbox server initiates an action by indicating noncompliance to the endpoint, which then performs the action of displaying a message indicating noncompliance and lack of access. Freund discloses authorizing access of endpoint 320 based on its compliance state in compliance table 312, according to which router 310 may authorize access by endpoint 320 to the internet or sandbox server 320. Check Point R60 discloses that once a protected computer is out of compliance for a certain number of heartbeats, the client executes whatever action is specified in an enforcement rule, such as observing (logging endpoint use), warning (logging and sending a warning to the endpoint), or restricting (restricting the endpoint according to whatever restriction rules are configured) computers that are out of compliance, thereby either allowing or denying access to a computing resource on a network. Couillard teaches the transmission of a compliance state message form the Compliancy Server Agent to the Compliancy Appliance Server which initiates an action

whereby the results of compliancy analysis are presented to a user of the device through a local web page, (or the device can be deemed non-compliant and be prevented from logging on to the computer system). Kuo teaches unified security management for endpoint protection that comprises software security sub 101 (i.e., software agents) which includes unified agent 109 that includes a function like monitoring host activities (i.e., operating conditions). The sub-agent of unified agent 109 uses a data collector to periodically transmit the user client device information to the Unified Agent Manager 601 of the management server 103 (i.e., a computing system). The Unified Agent Manager 601 determines by using function whether the received information about the endpoint is out of compliance or not (i.e., compliance state). If an out of compliance event occurred, the Unified Agent Manager 601 coordinates automatically with sub-agent 523 of the endpoint to download an update or take other proper measures (i.e., action) through Action Enforcer 530 (i.e., processor) of the endpoint device. If no out of compliance event occurred then unified agent manager 601 coordinates with subagent 523 of endpoint to take proper measures such as allowing endpoint to perform its operations. Cheng teaches access of a client machine to specific applications and resources, such as confidential financial databases, are controlled based on authorization determined by the remote computing system in response to the compliance state. A person of skill in the art would be motivated to combine these references because they each disclose methods and systems for monitoring operating conditions and enforcing policy compliance and were thus in the same general field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill in the art would have been motivated to specifically monitor the operating conditions specified in the policy and enforce compliance to ensure network security. For example, it would have been obvious to a person of skill to look at how other solutions in

this field accomplished monitoring operating conditions and enforcing compliance, such as through policies. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results. A person of ordinary skill would also have been motivated because restricting or authorizing network access was a well-known concept, and because allowing an endpoint to access the network while not compliant with a security policy would pose a security risk.

- For the Asserted Claims of the ’038 and ’918 patents, to the extent the references charted in any of the charts in Exhibits E, H and I do not expressly disclose “continuing to monitor the compliance state by the endpoint and restricting access to the computing resource if the compliance state changes,” and similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Cheng, Chickering, Farmer, Freund, Hermann I, Hermann II (and the incorporated Freund II), Sobel, Corley, Couillard, Kuo, Bluestone, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief. For example, Wright discloses that compliance state is continually monitored (e.g.,

by location detection module 208 and security features determination module 210) to allow access to a computing resource (e.g., a file on network 204 which may comprise the internet), and that access may be restricted if the location changes and/or security features on endpoint 201 change. Herrmann discloses periodic requests by client security module to allow for checking or rechecking, including by the use of heartbeat challenges that result in continuous monitoring and compliance, and Herrmann discloses that if the compliance state changes, access may be restricted (e.g., if a Kerberos ticket allowing access expires upon a certain heartbeat and an endpoint is no longer in compliance, access is restricted). Chickering discloses continued monitoring of compliance state and restriction of access if compliance state changes, including by sending updated access rights to the firewall on the endpoint, restricting access if compliance state changes. Herrmann II (and the incorporated Freund II) disclose the user of heartbeat challenges/requests to client devices requiring endpoints to provide specified information, and that if these challenges are failed (indicating noncompliance), restricting access to protected resources. Check Point R60 discloses regular SCV checks (i.e., continued monitoring of compliance state) and that gateways can drop connections with endpoints if compliance state changes (i.e., restricting access to the computing resources). Corley teaches a data processing system that comprises client computational device 716 (*i.e.*, endpoint) which comprises network security monitoring agent 718 that consists a copy of filtering policy 720 which contains rules and/or conditions that need to be checked against the network-related activities of client 716. The network security monitoring agent 718 (*i.e.*, software agents) performs checking and records the potentially problematic activity within the cached activity log 722 (*i.e.*, status information) and periodically sent it to a dynamic security level adjustment module 742 of the security compliance server 740 (*i.e.*, a computing system). Couillard teaches the Compliancy Appliance Server (CAS) has the ability to constantly monitor the entire

computer system as well as endpoint devices attempting to access the system, to restrict access to computing resources in the event of non-compliance and to repeat the compliancy verification process after a device is deemed non-compliant until the device is deemed compliant. Kuo teaches unified security management for endpoint protection that comprises software security sub 101 (i.e., software agents) which includes unified agent 109 that includes a function like monitoring host activities (i.e., operating conditions). The sub-agent of unified agent 109 uses a data collector to periodically transmit the user client device information to the Unified Agent Manager 601 of the management server 103 (i.e., a computing system). The Unified Agent Manager 601 determines by using function whether the received information about the endpoint is out of compliance or not (i.e., compliance state). If an out of compliance event occurred, the Unified Agent Manager 601 coordinates automatically with sub-agent 523 of the endpoint to download an update or take other proper measures (i.e., action) through Action Enforcer 530 (i.e., processor) of the endpoint device. Cheng teaches an access server is continuing to monitor the compliance state of a client machine and restricting access to the computing resource if the compliance state changes by placing the client machine in a temporary role and performing remediation procedures. A person of skill in the art would be motivated to combine these references because they each disclose methods and systems for monitoring operating conditions and enforcing policy compliance and were thus in the same general field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill in the art would have been motivated to specifically and continually monitor the operating conditions specified in the policy and enforce compliance to ensure network security. For example, it would have been obvious to a person of skill to look at how other solutions in this field accomplished monitoring operating conditions and enforcing compliance, such

as through policies. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results. A person of ordinary skill in the art would also have been motivated because even if an endpoint satisfies a policy at a given time, the endpoint may subsequently fall out of compliance with the policy and pose a security risk.

- For the Asserted Claims of the ’038 and ’918 patents, to the extent the references charted in any of the charts in Exhibits E, H and I do not expressly disclose “requesting, at the computing system, the status information on a periodic basis,” and similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Cheng, Chickering, Farmer, Freund, Hermann I, Hermann II (and the incorporated Freund II), Sobel, Corley, Couillard, Kuo, Bluestone, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief. For example, Wright discloses the “periodic” requesting of status information by the computing system from the endpoint, and that the disclosed methods may run on a continuous basis by using a “periodic time setting or a loop”.

Hermann discloses that the computing system (i.e., security checker 359 on ADS 460) may request the status information on a periodic basis, and that "[t]he security checker module may also issue periodic "heartbeat" challenges every N seconds or minutes, as determined by the monitoring frequency setting established by the administrator." Freund discloses that the computing system may issue a request (i.e., issue a "router challenge") for the status information (i.e., information related the monitored conditions on the endpoint such as whether a software program is installed or currently running) on a periodic basis (i.e., every few seconds). Herrmann II (and the incorporated Freund II) disclose the use of periodic check messages and heartbeat requests to client devices (endpoints), requesting status information from the endpoint. Check Point R60 discloses a heartbeat mechanism, though which "[c]onnected computers are rechecked every heartbeat to ensure that they remain compliant", as well as that the Integrity Server configures the Integrity Client with the heartbeat interval, thus requesting status information on a periodic basis, and that gateways can request from clients SCV status messages (e.g., every 20 seconds for 5 minutes), requesting SCV status information from the endpoint. Couillard teaches the Compliancy Appliance Server (CAS) has the ability to constantly monitor the entire computer system as well as endpoint devices attempting to access the system and also teaches repetition of the compliancy verification process after a device is deemed non-compliant until the device is deemed compliant. Kuo teaches unified security management for endpoint protection that comprises software security sub 101 (i.e., software agents) which includes unified agent 109 that includes a function like monitoring host activities (i.e., operating conditions). The sub-agent of unified agent 109 uses a data collector to periodically examine the host's configuration database, software version, patch signature information, security profile data, user profile and transmit it to the Unified Agent Manager 601 of the management server 103 (i.e., a computing system). Cheng teaches e.g., a policy

server that regularly receives up-to-date information so that up-to-date status information can be requested on a periodic basis. A person of skill in the art would be motivated to combine these references because they each disclose methods and systems for monitoring operating conditions and enforcing policy compliance and were thus in the same general field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill in the art would have been motivated to specifically and continually monitor the operating conditions specified in the policy and enforce compliance to ensure network security. For example, it would have been obvious to a person of skill to look at how other solutions in this field accomplished monitoring operating conditions and enforcing compliance, such as through policies. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results. A person of ordinary skill in the art would also have been motivated because there are a limited number of options for requesting status information (e.g., at request, on a periodic basis). A person of ordinary skill in the art would also have been motivated because even if an endpoint satisfies a policy at a given time, the endpoint may subsequently fall out of compliance with the policy and pose a security risk.

- For the Asserted Claims of the ’038 and ’918 patents, to the extent the references charted in any of the charts in Exhibits E, H and I do not expressly disclose “software agents comprise at least one service provided by the operating system of the endpoint,” and similar limitations, it would have been obvious to combine the teachings of any of these charted

references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Cheng, Chickering, Farmer, Freund, Hermann I, Hermann II (and the incorporated Freund II), Sobel, Corley, Couillard, Kuo, Bluestone, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief. For example, Wright discloses that the software agents (i.e., location detection module 208, security features determination module 210, client diagnostics module 246, file filter 824, TDI filter 822, and/or packet filter engine 818) may comprise at least one service provided by the operating system of the endpoint (e.g., the policy engine 832 determines [] from the operating system of the mobile device which security software programs . . . are currently running on the mobile device."). Herrmann II (and the incorporated Freund II) disclose that anti-virus plug-in 326 may read Windows registries to determine where information about anti-virus applications can be found, and Herrmann II discloses specific code implementations to read those registries in Windows NT, Windows 95, and Windows 98, as well as use of the Windows driver "Winsock", while Freund II discloses client monitor 311 is "preferably implemented as a Windows VxD driver," and which comprises various operating system services, including: "Winsock Hook 501 includes functionality for connecting or 'hooking' into the Winsock communication driver. File Hook 503, in a similar manner, includes functionality allowing the driver 440 to hook into the file subsystem provided by the underlying operating system. Process Hook 505, another subcomponent which hooks into the

underlying operating system, tracks all currently-executing applications or processes. All of these hooks, as implemented in a Windows VxD, are capable of executing at ring 0—that is, execution at the highest privileged level of the operating system." Farmer discloses that software agents can be configured to monitor "host configuration information, such as host operating system information and installed hardware and software information, file system information, such as file names and sizes, and information about currently running applications," and to "detect[] a version of a program (using meta-data, MD5 signatures and the like". Check Point R60 discloses that the Integrity Client and RAVS each comprise at least one application running on the endpoint in the form of a Graphical User interface and that each also comprises a Windows service which is also an application running on the endpoint. Corley discloses client devices hosting software services (e.g., applications) on operating systems capable of implementing portions of the recited functionality. Couillard teaches that the Compliance Security Agent (CSA) is an automated service set to detect and host software services capable of running on the operating system of an endpoint device. Kuo teaches that the Unified Agent 109 (i.e., software agent) functions in Security Utility Unit ("SUU") 304 are paired with an operating system of the endpoint. Cheng teaches that client agent software is installed on a client machine and can perform services, such as registry, process, application and service checks, which are provided by the operating system of the client machine. A person of skill in the art would be motivated to combine these references because they each disclose methods and systems for monitoring operating conditions and were thus in the same general field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill in the art would have been motivated to monitor operating by leveraging services provided by the operating system at the endpoint, to allow access to lower-level information about the

endpoint. Furthermore, it was known that software services provided by an operating system could be used to monitor endpoint conditions. For example, it would have been obvious to a person of skill to look at how other solutions in this field accomplished monitoring operating conditions, such as by leveraging services provided by the operating system at the endpoint. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’038 and ’918 patents, to the extent the references charted in any of the charts in Exhibits E, H and I do not expressly disclose “conditions comprise at least one hardware condition,” and similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Cheng, Chickering, Farmer, Freund, Hermann I, Hermann II (and the incorporated Freund II), Sobel, Corley, Couillard, Kuo, Bluestone, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief. For example, Wright discloses that the monitored operating conditions may comprise the type of connection, wire or wireless, used by the endpoint, or the type of network

adapter card being used by the endpoints (i.e., hardware conditions). Chickering discloses that the monitored operating conditions may comprise source and destination MAC addresses for the endpoint's network communications (i.e., hardware conditions). Hermann discloses that the monitored operating conditions may comprise the type of client device or the location of the device (i.e., hardware conditions). Herrmann II (and the incorporated Freund II) disclose that the monitored operating conditions may comprise information about the client "machine" or "certain mouse, and other interactive user activities" (i.e., hardware conditions). Farmer discloses that the monitored operating conditions may comprise "network information" and "installed hardware and software information" (i.e., hardware conditions). Check Point R60 discloses that the monitored operating conditions may comprise "cputype," "cpufamily," or "cpumodel" (i.e., hardware conditions). Couillard teaches the detection of the presence or absence of a number of hardware states and conditions (e.g. any file state of the hard disk) as well as the compliance verification of a number of hardware states and conditions. Kuo teaches unified security management for endpoint protection that comprises sub 101 which includes unified agent 109 that includes a function like monitoring host activities. The sub-agent of unified agent 109 uses a data collector to periodically examine the host's configuration database, software version, patch signature information, and security profile data. Cheng teaches the detection of, e.g., a MAC address of a client machine, to determine network access for the client machine. A person of skill in the art would be motivated to combine these references because they each disclose methods and systems for monitoring operating conditions and were thus in the same general field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill in the art would have been motivated to monitor the operating conditions including hardware conditions, to provide greater network security

protection. For example, it would have been obvious to a person of skill to look at how other solutions in this field accomplished monitoring operating conditions, including hardware conditions. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’038 and ’918 patents, to the extent the references charted in any of the charts in Exhibits E, H and I do not expressly disclose “configuring one or more application running on the endpoint to monitor at least a subset of the plurality of operating conditions,” and similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Cheng, Chickering, Farmer, Freund, Hermann I, Hermann II (and the incorporated Freund II), Sobel, Corley, Couillard, Kuo, Bluestone, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief. Check Point R60, Elemental Security, iPass. For example, Wright discloses that an endpoint application (e.g., location detection module 208, security features determination module 210, client diagnostics module 246, file filter 824, TDI filter 822, and/or packet

filter engine 818) may be configured to monitor at least a subset of operating conditions. Similarly, Chickering discloses that an endpoint application (e.g., agent) may be configured to monitor at least a subset of operating conditions (i.e., by "generat[ing] health information that describes a security status of respective endpoint devices 14"). Likewise, Freund discloses that an endpoint application (e.g., client security module 321 installed and running on computers 320 and 340) may be configured to monitor at least a subset of operating conditions (e.g., via a "client monitoring protocol (CMP)" which "is a simple monitoring protocol that ensures that the security solution is installed on the client computer or computers, [] assures that these clients are running the correct version of this security solution[,] and enforces any additional security policies selected by the administrator."). Herrmann discloses that an endpoint application (i.e., the TrueVector engine) may be configured to monitor at least a subset of operating conditions (e.g., "[i]f compliance only requires the presence of a security client, this can readily be determined by loading a TrueVector engine API library . . . and calling an API function to determine if the security client is running" and alternatively Herrmann discloses that "[t]hose skilled in the art will recognize that a similar compliance checking process can be implemented using various other security engines, including anti-virus, firewall, and/or spyware checkers."). Herrmann II (and the incorporated Freund II) discloses that an endpoint application (e.g., software agents TrueVector service 320, integrity agent 324, and anti-virus plug-in 326) may be configured to monitor at least a subset of operating conditions (e.g., "[t]he TrueVector® service or engine 320 is a module installed on the client computer 310 that is responsible for handling higher-level security and access rules" and via a "client-side module" disclosed by Freund II that is a "monitor component or program" that runs on an endpoint to monitor operating conditions). Farmer discloses that an endpoint application (e.g., data gathering engine 222 and/or policy engine 22) may be configured to

monitor at least a subset of operating conditions (e.g., "without limitation, user information, such as password and user name information, network information . . . host configuration information . . . file system information . . . and information about currently running applications"). Check Point R60 discloses that an endpoint may run additional scripts/application via "ScriptRun," which "can run a script which performs additional configuration checks," and which may be configured to monitor at least a subset of operating conditions, and further discloses Third Party SCV checks, which are applications written by third parties using the SCV SDK and which can be configured to monitor at least a subset of the operating conditions. Corley teaches a data processing system that comprises client computational device 716 (i.e., endpoint) which comprises network security monitoring agent 718 that consists a copy of filtering policy 720 which contains rules and/or conditions that need to be checked against the network-related activities of client 716. Couillard discloses the installation and configuration of an agent software on the endpoint device to provide a list of compliance rules to verify any number of operating conditions on the endpoint device. Kuo teaches unified security management for endpoint protection that comprises software security sub 101 which includes unified agent 109 that includes a function like monitoring host activities. The sub-agent of unified agent 109 uses a data collector to periodically examine the host's configuration database, software version, patch signature information, and security profile data. Cheng teaches that client agent software is an application installed and running on a client machine that can apply a policy rule set that includes a set of rules where only one of the rules must be satisfied. A person of skill in the art would be motivated to combine these references because they each disclose methods and systems for monitoring operating conditions and were thus in the same general field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A

person of skill in the art would have been motivated to monitor the operating conditions with an endpoint application, to afford additional flexibility. For example, it would have been obvious to a person of skill to look at how other solutions in this field accomplished monitoring operating conditions, such as via an endpoint application. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’038 and ’918 patents, to the extent the references charted in any of the charts in Exhibits E, H and I do not expressly disclose “wherein the endpoint comprises a mobile device,” and similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Cheng, Chickering, Farmer, Freund, Hermann I, Hermann II (and the incorporated Freund II), Sobel, Corley, Couillard, Kuo, Bluestone, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief. Check Point R60, Elemental Security, iPass. A person of skill in the art would be motivated to combine these references because they each disclose methods and systems for monitoring operating conditions and were thus in the same general field. It is the nature of

those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill in the art would have been motivated to monitor various endpoints, including mobile devices. For example, it would have been obvious to a person of skill to look at how other solutions in this field accomplished monitoring operating conditions and the types of endpoints. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results. A person of ordinary skill would have been motivated because mobile devices are one type of endpoint and it was well known that they could run software that was susceptible to security issues.

- For the Asserted Claims of the ’038 and ’918 patents, to the extent the references charted in any of the charts in Exhibits E, H and I do not expressly disclose “controlling access of the endpoint to computing resources,” and similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Cheng, Chickering, Farmer, Freund, Hermann I, Hermann II (and the incorporated Freund II), Sobel, Corley, Couillard, Kuo, Bluestone, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform,

iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief. A person of skill in the art would be motivated to combine these references because they each disclose methods and systems for monitoring operating conditions and were thus in the same general field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. For example, it would have been obvious to a person of skill to look at how other solutions in this field accomplished monitoring operating conditions and the types of endpoints. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results. A person of ordinary skill would have been motivated because an endpoint that is not compliant with a policy may pose a security risk and thus restricting access of the potentially malicious endpoint to computing resources may prevent harm.

- For the Asserted Claims of the ’038 and ’918 patents, to the extent the references charted in any of the charts in Exhibits E, H and I do not expressly disclose “wherein the computing system comprises a plurality of servers,” and similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Cheng, Chickering, Farmer, Freund, Hermann I, Hermann II (and the incorporated Freund II), Sobel, Corley, Couillard, Kuo, Bluestone, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security

benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief. A person of skill in the art would be motivated to combine these references because they each disclose methods and systems for monitoring operating conditions and were thus in the same general field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. For example, it would have been obvious to a person of skill to look at how other solutions in this field accomplished monitoring operating conditions and the types of endpoints. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results. A person of ordinary skill would have been motivated because use of multiple servers was a known alternative to a single server.

- For the Asserted Claims of the ’038 and ’918 patents, to the extent the references charted in any of the charts in Exhibits E, H and I do not expressly disclose “wherein the plurality of policies includes at least one policy that includes the at least one rule that identifies the action,” and similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Cheng, Chickering, Farmer, Freund, Hermann I, Hermann II (and the incorporated Freund

II), Sobel, Corley, Couillard, Kuo, Bluestone, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief. A person of skill in the art would be motivated to combine these references because they each disclose methods and systems for monitoring operating conditions and were thus in the same general field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. For example, it would have been obvious to a person of skill to look at how other solutions in this field accomplished monitoring operating conditions and the types of endpoints. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results. A person of ordinary skill would have been motivated because placing a rule within a policy would be a logical way to organize data.

- For the Asserted Claims of the ’038 and ’918 patents, to the extent the references charted in any of the charts in Exhibits E, H and I do not expressly disclose “maintaining the plurality of policies in a data store on the computing system,” and similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references

identified in Section III.A, including, for example, Cheng, Chickering, Farmer, Freund, Hermann I, Hermann II (and the incorporated Freund II), Sobel, Corley, Couillard, Kuo, Bluestone, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief. A person of skill in the art would be motivated to combine these references because they each disclose methods and systems for monitoring operating conditions and were thus in the same general field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. For example, it would have been obvious to a person of skill to look at how other solutions in this field accomplished monitoring operating conditions and the types of endpoints. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results. A person of ordinary skill would have been motivated because storing the policy at the computing system would store the policy in a central location.

- For the Asserted Claims of the ’038 and ’918 patents, to the extent the references charted in any of the charts in Exhibits E, H and I do not expressly disclose “one or more hardware processors at the computing system configured to,” and similar limitations, it would have been obvious to combine the teachings of any of these charted references with the

knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Cheng, Chickering, Farmer, Freund, Hermann I, Hermann II (and the incorporated Freund II), Sobel, Corley, Couillard, Kuo, Bluestone, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief. A person of skill in the art would be motivated to combine these references because they each disclose methods and systems for monitoring operating conditions and were thus in the same general field. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. For example, it would have been obvious to a person of skill to look at how other solutions in this field accomplished monitoring operating conditions and the types of endpoints. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results. A person of ordinary skill would have been motivated because electronic behavior can be configured into hardware.

Consistent with the exemplary motivations to combine each of the references discussed above, the below table provides a summary of the exemplary combinations of references rendering obvious

to a POSA, with or without a POSA's knowledge (including the AAPA and general state of the art), skill and/or creativity, the Asserted Claims of the '038 and '918 Patents.

Reference	Reference Alone or in Combination With One or More of the Following
Cheng	Chickering, Farmer, Freund, Hermann I, Hermann II, Sobel, Corley, Couillard, Kuo, Bluestone, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief.
Chickering	Cheng, Farmer, Freund, Hermann I, Hermann II, Sobel, Corley, Couillard, Kuo, Bluestone, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief.
Farmer	Cheng, Chickering, Freund, Hermann I, Hermann II, Sobel, Corley, Couillard, Kuo, Bluestone, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief.
Freund	Cheng, Chickering, Farmer, Hermann I, Hermann II, Sobel, Corley, Couillard, Kuo, Bluestone, Wright, C CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief.
Hermann I	Cheng, Chickering, Farmer, Freund, Hermann II, Sobel, Corley, Couillard, Kuo, Bluestone, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief.

Reference	Reference Alone or in Combination With One or More of the Following
Hermann II	Cheng, Chickering, Farmer, Freund, Hermann I, Sobel, Corley, Couillard, Kuo, Bluestone, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief.
Sobel	Cheng, Chickering, Farmer, Freund, Hermann I, Hermann II, Corley, Couillard, Kuo, Bluestone, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief.
Corley	Cheng, Chickering, Farmer, Freund, Hermann I, Hermann II, Sobel, Couillard, Kuo, Bluestone, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief.
Couillard	Cheng, Chickering, Farmer, Freund, Hermann I, Hermann II, Sobel, Corley, Kuo, Bluestone, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief.
Kuo	Cheng, Chickering, Farmer, Freund, Hermann I, Hermann II, Sobel, Corley, Couillard, Bluestone, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief.
Bluestone	Cheng, Chickering, Farmer, Freund, Hermann I, Hermann II, Sobel, Corley, Couillard, Kuo, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security,

Reference	Reference Alone or in Combination With One or More of the Following
	InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief.
Wright	Cheng, Chickering, Farmer, Freund, Hermann I, Hermann II, Sobel, Corley, Couillard, Kuo, Bluetsone, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief.
CyberGatekeeper and/or associated publications including Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity	Cheng, Chickering, Farmer, Freund, Hermann I, Hermann II, Sobel, Corley, Couillard, Kuo, Bluestone, Wright, Elemental Security, Check Point R60, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief.
Elemental Security and/or associated publications	Cheng, Chickering, Farmer, Freund, Hermann I, Hermann II, Sobel, Corley, Couillard, Kuo, Bluestone, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief.
iPass and/or associated publications,	Cheng, Chickering, Farmer, Freund, Hermann I, Hermann II, Sobel, Corley, Couillard, Kuo, Bluestone, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan

Reference	Reference Alone or in Combination With One or More of the Following
including iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief	Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration IPass Endpoint Policy Management, IPass Endpoint Technical Brief.
Check Point R60 and/or associated publications	Cheng, Chickering, Farmer, Freund, Hermann I, Hermann II, Sobel, Corley, Couillard, Kuo, Bluestone, Wright, CyberGatekeeper, Behr, Demaria, InfoExpress Web, InfoExpress lan security benefits, InfoExpress lan Security, InfoExpress lan Security FAQ, InfoExpress Security Products, InfoExpress Product Documentation, Enforcing Network Integrity, Check Point R60, Elemental Security, iPass, iPass Company Story, iPass Company Story, iPass Platform, iPass Policy Orchestration, iPass Endpoint Policy Management, IPass Endpoint Technical Brief.

Additionally, Defendant incorporates by reference each of the obviousness combinations and motivations to combine presented in the petition for the following *inter partes* review proceeding(s) (IPRs) and any additional petitions for IPRs that may be filed challenging the patentability of the ’038 and/or ‘918 patent:

- IPR2023-00573
- IPR2023-00574

6. ’948 Patent

To the extent that Plaintiff contends that any of the references identified as anticipatory references fail to disclose one or more elements, those anticipatory references render the Asserted Claims obvious, either alone or in combination with other references. Defendant contends that one of skill in the art, at the time the alleged inventions were made, would have been motivated to combine the references disclosed herein in such a way as to reach the alleged inventions, as

described in further detail below. Defendant presently believes a reasonable basis exists that each of the claims asserted against it would have been obvious to one of ordinary skill in the art at the time of the alleged invention.

Below, Defendant identifies their current bases for obviousness. As seen below, Defendant presently asserts certain combinations of prior art reference(s) with or without the knowledge of one of ordinary skill in the art. Defendant offers these combinations without conceding that the reference(s) standing alone are insufficient to invalidate the relevant claims. Defendant specifically reserves their rights to combine any of its anticipatory references or any combinations of references with the knowledge of ordinary skill in the art to rebut a contention of Plaintiff that such reference(s) standing alone do not disclose every element of the Asserted Claims.

Defendant lists below prior art references and combinations of prior art references now known to Defendant, which Defendant contends render at least obvious the '948 Asserted Claims. The corresponding charts in Exhibit F for each prior art reference identifies where and how each alleged item of prior art meets one or more limitations of the Asserted Claims. Pursuant to Patent Local Rule 3-6, Defendant expressly reserves the right to supplement these charts to provide additional details about obviousness should the Court's claim construction require, or should Plaintiff serve adequate Infringement Contentions that better serve to notify Defendant which elements of their products allegedly correspond with the claim elements. Defendant's contention that the references in this section in various combinations render obvious the '948 Asserted Claims under 35 U.S.C. § 103 is in no way an admission or suggestion that each reference does not independently anticipate or render obvious the '948 Asserted Claims under 35 U.S.C. § 102. These combinations would have been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field.

It would been obvious to combine each of the references discussed below because they are analogous art generally directed to network security. The technology described in the ’948 Patent was widely known before the alleged priority date of the ’948 Patent and provided numerous motivations to combine the prior art references and systems as exemplified below. The ’948 Patent combines these components according to their known, predictable and intended uses, just as they were and/or would have been combined by a POSA in the prior art. None of the combinations presents an unexpected result, a counterintuitive combination, or a technological challenge; and no such description exists in the common specification of the asserted patents.

Accordingly, the prior art cited herein would have been readily combined according to methods known to those of ordinary skill within the field of network security to yield predictable results. The substitution of one element in this system could have been predictably achieved by one of ordinary skill at the time of the alleged invention. One of ordinary skill in the art would have been able to select appropriate attributes of one for inclusion in another. Those of ordinary skill in the art could have employed known techniques to improve similar prior art devices in the same way as claimed in the ’948 Patent. Thus, the combinations all have well defined, predictable, expectation of success.

- For the Asserted Claims of the ’948 patent, to the extent the preamble of Claim 1 is found to be limiting, and to the extent the references charted in any of the charts in Exhibit F do not expressly disclose a “a native computing environment” or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent U.S. Patent No. 10,692,032 (“Datta Ray”) or its provisional application U.S. Provisional Patent Application No. 61/547,610 (“Datta Ray Provisional”) is deemed to lack sufficient disclosure of “a native computing

environment" in Claim 1, it would have been obvious to a POSITA that the system disclosed in Datta Ray could have been implemented in a native computing environment. Using a native computing environment is well known in the art. The '948 Patent acknowledges that the "computing environment" was well known. Specifically, the '948 Patent describes the term "computing environment" several times as "cloud computing environment." And the "cloud computing environment" had been used by many enterprises or companies at the time. For example, "[m]any enterprises and organizations are moving there [sic] IT infrastructures to cloud computing environments (from self-managed on-premise data centers to service-provider managed outsourced virtual data centers)," *id*., 1:46-49; "[d]ata confidentiality, data integrity and data availability can be maintained in the cloud computing environment even when such applications may be controlled by these third parties," *id*., 1:60-63; "methods, apparatuses, systems and computer readable media comprise a plurality of services that enable visibility, control, and/or compliance in a cloud computing environment with a runtime dashboard capable of displaying operational integrity metrics of systems based on a plurality of threat vectors," *id*., 4:39-44; "the servers can also be part of a remote, service-provider managed outsourced virtual data center (i.e., a cloud computing environment)." *Id*., 35:2-4. A "native" environment is also well known in the art. The '948 Patent describes the term "native" as part of the terms "native machine instrumentation" (or "native device and/or platform instrumentation") and "native interface." *See, e.g.*, *id*., 5:55; 13:13-14; 13:43-56; 26:62-63. The native machine instrumentation or native device can represent any event subscription, callback, notification mechanism provided by the supported operating system (OS) on a device, and the native machine instrumentation can generate raw events. *Id*., 13:43-51. The native machine instrumentation or native device as the '948 Patent describes was well known. For example, Njemanze describes a native computing environment including a native device

that receives the event information from the associated network devices or applications in the devices' or applications' native formats and convert (or normalize) the information to a common schema. *See* Njemanze, ¶¶ 23, 29. Kraemer describes operating within computer systems that can be any operating system including a native operating system. Kraemer, 8:55-59. Farley describes generating raw events by detectors of intrusion detection systems. Farley, 4:10-11. Datta Ray discloses "pervasive, domain and situational-aware, adaptive, automated, and coordinated analysis and control of enterprise-wide computers, networks, and applications." Datta Ray, 1:28-33; *see also* U.S. Provisional Patent Application No. 61/547,610 ("Datta Ray Provisional"), p.1. Datta Ray discusses "an enterprise-wide computer network … as the aggregation of hardware and software comprising subsystems that include (a) One or more operations technology (OT) systems supporting the various business processes of an enterprise; and (b) One or more information technology (IT) systems supporting all other business and information processing needs of the enterprise." Datta Ray, 11:45-53; *see also* Datta Ray Provisional, p.6. It would have been obvious to a POSITA that such a computer network could be a native computing environment.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose a "network analyzer, an integrity processor, an event correlation matrix, a risk correlation matrix, and a trust supervisor" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Datta Ray is deemed to lack sufficient disclosure of "a network analyzer, an integrity processor, an event correlation matrix, a risk correlation matrix, and a trust supervisor" in a native

computing environment in Claim 1, it would have been obvious to a POSITA to name the functional components of Datta Ray's computing environment so that each functional component performs a particular function. Data Ray describes that the cyber security and risk management system has the ability to process or analyze a broad range of contextual information which includes real-time operating conditions, as well as physical, operational, legal, and regulatory constraints of the enterprise IT infrastructure, which also hosts the cyber security infrastructure (ST); trends in situational transitions, e.g., traffic flow, performance curves, of the above two. Datta Ray, 7:36-62; *see also* Datta Ray Provisional, p.5. It would have been obvious to a POSITA that this function could be performed by a functional unit named "a network analyzer." Data Ray describes that the cyber security and risk management system has the ability to process or analyze correlating information of prevailing threat environment, as well as security and operational blueprints, policies, processes, and rules of the enterprise. Datta Ray, 7:36-62; *see also* Datta Ray Provisional, p.5. It would have been obvious to a POSITA that this function could be performed by a functional unit named "an integrity processor." Data Ray also describes a function for event, security and business risk (threat profile (TPj) for a business function (BFi)). Datta Ray, 22:63-23:8, *see also* Datta Ray Provisional, p.15. It would have been obvious to a POSITA that this function could be performed by a functional unit named "an event correlation matrix" or "a risk correlation matrix." Data Ray further describes the cyber security and risk management system has the ability to process or analyze correlating information of active solicitation of information, evidence, and knowledge from internal, e.g., normal operating domain, and external sources. Datta Ray, 7:36-62, *see also* Datta Ray Provisional, p.5. It would have been obvious to a POSITA that this function could be performed by a functional unit named "a trust supervisor." It would have been obvious to

a POSITA that the above functions or matrices are for separate components, and it would have been a design choice to separate them out or have them integrated.

- For the Asserted Claims of the ’948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose “a native computing environment” or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent the preamble of Claim 1 is found to be limiting and Datta Ray is deemed to lack sufficient disclosure of “a native computing environment” in Claim 1, a POSITA would have also been motivated to combine the teachings of Datta Ray with the teachings of U.S. Patent Publication No. 2011/0145711 (“Njemanze”), which describes a native computing environment including a native device that receives the event information from the associated network devices or applications in the devices' or applications' native formats and convert (or normalize) the information to a common schema. *See* Njemanze, ¶¶ 23, 29. Both Datta Ray and Njemanze disclose monitoring and analyzing security events of computer networks. Datta Ray is related to “pervasive, domain and situational-aware, adaptive, automated, and coordinated analysis and control of enterprise-wide computers, networks, and applications for mitigation of business and operational risks, including efficiency and effectiveness of business processes and enhancement of cyber security.” Datta Ray, 1:28-33; *see also* Datta Ray Provisional, p.1. Datta Ray describes that the enterprise-wide computer network is an “aggregation of hardware and software,” Datta Ray, 11:45-53; *see also* Datta Ray Provisional, p.6, like any other computer network. A POSITA would have understood that a computer network can be a native computing environment such as the one disclosed by Njemanze. Njemanze is related to “provid[ing]

a historical view of the event traffic" including that of the security events of computer network devices and its method is applicable to "any computer-based or network-based security system." Njemanze, ¶¶ 18-19, 49. Njemanze describes that agents 12 can collect events from any source that produces event logs or messages and can operate at the native device. *Id*., ¶ 23. The agents 12 also receive the event information from the associated network devices or applications in the devices' or applications' native formats and convert (or normalize) the information to a common schema. *Id*., ¶ 29. A POSITA would have been motivated to use Njemanze's teachings of applying a native computing environment to monitor the security events of a network of Datta Ray in order to mitigate risks to Datta Ray's system.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "a generating real-time behavior based events for determining the real-time operational integrity" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Datta Ray is deemed to lack disclosure of "generating real-time behavior based events for determining the real-time operational integrity" in Claim 1, a POSITA would have been motivated to combine the teachings of Datta Ray with the teachings of Njemanze. The '948 Patent acknowledges that existing technology enables warning for "threat identification and diagnosis" based on forensic evidence and evaluation of the identity and integrity of components of a target system. *Id*., 2:48-58. As discussed below, continuously monitoring the real-time operations of applications was well known in the art, and the art usually "determine[d] the real-time operational integrity of the application" based on the monitored real-time operations of

applications. For example, Data Ray describes that the cyber security and risk management system has the ability to process or analyze a broad range of contextual information which includes real-time operating conditions. Datta Ray, 7:36-62; *see also* Datta Ray Provisional, p.5. Datta Ray further describes that the system that is "provided with a broad range of contextual information and the ability to process that information" "[t]o identify and synthesize appropriate security and control postures automatically." Datta Ray, 7:36-62; *see also* Datta Ray Provisional, p.5. Njemanze describes analyzing event data "live." Njemanze, ¶¶ 18, 37. "[G]enerating real-time behavior based events" for determining the real-time operational integrity of the application was also well known in the art. For example, Njemanze describes generating meta-events or alerts from the analysis the live event data. Njemanze, ¶¶ 18, 37, and the generated meta-events are for investigating the events to gather information regarding similar attack profiles and/or to take action in accordance with specified procedures. *Id*., ¶ 35. Kraemer describes generating probabilistic signature for identifying malicious attacks. Kraemer, 3:45-50. As discussed above, Data Ray describes that the cyber security and risk management system has the ability to process or analyze a broad range of contextual information which includes real-time operating conditions. Datta Ray, 7:36-62; *see also* Datta Ray Provisional, p.5. Datta Ray further describes a cyber security and risk management system that is "provided with a broad range of contextual information and the ability to process that information" "[t]o identify and synthesize appropriate security and control postures automatically." Datta Ray, 7:36-62; *see also* Datta Ray Provisional, p.5. Njemanze describes analyzing event data "live" and generating meta-events or alerts from the analysis. Njemanze, ¶¶ 18, 37. Njemanze's meta-events are for investigating the events to gather information regarding similar attack profiles and/or to take action in accordance with specified procedures. *Id*., ¶ 35. Moreover, the investigation is by accessing a knowledge base 28 that contains reference documents

that provide a description of the threat, recommended solutions, reference information, company procedures and/or links to additional resources. *Id*. Because Datta Ray's cyber security and risk management system is designed to mitigate risks, a POSITA would have been motivated to use Njemanze's method of generating meta-events or alerts to automatically identify and synthesize appropriate security and control postures of Datta Ray. A POSITA would have readily appreciated the value of implementing generating meta-events or alerts as taught by Njemanze in Datta Ray's cyber security and risk management system.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "in a plurality of runtime dashboards of an administrative console of the computing environment" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Datta Ray is deemed to lack disclosure of "in a plurality of runtime dashboards of an administrative console of the computing environment" in Claim 1, a POSITA would have been motivated to combine the teachings of Datta Ray with the teachings of Njemanze. The '948 patent acknowledges that the terms "console display," "display," "display screen," and "screen" are used interchangeably to refer to "any type of display device or screen coupled to or integrated with a computing device for displaying content viewable by a user of the computing device." '948 Pat, 26:47-51. Such a "display" can be a touch-screen liquid crystal display (LCD), GUIs viewed on a display of a mobile device, a server (i.e., a server console), a desktop computer (i.e., a PC monitor), or a laptop display. *Id*., 26:54-60. The 948 patent also uses the GUI and dashboard interchangeably. For example, the '948 patent describes

"the score and weight can be predefined or customizable through a graphical user interface GUI or dashboard, such as the GUIs described below with reference to FIGS. 14-18." *Id*., 17:4-7. It is well known in the art that displays, GUIs, or dashboards are used to display information. For example, Datta Ray describes various forms to present information: "graphical displays to show quantitative, geographical and temporal information at a glance, highlight MCE that need extra attention, to archive logs sorted and classified according to various specified criteria, provide summaries as needed to support the generation of various compliance reports, history of significant security events along with their actual and potential adverse impacts on business processes." Datta Ray, 26:57-27:3; *see also* Datta Ray Provisional, p.19. Marvasti describes displaying, in a dashboard where administrators of the complex system are alerted of a potential abnormality, alerts and the criteria and metrics that are indicating an abnormality for optimization of the complex system, study of the system, or to resolve problems. *Id*., 10:56-64. It is also well known in the art that using a display, GUI, or dashboard can display real-time status information. For example, Data Ray describes that the cyber security and risk management system has the ability to process or analyze a broad range of contextual information which includes real-time operating conditions. Datta Ray, 7:36-62; *see also* Datta Ray Provisional, p.5. Njemanze describes displaying, on a GUI for console 16, security events and a status bar that describes the current state of the time-based events. Njemanze, ¶ 49. Marvasti is configured to provide integrity management through the real-time monitoring and alerting of variations in complex systems. Marvasti, 12:19-34. Kraemer describes that security agents can mark or otherwise identify sequences of processing operations that led up to the security violation as a disallowed sequence of processing operations and can disseminate this information in real-time (e.g., upon detection). Kraemer, 4:22-32. As discussed above, Data Ray describes that the cyber security and risk management system has the ability to

process or analyze a broad range of contextual information which includes real-time operating conditions. Datta Ray, 7:36-62; *see also* Datta Ray Provisional, p.5. Datta Ray further describes various forms to present information: "graphical displays to show quantitative, geographical and temporal information at a glance, highlight MCE that need extra attention, to archive logs sorted and classified according to various specified criteria, provide summaries as needed to support the generation of various compliance reports, history of significant security events along with their actual and potential adverse impacts on business processes." Datta Ray, 26:57-27:3; *see also* Datta Ray Provisional, p.19. A POSITA would have found it inherent or obvious that the information presentation forms such as the graphical displays would be in the form of dashboards of an administrative console of a computing environment, like the ones disclosed in Njemanze. Njemanze describes displaying, on a GUI for console 16, security events and a status bar that describes the current state of the time-based events. Njemanze, ¶ 49. Because Datta Ray's cyber security and risk management system is designed to mitigate risks, a POSITA would have been motivated to use Njemanze's GUI to display security events on the graphical displays of Datta Ray. A POSITA would have readily appreciated that the functionalities of Njemanze's GUI could readily and beneficially be implemented in Datta Ray's cyber security and risk management system to mitigate risks.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "the plurality of runtime dashboards are a plurality of application runtime reputation score dashboards that comprise one or more of: a global view dashboard; a system configuration dashboard; a resource utilization dashboard; an application integrity dashboard; and a network activity dashboard" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary

skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Datta Ray is deemed to lack disclosure that "the plurality of runtime dashboards are a plurality of application runtime reputation score dashboards that comprise one or more of: a global view dashboard; a system configuration dashboard; a resource utilization dashboard; an application integrity dashboard; and a network activity dashboard" in Claim 5, a POSITA would have been motivated to combine the teachings of Datta Ray with the teachings of Njemanze. Datta Ray describes providing the relevant logs and other displays for security administrations, Datta Ray, 26:46-51; *see also* Datta Ray Provisional, p.19, as well as many forms to present information: "graphical displays to show quantitative, geographical and temporal information at a glance, highlight MCE that need extra attention, to archive logs sorted and classified according to various specified criteria, provide summaries as needed to support the generation of various compliance reports, history of significant security events along with their actual and potential adverse impacts on business processes." Datta Ray, 26:57-27:3; *see also* Datta Ray Provisional, p.19. A POSITA would have found it obvious that the information presentation forms such as the graphical displays would be in the form of dashboards, like the ones disclosed in Njemanze. Njemanze describes that the GUI for console 16 comprises a views panel that allows a user to view security events from heterogeneous sources, a navigation panel that allows a user to view system assets (as well as dashboards) in an organized manner, and a status bar that describes the current state of the time-based events. Njemanze, ¶ 49. Because Datta Ray's cyber security and risk management system is designed to mitigate risks, a POSITA would have been motivated to use Njemanze's GUI comprising different panels to display security events on the graphical displays of Datta Ray. A POSITA would have readily appreciated that the

functionalities of Njemanze's GUI could readily and beneficially be implemented in Datta Ray's cyber security and risk management system to mitigate risks.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "the scans [being] performed as per a schedule, and wherein the schedule is one or more of hourly, daily, weekly, bi-weekly, or monthly" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Datta Ray is deemed to lack sufficient disclosure of "the scans [being] performed as per a schedule, and wherein the schedule is one or more of hourly, daily, weekly, bi-weekly, or monthly" in Claim 4, a POSITA would have found it obvious to combine the teachings of Datta Ray with the teachings of U.S. 7,467,067 ("Marvasti"). Both Datta Ray and Marvasti disclose monitoring and alerting security events of computer networks. Datta Ray discloses monitoring continuously evolving threats and vulnerabilities and periodically obtaining state information in real-time. Datta Ray, 19:66-20:1; *see also* Datta Ray Provisional, pp.1, 7. Datta Ray further describes that depending on the elapsed time requirements of the relevant analytical needs, each MCE is assigned to a particular security cycle, and the cycles may be daily, hourly, 5-minutes, 1-minute, 10-seconds, 1-seconds, 100-milliseconds, etc. Datta Ray, 18:1-22; *see also* Datta Ray Provisional, p.12. Datta Ray also describes that control postures (617) may be used to adjust and/or update security and business risk administration knowledge (616) either periodically or on occurrence of a significant event, and the updates to security rules (623) may take place once an hour or once a day. Updates to security policies (624) may take place once a day, a week, or a month. Updates to security blueprints (625) may take place once a year or two. Datta Ray,

26:10-16; *see also* Datta Ray Provisional, p.18. Marvasti discloses that the dynamic thresholding engine 402 collects, evaluates, and determines new thresholds at midnight each night, or resets the dynamic thresholds at other time intervals such as weekly, biweekly, or monthly. *See* Marvasti, 8:17-27. Datta Ray is configured to provide "automated and adaptive methodologies to monitor situational contexts and refine their control postures as responses to such changes" as threats and vulnerabilities to existing and emergent services are continuously evolving. Datta Ray, 2:39-44; *see also* Datta Ray Provisional, p.1. In this way, Datta Ray could "improve the integrity of such dynamic and interconnected risk management system." *Id*. A POSITA would have been motivated to use Marvasti's method or system of monitoring on a daily, weekly, 229biweekly, or monthly schedule to better monitor situational contexts as threats and vulnerabilities are continuously evolving. A POSITA would have readily appreciated that the method of daily, weekly, biweekly, or monthly monitoring as taught by Marvasti could readily and beneficially be implemented in Datta Ray's monitoring and alerting system.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "monitoring…system operations initiated by the application, a runtime configuration of the application" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Njemanze is deemed to lack disclosure of "monitoring…system operations initiated by the application, a runtime configuration of the application" in Claim 1, a POSITA would have been motivated to combine Njemanze with Datta Ray's teachings on monitoring. The '948 Patent acknowledges that it was well known in the art that "[a]pplication runtime operational

integrity is a critical factor in building end-to-end trust from device to service." *Id*., 2:18-20. And the patent describes the current security appliances such as network firewalls and intrusion detection/prevention systems and antivirus and network-based integrity measurement and verification services. *Id*., 2:18-31. Indeed, monitoring by a plurality of sensory inputs was well known. For example, Datta Ray describes that events in the real world (601) drive the inputs to one or more security monitors. Datta Ray, 19:3-7; *see also* Datta Ray Provisional, p.12. Farley describes that the multiple data sources can comprise sensors or detectors that monitor network traffic or individual computers or both. Farley, Abstract. Kraemer describes a system receives information from at least one security interceptor associated with at least one computer system. Kraemer, Abstract. Marvasti describes capturing fingerprints of the relevant nodes within a complex system. Marvasti, Abstract. Monitoring various parameters including one or more of network dialogs of the application, system operations initiated by the application, a runtime configuration of the application, resource utilization by the application, and integrity of the application was well known. For example, Lo describes monitoring performance of network infrastructure and applications. Lo, Abstract. Marvasti describes choosing metrics such as packets received, packets sent, packets dropped, and collisions in a network to monitor chosen variables within each node. Marvasti, 4:59-5:3. Uner describes that the system security monitor "may observe the operations of the application, process or task for any activity that violates any of the security rules in the monitor." Uner, 4:49-53. Both Njemanze and Datta Ray disclose monitoring and analyzing security events of computer networks. Njemanze discloses monitoring security events from operating system logs, Njemanze, ¶ 23, or creating rules designed to capture threats and attacks that are typical in large, diverse networks and may be organized to provide multiple lines of defense by detecting specific activities and grouping them according to level of threat. *Id*., ¶ 41. Datta Ray describes monitoring or

obtaining vulnerability knowledge in IT systems (611) such as applications running on an operating system suffering from SQL injection or buffer overflow weakness. Datta Ray, 19:3-7, 21:53-57; *see also* Datta Ray Provisional, p.14. Datta Ray further describes that the cyber security and risk management system is provided with a broad range of contextual information consists of data from various sources including, but not limited to: Real-time operating conditions, as well as physical, operational, legal, and regulatory constraints of the enterprise business and operational processes; Real-time operating conditions, as well as physical, operational, legal, and regulatory constraints of the enterprise IT infrastructure, which also hosts the cyber security infrastructure (ST). Datta Ray, 7:32-62; *see also* Datta Ray Provisional, p.5. Because Njemanze's system is configured to "provide accurate and timely intrusion detection and alert generation so as to effectively combat attempts to compromise a computer network or system," Njemanze, ¶ 8, a POSITA would have been motivated to use Datta Ray's method or system of monitoring or obtaining vulnerability knowledge such as applications running on an operating system or contextual information such as real-time operating conditions to achieve Njemanze's purpose. A POSITA would have readily appreciated that the method of monitoring or obtaining vulnerability knowledge such as applications running on an operating system or contextual information such as real-time operating conditions as taught by Datta Ray could readily and beneficially be implemented in Njemanze's monitoring and alerting system.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "the real-time operations of applications are monitored continuously" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example,

to the extent Njemanze is deemed to lack disclosure of "the real-time operations of applications are monitored continuously" in Claim 2, a POSITA would have been motivated to combine Njemanze with Datta Ray's teachings on monitoring. Continuously monitoring the real-time operations of applications was also well known. For example, Datta Ray describes that continuously evolving threats and vulnerabilities are monitored or state information is obtained in real-time periodically. Datta Ray, 2:39-44, 19:66-20:1; *see also* Datta Ray Provisional, pp.1, 7. Kraemer describes that security agent processes 150 and management center application 160 dynamically detecting malicious attacks. Kraemer, 7:26-31. Njemanze discloses real-time capturing and analyzing event data "live" (i.e., at a time when the event data was captured from the sources) from a variety of network security devices and/or applications. Njemanze, ¶¶ 18, 23, 37. It would have been obvious to a POSITA that capturing and analyzing event data are performed continuously to obtain real-time data and to analyze the data live events. Datta Ray describes that continuously evolving threats and vulnerabilities are monitored or state information is obtained in real-time periodically. Datta Ray, 2:39-44, 19:66-20:1; *see also* Datta Ray Provisional, pp.1, 7. Thus, a POSITA would have readily used Datta Ray's method or system of continuously monitoring or periodically obtaining state information to achieve Njemanze's objective.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "the scans [being] performed as per a schedule, and wherein the schedule is one or more of hourly, daily, weekly, bi-weekly, or monthly" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Njemanze is deemed to lack sufficient disclosure of "the scans [being] performed as per a

schedule, and wherein the schedule is one or more of hourly, daily, weekly, bi-weekly, or monthly" in Claim 4, a POSITA would have found it obvious to combine the teachings of Njemanze with the teachings of Marvasti. Both Njemanze and Marvasti disclose monitoring and alerting security events of computer networks. Njemanze discloses real-time capturing and analyzing event data "live" (i.e., at a time when the event data was captured from the sources) from a variety of network security devices and/or applications. Njemanze, ¶¶ 18, 23, 37. Marvasti discloses that the dynamic thresholding engine 402 collects, evaluates, and determines new thresholds at midnight each night, or resets the dynamic thresholds at other time intervals such as weekly, biweekly, or monthly. *See* Marvasti, 8:17-27. Because Njemanze's system is configured to "provide accurate and timely intrusion detection and alert generation so as to effectively combat attempts to compromise a computer network or system," Njemanze, ¶ 8, a POSITA would have been motivated to use Marvasti's method or system of monitoring on a daily, weekly, biweeky, or monthly schedule to capture and analyze event data. A POSITA would have readily appreciated that the method of daily, weekly, biweekly, or monthly monitoring as taught by Marvasti could readily and beneficially be implemented in Njemanze's monitoring and alerting system.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "correlating… threat classifications based on the temporal sequence of" the events in Claim 1 or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Marvasti is deemed to lack sufficient disclosure of "correlating… threat classifications based on the temporal sequence of" the

events in Claim 1, a POSITA would have found it obvious to combine the teachings of Marvasti with the teachings of U.S. 8,141,157 ("Farley"). The '948 patent acknowledges that the industry has adopted multi-factor authentication technologies for increased protection from malicious activities. '948 Pat, 2:57-58. Classification of threat based on the temporal sequence of events was well known in the art. For example, Farley's raw event classification database can permit a fusion engine to classify raw computer events while the knowledge database can permit the fusion engine to rank and evaluate the risk of a raw computer event based upon the context of the raw computer event. *Id*., 4:44-58. Lo describes using a moving window to approximate a real-time environment and that the difference between the moving window and the baseline values are classified by whether it is within or exceeding the threshold. Lo, ¶¶ 88, 89. Correlating, by certain rules or matrices, threat classification was also well known. For example, Farley describes using correlation rules together with the raw event classification database to identify event patterns that indicate either malicious or nonmalicious activity. *See* Farley, 19:65-20:1. Marvasti discloses comparing the monitored data to heuristics, which represent events or metric conditions and are generated from previously gathered data from prior variations that led to certain events (e.g., abnormalities, problems, etc.). Marvasti, 3:60-66, 5:32-41. Marvasti also describes matching substantially real-time data in the complex system against the fingerprint indicative of potential events. *Id*., 6:41-49. Kraemer describes that "[w]hen the probability of an attack being executed crosses a specified threshold, the probabilistic signature generation process probabilistically links the observed events to the probability that an attack is occurring, and attempts to identify a root cause." Kraemer, 4:41-52. Lo describes an Engine correlates various moving window metrics and the baseline values of those metrics and determining if the composite different metric is within or exceeding the threshold. Lo, ¶ 89. Both Marvasti and Farley disclose identifying security

events of computer networks. Marvasti discloses comparing the monitored data to heuristics, which represent events or metric conditions and are generated from previously gathered data from prior variations that led to certain events (e.g., abnormalities, problems, etc.). Marvasti, 3:60-66, 5:32-41. Marvasti also describes matching substantially real-time data in the complex system against the fingerprint indicative of potential events. *Id*., 6:41-49. Farley describes using correlation rules with the raw event classification database to identify event patterns that indicate either malicious or nonmalicious activity. *See* Farley, 19:65-20:1. Farley further describes that a fusion engine can utilize the raw event classification database and the knowledge database to assess risks and determine ranks of real-time raw events. Particularly, Farley's raw event classification database can permit the fusion engine to classify raw computer events while the knowledge database can permit the fusion engine to rank and evaluate the risk of a raw computer event based upon the context of the raw computer event. *Id*., 4:44-58. Farley also correlates threat classifications based on temporal sequence of the generated event. For example, Farley determines whether the raw event (integrity attack) being considered as an inbound attack is an integrity attack and occurs earlier than at least one event in the outbound attacks list of the current correlation event. Farley generates an Attack From Attacked Host event when an Integrity attack is seen against a host followed by a Confidentiality, Integrity, or Availability attack originating from that host. When the controller 655 loads a correlation rule 620 such as the AFAH rule during system initialization, it associates the rule with all of the event types included in the event categories of interest to the rule. *Id*., 30:27-42. Because Marvasti's system is configured to "monitor[] variations or abnormalities in complex systems … to predict the occurrence of an event before the event occurs based upon similarities to a prior history of variations or abnormalities within the complex system," Marvasti, 2:39-44, a POSITA would have been motivated to use Farley's method

or system to identify event patterns that indicate either malicious or nonmalicious activity using correlation rules and the raw event classification database. A POSITA would have readily appreciated that Farley's method could readily and beneficially be implemented in Marvasti's monitoring and alerting system to achieve Marvasti's objective.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "a native computing environment" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent the preamble of Claim 1 is found to be limiting and Marvasti is deemed to lack sufficient disclosure of "a native computing environment" in Claim 1, a POSITA would have also been motivated to combine the teachings of Marvasti with the teachings of Njemanze, which describes a native computing environment including a native device that receives the event information from the associated network devices or applications in the devices' or applications' native formats and convert (or normalize) the information to a common schema. *See* Njemanze, ¶¶ 23, 29. Both Marvasti and Njemanze disclose monitoring and identifying security events of computer networks. Marvasti is configured to provide integrity management through the real-time monitoring and alerting of variations in complex systems in "special purpose computing system environments." Marvasti, 12:19-34. Njemanze is related to "provid[ing] a historical view of the event traffic" including that of the security events of computer network devices, and its method is applicable to "any computer-based or network-based security system." Njemanze, ¶¶ 18-19, 49. Njemanze describes that agents 12 can collect events from any source that produces event logs or messages and can operate at the native device. *Id.*, ¶ 23. The agents 12 also receive the

event information from the associated network devices or applications in the devices' or applications' native formats and convert (or normalize) the information to a common schema. *Id*., ¶ 29. A POSITA would have been motivated to use Njemanze's teachings of applying a native computing environment to monitor the variations in complex systems of Marvasti in order to provide integrity management of Marvasti's system.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "scans [being] performed … by a third party collaboration service to score the severity of an application based on the vulnerability, compliance, and patch status to affirm positive presence of a threat" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Marvasti is deemed to lack sufficient disclosure of "scans [being] performed … by a third party collaboration service to score the severity of an application based on the vulnerability, compliance, and patch status to affirm positive presence of a threat" in Claim 4, a POSITA would have also been motivated to combine the teachings of Marvasti with the teachings of Njemanze. Using a third party collaboration service to score the severity of an application is well known in the art. As acknowledged by the '948 Patent, it is a "recent trend in computing" that enterprise software is no longer owned by the customer, but instead the enterprise's Information Technology (IT) infrastructure can be provided by a third party and enterprise software applications are sold as service offerings." '948 Pat, 1:21-26. And many enterprises and organizations are moving their IT infrastructures from self-managed on-premise data centers to service-provider managed outsourced virtual data centers, wherein which third parties may provide shared computing resources and applications running on

those resources. *Id*., 1:46-52. Indeed, applying these third-party services to access applications was well known. For example, Njemanze discloses that knowledge base 28 contains reference documents (e.g., in the form of web pages and/or downloadable documents) from third-party sources such as user-authored articles, third-party articles, and/or security vendors' reference material. Njemanze, ¶ 35. These documents provide a description of the threat, recommended solutions, reference information, company procedures and/or links to additional resources. *Id*. Similarly, Datta Ray discloses receiving information from third-party services such as risk assessments and decisions with government agencies and third party collaborations. *See* Datta Ray, 1:54-58; *see also* Datta Ray Provisional, p.1. Marvasti uses "[a] self-learning dynamic thresholding module [] to assist in discovering when variations or abnormalities occur within the complex system." Marvasti, 2:48-51. As an example, Marvasti applies a dynamic thresholding engine 402 that determines new dynamic thresholds at predetermined time intervals and performs scans. *Id*., 8:17-27. A POSITA would have found it obvious to incorporate a third-party service to address potential abnormalities. Njemanze discloses that knowledge base 28 contains reference documents (e.g., in the form of web pages and/or downloadable documents) from third-party sources such as user-authored articles, third-party articles, and/or security vendors' reference material. Njemanze, ¶ 35. These documents provide a description of the threat, recommended solutions, reference information, company procedures and/or links to additional resources. *Id*. Because Marvasti is configured to "predict the occurrence of an event before the event occurs based upon similarities to a prior history of variations or abnormalities within the complex system," Marvasti, 2:39-44, a POSITA would have been motivated to use Njemanze's teachings of third-party sources documents in Marvasti's systems in order to better predict the occurrence of an event or abnormalities.

- For the Asserted Claims of the ’948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose “the plurality of runtime dashboards are a plurality of application runtime reputation score dashboards that comprise one or more of: a global view dashboard; a system configuration dashboard; a resource utilization dashboard; an application integrity dashboard; and a network activity dashboard” or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Marvasti is deemed to lack disclosure that “the plurality of runtime dashboards are a plurality of application runtime reputation score dashboards that comprise one or more of: a global view dashboard; a system configuration dashboard; a resource utilization dashboard; an application integrity dashboard; and a network activity dashboard” in Claim 5, a POSITA would have been motivated to combine the teachings of Marvasti with the teachings of Njemanze. Marvasti describes displaying, in a dashboard where administrators of the complex system are alerted of a potential abnormality, alerts and the criteria and metrics that are indicating an abnormality for optimization of the complex system, study of the system, or to resolve problems. *Id*., 10:56-64. A POSITA would have found it inherent or obvious that alerts and criteria and metrics that are indicating an abnormality would be displayed in a plurality of runtime dashboards, like the ones disclosed in Njemanze. Njemanze describes that the GUI for console 16 comprises a views panel that allows a user to view security events from heterogeneous sources, a navigation panel that allows a user to view system assets (as well as dashboards) in an organized manner, and a status bar that describes the current state of the time-based events. Njemanze, ¶ 49. Marvasti is configured to “provid[e] integrity management through the real-time monitoring of such variations

and the alerting of potential effects from such variations before such effects occur." Marvasti, 1:17-20. This is implemented by having "[a]n integrity management system and method … that monitors variations or abnormalities in complex systems … to predict the occurrence of an event before the event occurs based upon similarities to a prior history of variations or abnormalities within the complex system." *Id*., 2:39-44. A POSITA would have been motivated to use Njemanze's GUI comprising different panels to display security events to better facilitate Marvasti's monitoring of variations or abnormalities in complex systems. A POSITA would have readily appreciated that the functionalities of Njemanze's GUI could readily and beneficially be implemented in Marvasti's real-time monitoring and alerting system.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "monitoring … system operations initiated by the application" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Marvasti is deemed to lack sufficient disclosure of "monitoring … system operations initiated by the application" in Claim 1, a POSITA would have also been motivated to combine the teachings of Marvasti with the teachings of U.S. Patent No. 9,286,469 ("Kraemer"). Both Marvasti and Kraemer disclose monitoring of and providing security for computer networks. Marvasti describes capturing fingerprints of the relevant nodes, including software applications, within a complex system and variations or abnormalities in complex systems such as an IT infrastructure or other types of systems. *Id*., Abstract, 2:39-44, 4:45-58. Kraemer describes monitoring an instance of a new service installation or a first time instance of an application invocation. Kraemer, 10:29-44.

Because Marvasti is configured to "predict the occurrence of an event before the event occurs based upon similarities to a prior history of variations or abnormalities within the complex system," Marvasti, 2:39-44, a POSITA would have been motivated to use Kraemer's teachings of monitoring an instance of a new service installation or a first time instance of an application invocation in complex systems of Marvasti in order to capture fingerprints of the relevant nodes including software applications and to predict the occurrence of an event.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "scans [being] performed … by a third party collaboration service to score the severity of an application based on the vulnerability, compliance, and patch status to affirm positive presence of a threat" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Marvasti is deemed to lack sufficient disclosure of "scans [being] performed … by a third party collaboration service to score the severity of an application based on the vulnerability, compliance, and patch status to affirm positive presence of a threat" in Claim 4, a POSITA would have also been motivated to combine the teachings of Marvasti with the teachings of Datta Ray. Both Marvasti and Datta Ray disclose monitoring security events for computer networks. Marvasti describes applying a thresholding system that addresses potential abnormalities and predicts potential problems. Particularly, Marvasti uses "[a] self-learning dynamic thresholding module [] to assist in discovering when variations or abnormalities occur within the complex system." *Id.*, 2:48-51. As an example, Marvasti applies a dynamic thresholding engine 402 that determines new dynamic thresholds at predetermined time

intervals and performs scans. *Id*., 8:17-27. A POSITA would have found it obvious to incorporate a third-party service to addresses potential abnormalities. Datta Ray discloses such a service such as risk assessments and decisions with government agencies and third party collaborations. *See* Datta Ray, 1:54-58; *see also* Datta Ray Provisional, p.1. Because Marvasti is configured to "predict the occurrence of an event before the event occurs based upon similarities to a prior history of variations or abnormalities within the complex system," Marvasti, 2:39-44, a POSITA would have been motivated to use Datta Ray's teachings of government agencies and third party collaborations in Marvasti's systems in order to better predict the occurrence of an event or abnormalities.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "m correlating… threat classifications based on the temporal sequence of" the events in Claim 1 or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Kraemer is deemed to lack sufficient disclosure of "correlating… threat classifications based on the temporal sequence of" the events in Claim 1, a POSITA would have found it obvious to combine the teachings of Kraemer with the teachings of Farley. Both Kraemer and Farley disclose identifying security events of computer networks. Kraemer describes that "[w]hen the probability of an attack being executed crosses a specified threshold, the probabilistic signature generation process probabilistically links the observed events to the probability that an attack is occurring, and attempts to identify a root cause." Kraemer, 4:41-52. Farley describes using correlation rules with the raw event classification database to identify event patterns that indicate either malicious or nonmalicious activity. *See* Farley, 19:65-20:1.

Farley further describes that a fusion engine can utilize the raw event classification database and the knowledge database to assess risks and determine ranks of real-time raw events. Particularly, Farley's raw event classification database can permit the fusion engine to classify raw computer events while the knowledge database can permit the fusion engine to rank and evaluate the risk of a raw computer event based upon the context of the raw computer event. *Id*., 4:44-58. Farley also correlates threat classifications based on temporal sequence of the generated event. For example, Farley determines whether the raw event (integrity attack) being considered as an inbound attack is an integrity attack and occurs earlier than at least one event in the outbound attacks list of the current correlation event. *Id*., 30:27-42. Farley generates an Attack From Attacked Host event when an Integrity attack is seen against a host followed by a Confidentiality, Integrity, or Availability attack originating from that host. *Id*. When the controller 655 loads a correlation rule 620 such as the AFAH rule during system initialization, it associates the rule with all of the event types included in the event categories of interest to the rule. *Id*. Because Kraemer's system is configured to "rapidly identif[y] malicious attacks and prevents the spread of such attacks to other computer systems," Kraemer, 3:48-50, a POSITA would have been motivated to use Farley's method or system to identify event patterns that indicate either malicious or nonmalicious activity using correlation rules and the raw event classification database. A POSITA would have readily appreciated Farley's method to implement in Kraemer's monitoring and alerting system to achieve Kraemer's objective.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "scans [being] performed … by a third party collaboration service to score the severity of an application based on the vulnerability, compliance, and patch status to affirm positive presence of a threat" or similar limitations,

it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Kraemer is deemed to lack sufficient disclosure of "scans [being] performed … by a third party collaboration service to score the severity of an application based on the vulnerability, compliance, and patch status to affirm positive presence of a threat" in Claim 4, a POSITA would have also been motivated to combine the teachings of Kraemer with the teachings of Datta Ray. Both Kraemer and Datta Ray disclose monitoring and identifying malicious attacks or vulnerabilities of computer networks. Kraemer describes a probabilistic signature generation process 155 that identifies an event and weights each event to determine the probability of an attack on the computer system. Kraemer, 13:61-67. A POSITA would have found it obvious to incorporate a third-party service to detect the event that is potentially an attack. Datta Ray discloses such a service such as risk assessments and decisions with government agencies and third party collaborations. *See* Datta Ray, 1:54-58; *see also* Datta Ray Provisional, p.1. Because Kraemer is configured to "rapidly identif[y] malicious attacks and prevents the spread of such attacks to other computer systems," Kraemer, 3:48-50, a POSITA would have been motivated to use Datta Ray's teachings of government agencies and third party collaborations in Kraemer's systems in order to rapidly identify malicious attacks.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "the real-time operations of applications are monitored continuously" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted

references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Kraemer is deemed to lack disclosure of "the real-time operations of applications are monitored continuously" in Claim 2, a POSITA would have been motivated to combine Kraemer with Datta Ray's teachings on monitoring. Kraemer describes that security agents can also mark or otherwise identify sequences of processing operations that led up to the security violation as a disallowed sequence of processing operations and can disseminate this information to other security agents operating on other computer systems in the network in real-time (e.g., upon detection). Kraemer, 4:22-32. Kraemer further describes that security agent processes 150 and management center application 160 dynamically detect malicious attacks without requiring the security agent processes 150 to continuously and periodically download signature or virus definition files. *Id*., 7:26-31. It would have been obvious to a POSITA that dynamically detecting malicious attacks are performed continuously to obtain real-time data so that there is no need to "continuously and periodically download signature or virus definition files." Datta Ray describes continuously evolving threats and vulnerabilities are monitored or state information is obtained in real-time periodically. Datta Ray, 2:39-44, 19:66-20:1; *see also* Datta Ray Provisional, pp.1, 7. Thus, a POSITA would readily use Datta Ray's method or system of continuously monitoring or periodically obtaining state information to achieve Kraemer's objective.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "the scans [being] performed as per a schedule, and wherein the schedule is one or more of hourly, daily, weekly, bi-weekly, or monthly" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the

AAPA, and/or of those references identified in Section III.A. For example, to the extent Kraemer is deemed to lack sufficient disclosure of "the scans [being] performed as per a schedule, and wherein the schedule is one or more of hourly, daily, weekly, bi-weekly, or monthly" in Claim 4, a POSITA would have found it obvious to combine the teachings of Kraemer with the teachings of Marvasti. Both Kraemer and Marvasti disclose monitoring and alerting security events, such as malicious attacks, of computer networks. Kraemer discloses a system that receives information from at least one security interceptor, "the information identif[ying] details associated with a traffic flow in a computer system of the computer networking environment." Kraemer, Abstract. Kraemer describes that "security agent processes 150 and management center application 160 dynamically detect and prevent malicious attacks on the computer systems 110." *Id*., 7:26-31. Marvasti discloses that dynamic thresholding engine 402 collects, evaluates, and determines new thresholds at midnight each night, or resets the dynamic thresholds at other time intervals such as weekly, biweekly, or monthly. *See* Marvasti, 8:17-27. Because Kraemer's system is configured to "rapidly identif[y] malicious attacks and prevents the spread of such attacks to other computer systems," Kraemer, 3:48-50, a POSITA would have been motivated to use Marvasti's method or system of monitoring on a daily, weekly, 246iweekly, or monthly schedule to rapidly identify malicious attacks. A POSITA would have readily appreciated that the method of daily, weekly, biweekly, or monthly monitoring as taught by Marvasti could readily and beneficially be implemented in Kraemer's monitoring and alerting system.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "displaying, in a plurality of runtime dashboards of an administrative console of the computing environment, real-time status indications for operational integrity of the application" or similar limitations, it would have

been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Kraemer is deemed to lack disclosure of "displaying, in a plurality of runtime dashboards of an administrative console of the computing environment, real-time status indications for operational integrity of the application" in Claim 1 and "the plurality of runtime dashboards are a plurality of application runtime reputation score dashboards that comprise one or more of: a global view dashboard; a system configuration dashboard; a resource utilization dashboard; an application integrity dashboard; and a network activity dashboard" in Claim 5, a POSITA would have been motivated to combine the teachings of Kraemer with the teachings of Njemanze. Both Kraemer and Njemanze are related to providing computer network security. Kraemer provides a probabilistic link to determine a probability of an attack on a computer system. Kraemer, Abstract. Kraemer discloses operating within computer systems that can be any operating system including a native operating system. *Id*., 8:55-59. As discussed above, Kraemer describes that security agents can mark or otherwise identify sequences of processing operations that led up to the security violation as a disallowed sequence of processing operations and can disseminate this information in real-time (e.g., upon detection), *id*., 4:22-32, which discloses "real-time status indications for operational integrity of the application." Kraemer further describes an administrator 103 who installs security agent processes 150 (including the probabilistic signature generation process 155) on the computer systems 110 that are to be protected and the security agent processes 150 are responsible for enforcing the appropriate security policies on those systems. *Id*., 7:31-36. A POSITA would have found it inherent or obvious that the determination of a probability of an attack would be output to dashboards of a computing environment, like

the ones disclosed in Njemanze. Njemanze describes displaying, on a GUI for console 16, security events and a status bar that describes the current state of the time-based events. Njemanze, ¶ 49. Specifically, the GUI for console 16 comprises a views panel that allows a user to view security events from heterogeneous sources, a navigation panel that allows a user to view system assets (as well as dashboards) in an organized manner, and a status bar that describes the current state of the time-based events. *Id*. Because Kraemer's system is configured to "rapidly identif[y] malicious attacks and prevents the spread of such attacks to other computer systems," Kraemer, 3:48-50, a POSITA would have been motivated to use Njemanze's GUI comprising different panels to display the determination of a probability of an attack on a computer system of Kraemer. A POSITA would have readily appreciated that the functionalities of Njemanze's GUI could readily and beneficially be implemented in Kraemer's system to rapidly identify malicious attacks.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "scans [being] performed … by a third party collaboration service to score the severity of an application based on the vulnerability, compliance, and patch status to affirm positive presence of a threat" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Kraemer is deemed to lack sufficient disclosure of "scans [being] performed … by a third party collaboration service to score the severity of an application based on the vulnerability, compliance, and patch status to affirm positive presence of a threat" in Claim 4, a POSITA would have also been motivated to combine the teachings of Kraemer with the teachings of Njemanze. Particularly, Kraemer describes that security agent processes 150 and management center

application 160 dynamically detect and prevent malicious attacks on the computer systems 110. *Id*., 7:26-31. And "probabilistic signature generation process 155 weights each event to determine the probability of an attack on the computer system." *Id*., 13:61-67. A POSITA would have found it obvious to incorporate a third-party service to detect and weigh events to determine the probability of an attack. Njemanze discloses that knowledge base 28 contains reference documents (e.g., in the form of web pages and/or downloadable documents) from third-party sources such as user-authored articles, third-party articles, and/or security vendors' reference material. Njemanze, ¶ 35. These documents provide a description of the threat, recommended solutions, reference information, company procedures and/or links to additional resources. *Id*. Because Kraemer's system is configured to "rapidly identif[y] malicious attacks and prevents the spread of such attacks to other computer systems," Kraemer, 3:48-50, a POSITA would have been motivated to use Njemanze's teachings of third-party sources materials in Kraemer's systems in order to better predict detect and weigh events to determine the probability of an attack and thereby to rapidly identify malicious attacks.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "a network analyzer, an integrity processor, an event correlation matrix, a risk correlation matrix, and a trust supervisor" in a native computing environment in Claim 1 or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Kraemer is deemed to lack sufficient disclosure of "a network analyzer, an integrity processor, an event correlation matrix, a risk correlation matrix, and a trust supervisor" in a native computing environment in Claim 1,

it would have been obvious to a POSITA to name the functional components or security interceptors of Kraemer's computing environment so that each functional component performs a particular function. Kraemer describes that "a system receives information from at least one security interceptor associated with at least one computer system. The information identifies details associated with a traffic flow in a computer system of the computer networking environment." Kraemer, Abstract. Kraemer's probabilistic signature generation process 155 receives information from the at least one security interceptor monitoring downloaded content and at least one network connection. *Id*., 10:29-44. It would have been obvious to a POSITA that this function could be performed by a security interceptor named "a network analyzer." Kraemer also describes that the probabilistic signature generation process 155 receives information from the at least one security interceptor monitoring an instance of at least one root-kit detection. *Id*. It would have been obvious to a POSITA that this function could be performed by a security interceptor named "an integrity processor" or "a trust supervisor." Kraemer further describes that when the probability of an attack being executed crosses a specified threshold, the probabilistic signature generation process probabilistically links the observed events to the probability that an attack is occurring, and attempts to identify a root cause. *Id*., 4:41-52. The probabilistically linking may be accomplished by examining histories of received data collected over one or more repeated attacks. *Id*. A signature can then be generated from this data, and applied as a filter at the appropriate data interface to prevent subsequent attacks. *Id*. It would have been obvious to a POSITA that the histories of received data are "an event correlation matrix" or "a risk correlation matrix." It would have been obvious to a POSITA that the above functions or matrices are for separate components, and it would have been a design choice to separate them out or have them integrated in Kraemer's system.

- For the Asserted Claims of the ’948 patent, to the extent the preamble of Claim 1 is found to be limiting, and to the extent the references charted in any of the charts in Exhibit F do not expressly disclose “a native computing environment” or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Farley is deemed to lack sufficient disclosure of “a native computing environment” in Claim 1, a POSITA would have also been motivated to combine the teachings of Farley with the teachings of Njemanze, which describes a native computing environment including a native device that receives the event information from the associated network devices or applications in the devices’ or applications’ native formats and convert (or normalize) the information to a common schema. *See* Njemanze, ¶¶ 23, 29. Both Farley and Njemanze disclose identifying and analyzing security events of computer networks. Farley is configured to provide a computer security management system that can manage security information for an entire network in “either a stand-alone or in a distributed computing environment or both.” Farley, 2:54-58, 3:27-30, 6:30-39. Njemanze is related to “provid[ing] a historical view of the event traffic” including that of the security events of computer network devices, and its method is applicable to “any computer-based or network-based security system.” Njemanze, ¶¶ 18-19, 49. A POSITA would have been motivated to use Njemanze’s teachings of applying a native computing environment to manage security information for the network of Farley.
- For the Asserted Claims of the ’948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose “scans [being] performed … by a third party collaboration service to score the severity of an application based on the vulnerability, compliance, and patch status to affirm positive presence of a threat” or similar limitations,

it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Farley is deemed to lack sufficient disclosure of "scans [being] performed … by a third party collaboration service to score the severity of an application based on the vulnerability, compliance, and patch status to affirm positive presence of a threat" in Claim 4, a POSITA would have also been motivated to combine the teachings of Farley with the teachings of Njemanze. Particularly, Farley describes that detectors 28 of intrusion detection systems scan raw network traffic or local system events for predefined patterns. Farley, 9:67-10:1. Farley determines whether the difference between the current processing time and the updated time of each correlation event exceeds a predetermined threshold by determining "whether the mature correlation events contained within the mature event list 650 have become old or stale in that no computer activity or raw events have occurred for these correlation events over some period of time." *Id*., 33:40-44. Farley also describes a database that comprises "a context or knowledge database that includes network context information, such as host vulnerability statuses, historical computer event frequency values, and network zone definitions." *Id*., 3:64-67. A POSITA would have found it obvious to incorporate a third-party service to detect the event that is potentially an attack. A POSITA would have found it obvious to incorporate a third-party service to scan raw network traffic or local system events for predefined patterns and use the database of Farley to score the severity of an application like that disclosed in Njemanze. Njemanze discloses that knowledge base 28 contains reference documents (e.g., in the form of web pages and/or downloadable documents) from third-party sources such as user-authored articles, third-party articles, and/or security vendors' reference material. Njemanze, ¶ 35. These documents provide a

description of the threat, recommended solutions, reference information, company procedures and/or links to additional resources. *Id*. Because Farley is configured to provide a computer security management system that can manage security information for an entire network in "either a stand-alone or in a distributed computing environment or both." Farley, 2:54-58, 3:27-30, 6:30-39, a POSITA would have been motivated to use Njemanze's teachings of third-party sources materials in Farley's systems in order to facilitate scanning raw network traffic or local system events to better manage security information of Farley's network.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "the plurality of runtime dashboards are a plurality of application runtime reputation score dashboards that comprise one or more of: a global view dashboard; a system configuration dashboard; a resource utilization dashboard; an application integrity dashboard; and a network activity dashboard" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Farley is deemed to lack disclosure that "the plurality of runtime dashboards are a plurality of application runtime reputation score dashboards that comprise one or more of: a global view dashboard; a system configuration dashboard; a resource utilization dashboard; an application integrity dashboard; and a network activity dashboard" in Claim 5, a POSITA would have been motivated to combine the teachings of Farley with the teachings of Njemanze. Farley describes a fusion engine displaying, on a console or the display monitor connected to the system bus, risk and rank information as messages on a console and sending updates related to mature correlation events to a console. Farley, 4:35-39, 7:66-

8:1. A POSITA would have found it inherent or obvious that various information including risk and rank information and updates related to mature correlation events would be displayed in a plurality of runtime dashboards of a console, like the ones disclosed in Njemanze. Njemanze describes displaying, on a GUI for console 16, security events and a status bar that describes the current state of the time-based events. Njemanze, ¶ 49. Specifically, the GUI for console 16 comprises a views panel that allows a user to view security events from heterogeneous sources, a navigation panel that allows a user to view system assets (as well as dashboards) in an organized manner, and a status bar that describes the current state of the time-based events. *Id*. Farley is configured to provide a computer security management system that can manage security information for an entire network in "either a stand-alone or in a distributed computing environment or both." Farley, 2:54-58, 3:27-30, 6:30-39. A POSITA would have been motivated to use Njemanze's GUI comprising different panels to display security events to better facilitate Farley's displaying of various information including risk and rank information and updates related to mature correlation events management. A POSITA would have readily appreciated that the functionalities of Njemanze's GUI could readily and beneficially be implemented in Farley's management of security information for a network.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "the scans [being] performed as per a schedule, and wherein the schedule is one or more of hourly, daily, weekly, bi-weekly, or monthly" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Farley is deemed to lack sufficient disclosure of "the scans [being] performed as per a

schedule, and wherein the schedule is one or more of hourly, daily, weekly, bi-weekly, or monthly" in Claim 4, a POSITA would have found it obvious to combine the teachings of Farley with the teachings of Marvasti. Both Farley and Marvasti disclose integrity and security events management of computer networks. Farley discloses that "detectors 28 of intrusion detection systems scan raw network traffic or local system events for predefined patterns." Farley, 9:67-10:1. Farley also determines whether the difference between the current processing time and the update time of each correlation event exceeds a predetermined threshold by determining "whether the mature correlation events contained within the mature event list 650 have become old or stale in that no computer activity or raw events have occurred for these correlation events over some period of time." *Id*., 33:40-44. Marvasti discloses that dynamic thresholding engine 402 collects, evaluates, and determines new thresholds at midnight each night, or resets the dynamic thresholds at other time intervals such as weekly, biweekly, or monthly. *See* Marvasti, 8:17-27. Because Farley's system is configured to provide a computer security management system that can manage security information for an entire network in "either a stand-alone or in a distributed computing environment or both." Farley, 2:54-58, 3:27-30, 6:30-39, a POSITA would have been motivated to use Marvasti's method or system of monitoring on a daily, weekly, biweeky, or monthly schedule to determine whether "no computer activity or raw events have occurred for these correlation events over some period of time." A POSITA would have readily appreciated the method of daily, weekly, biweekly, or monthly monitoring as taught by Marvasti to manage security information for the network of Farley.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "the real-time operations of applications are monitored continuously" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or

creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Farley is deemed to lack disclosure of "the real-time operations of applications are monitored continuously" in Claim 2, a POSITA would have been motivated to combine Farley with Datta Ray's teachings on monitoring. Both Farley and Datta Ray disclose monitoring and identifying vulnerabilities of computer networks. Farley describes determining if computer activity or raw events have occurred over some period of time. Farley, 33:40-44. It would have been obvious to a POSITA that determining if computer activity or raw events have occurred over some period of time is performed continuously as Datta Ray discloses. Datta Ray describes continuously evolving threats and vulnerabilities are monitored or state information is obtained in real-time periodically. Datta Ray, 2:39-44, 19:66-20:1 *see also* Datta Ray Provisional, pp.1, 7. Thus, a POSITA would readily use Datta Ray's method or system of continuously monitoring or periodically obtaining state information to achieve Farley's objective.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "scans [being] performed … by a third party collaboration service to score the severity of an application based on the vulnerability, compliance, and patch status to affirm positive presence of a threat" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Farley is deemed to lack sufficient disclosure of "scans [being] performed … by a third party collaboration service to score the severity of an application based on the vulnerability, compliance, and patch status to affirm positive presence of a threat" in Claim 4, a POSITA would have also been

motivated to combine the teachings of Farley with those of Datta Ray. Farley discloses that "detectors 28 of intrusion detection systems scan raw network traffic or local system events for predefined patterns." Farley, 9:67-10:1. Farley also determines whether the difference between the current processing time and the update time of each correlation event exceeds a predetermined threshold by determining "whether the mature correlation events contained within the mature event list 650 have become old or stale in that no computer activity or raw events have occurred for these correlation events over some period of time." *Id*., 33:40-44. A POSITA would have found it obvious to incorporate a third-party service to detect the event that is potentially an attack as one that Datta Ray has dicsloed. Datta Ray discloses such a service such as risk assessments and decisions with government agencies and third party collaborations. *See* Datta Ray, 1:54-58; *see also* Datta Ray Provisional, p.1. Because Farley's system is configured to provide a computer security management system that can manage security information for an entire network in "either a stand-alone or in a distributed computing environment or both," Farley, 2:54-58, 3:27-30, 6:30-39, a POSITA would have been motivated to use Datta Ray's teachings of government agencies and third-party collaborations in Farley's systems to manage security information for the network of Farley.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "the event and behavior correlation engine receives one or more of: ...system events pertaining to a processor, network, memory, ...; and system events pertaining to image profiles … from an application integrity sensor" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Farley is

deemed to lack sufficient disclosure that "the event and behavior correlation engine receives one or more of: ...system events pertaining to a processor, network, memory, ...; and system events pertaining to image profiles … from an application integrity sensor" in Claim 3, it would have been obvious to a POSITA that the system disclosed in Farley could have been implemented such that the correlation rules receive the above information when receiving correlation events detected on the target system. Farley describes that "[t]he multiple data sources can comprise sensors or detectors that monitor network traffic or individual computers or both. The sensors can comprise devices that may be referred to as intrusion detection systems (IDS)." *Id*., 3:49-52. A POSITA would have found it obvious that the various sensors or devices in Farley receive various types of events related to the system or the network. Farley describes that the correlation rules detect correlation events detected by a network-based sensor including those generated when an availability attack is seen against a target IP address followed by another event indicating that the target is no longer behaving normally and when an external network detects a duplicate internal IP address ("system events and infection profiles from a network activity sensor"); integrity attack followed by start of a new service reported from a host-based sensor ("integrity measurement and verification reports of scans from a system configuration sensor"); events indicating system resources detected on the target system itself by a host-based sensor ("system events pertaining to … storage resource consumption from a resource utilization sensor"); and events detected on the target system itself by a host-based sensor ("system events pertaining to …local execution context from an application integrity sensor"). *Id*., 20:2-5, 20-29. Farley further describes that the correlation rules detect correlation events including availability attack sweep (Multihost DoS Attack) event in which two or more different types of availability attacks originating from the same source IP address are seen against multiple target IP addresses. *Id*., 20:10-14. A POSITA would have found it inherent

or obvious that the availability attack sweep event and events indicating system resources would be "pertaining to a processor, network, memory." A POSITA would have found it obvious that one of Farley's sensors or devices such as the host-based sensor detects these events. Farley also describes that the correlation rules detect correlation events including an event generated when login failures of the same user are reported by multiple network- or host-based sensors. *Id*., 21:1-4. A POSITA would have found it inherent or obvious that the login failures are the "system events" that could "pertain[] to image profiles." And a POSITA would have found it obvious that one of Farley's sensors or devices detects these events. Because Farley's system is configured to provide a computer security management system that can manage security information for an entire network in "either a stand-alone or in a distributed computing environment or both," *Id*., 2:54-58, 3:27-30, 6:30-39, a POSITA would have found it inherent or obvious that detecting system events pertaining to a processor, network, memory and image profiles with one of the sensors or devices in Farley's systems would facilitate security management for the network of Farley.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "a network analyzer, an integrity processor, an event correlation matrix, a risk correlation matrix, and a trust supervisor" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Farley is deemed to lack sufficient disclosure of "a network analyzer, an integrity processor, an event correlation matrix, a risk correlation matrix, and a trust supervisor" in a native computing environment in Claim 1, it would have been obvious to a POSITA to name the functional components or security interceptors of Farley's computing environment so that each

functional component performs a particular function. Farley describes that "[a] security management system includes a fusion engine which 'fuses' or assembles information from multiple data sources and analyzes this information in order to detect relationships between raw events that may indicate malicious behavior and to provide an organized presentation of information to consoles without slowing down the processing performed by the data sources. The multiple data sources can comprise sensors or detectors that monitor network traffic or individual computers or both." *Id*., Abstract. It would have been obvious to a POSITA that this function is performed by a fusion engine named "a network analyzer" because it analyzes received information from the network. It would have also been obvious to a POSITA that this function is performed by a fusion engine named "an integrity processor" or "a trust supervisor" because they analyze the received information in order to detect relationships between raw events that may indicate malicious behavior and to provide an organized presentation of information to consoles without slowing down the processing performed by the data sources. Moreover, Farley describes using correlation rules with the Raw Event Classification Database to identify event patterns. *Id*., 19:65-20:1. It would have been obvious to a POSITA that the function performed by the fusion engine or the correlation rules is by referring to "an event correlation matrix" or "a risk correlation matrix" because the fusion engine detects relationships between raw events that may indicate malicious behavior and the correlation rules use the Raw Event Classification Database to identify event patterns. It would have been obvious to a POSITA that the above functions or matrices are for separate components, and it would have been a design choice to separate them out or have them integrated in Farley's system.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "the event and behavior correlation engine receives one or more of: ...system events pertaining to a processor, network, memory, ...;

and system events pertaining to image profiles … from an application integrity sensor" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Farley is deemed to lack sufficient disclosure that "the event and behavior correlation engine receives one or more of: ...system events pertaining to a processor, network, memory, ...; and system events pertaining to image profiles … from an application integrity sensor" in Claim 3, a POSITA would have been motivated to combine Farley with the teachings of U.S. Patent No. 7,904,278 ("Wilson") such that the correlation rules receive the above information when receiving other correlation events detected on the target system. Both Farley and Wilson disclose ensuring security of computer networks. As discussed above, a POSITA would have found it obvious that the various sensors or devices in Farley receive different types of events related to the system or the network. Particularly, Farley describes that the correlation rules detect correlation events including availability attack sweep (Multihost DoS Attack) event in which two or more different types of availability attacks originating from the same source IP address are seen against multiple target IP addresses. *Id*., 20:10-14. A POSITA would have found it inherent or obvious that the availability attack sweep event and events indicating system resources would be "pertaining to a processor, network, memory" as disclosed in Wilson. Wilson describes receiving system events pertaining to a processor and network. For example, Wilson describes "retriev[ing] the measurement results for a particular computer system and use those results to determine whether or not the system's integrity is intact." Wilson, 8:20-37. Wilson further describes that the Linux kernel ("event and behavior correlation engine") receives bugs or extra capabilities in the currently executing processor to provide for the opportunity to replace

individual program instructions. *Id*., 12:64-66. Wilson describes receiving system events pertaining to a memory. For example, Wilson describes that CoPilot measures the memory image of a running system and that "[t]he ability to inspect the runtime memory of the target is an improvement over file system hashes because it enables decisions about runtime state." *Id*., 4:57-65. Farley also describes that the correlation rules detect correlation events including an event generated when login failures of the same user are reported by multiple network- or host-based sensors. Farley, 21:1-4. A POSITA would have found it inherent or obvious that the login failures are the "system events" that could "pertain[] to image profiles" as disclosed in Wilson. Wilson describes receiving system events pertaining to image profiles. For example, Wilson describes that "[u]sing Tripwire enables complete integrity measurement of file system objects such as executable images or configuration files," Wilson, 3:58-60, "measuring and verifying the integrity of a computer program," *id*., 6:5-6, and measuring "the Linux Kernel's process image may include a traversal of structures in the virtual file system (VFS)." *Id*., 13:39-41. Because Farley's system is configured to provide a computer security management system that can manage security information for an entire network in "either a stand-alone or in a distributed computing environment or both," Farley, 2:54-58, 3:27-30, 6:30-39, a POSITA would have been motivated to use Wilson's teachings of retrieving measurement results for a particular computer system, receiving bugs or extra capabilities in the currently executing processor, measuring the memory image of a running system, and measuring file system objects such as executable images or configuration files in Farley's systems to manage security information for the network of Farley.

- For the Asserted Claims of the '948 patent, to extent the preamble of Claim 1 is found to be limiting, and to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "a native computing environment" or similar limitations, it would have

been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent that U.S. Patent Publication No. 2006/0020866 ("Lo") is deemed to lack sufficient disclosure of "a native computing environment" in Claim 1, a POSITA would have also been motivated to combine the teachings of Lo with the teachings of Njemanze, which describes a native computing environment including a native device that receives event information from the associated network devices or applications in the devices' or applications' native formats and convert (or normalize) the information to a common schema. *See* Njemanze, ¶¶ 23, 29. Both Lo and Njemanze disclose monitoring computer networks. Lo describes monitoring performance of network infrastructure and applications or managing software and infrastructure components based on real time deviations using computer program that simultaneously manages multiple, flexible groupings of software and infrastructure components. Lo, Abstract, ¶ 47. Lo is configured to "monitor[] performance of network infrastructure and applications by automatically identifying system variables or combinations constructed from such variables that dominate variance of system performance." *Id.*, ¶ 26. Njemanze is related to "provid[ing] a historical view of the event traffic" including that of the security events of computer network devices, and its method is applicable to "any computer-based or network-based security system." Njemanze, ¶¶ 18-19, 49. A POSITA would have been motivated to use Njemanze's teachings of applying a native computing environment to monitor performance of network infrastructure and applications for the network of Lo.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "scor[ing] the severity of an application

based on the vulnerability, compliance, and patch status to affirm positive presence of a threat" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Lo is deemed to lack sufficient disclosure of "scor[ing] the severity of an application based on the vulnerability, compliance, and patch status to affirm positive presence of a threat" in Claim 4, a POSITA would have also been motivated to combine the teachings of Lo with the teachings of Njemanze. Particularly, Lo describes "monitoring performance of network infrastructure and applications by automatically identifying system variables or combinations constructed from such variables that dominate variance of system performance" and "monitor[ing] performance of executing software applications and execution infrastructure components to detect deviations in performance." Lo, Abstract. A POSITA would have found it obvious that monitoring performance of executing software applications and execution infrastructure components to detect deviations in performance would include scoring the severity of an application based on the vulnerability, compliance, and patch status to affirm positive presence of a deviation/threat as disclosed in Njemanze. Njemanze discloses that knowledge base 28 contains reference documents (e.g., in the form of web pages and/or downloadable documents) from third-party sources such as user-authored articles, third-party articles, and/or security vendors' reference material. Njemanze, ¶ 35. These documents provide a description of the threat, recommended solutions, reference information, company procedures and/or links to additional resources. *Id*. Because Lo is configured to "monitor[] performance of network infrastructure and applications by automatically identifying system variables or combinations constructed from such variables that dominate variance of system performance," Lo, ¶ 26, a POSITA

would have been motivated to use Njemanze's teachings of third-party sources materials that provide a description of the threat, recommended solutions, reference information, company procedures and/or links to additional resources in Lo's systems in order to better monitor performance of network infrastructure and applications for the network of Lo.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "displaying, in a plurality of runtime dashboards of an administrative console of the computing environment, real-time status indications for operational integrity of the application" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Lo is deemed to lack disclosure of "displaying, in a plurality of runtime dashboards of an administrative console of the computing environment, real-time status indications for operational integrity of the application" in Claim 1 and "the plurality of runtime dashboards are a plurality of application runtime reputation score dashboards that comprise one or more of: a global view dashboard; a system configuration dashboard; a resource utilization dashboard; an application integrity dashboard; and a network activity dashboard" in Claim 5, a POSITA would have been motivated to combine the teachings of Lo with the teachings of Njemanze. Lo is related to monitoring performance of groupings of network infrastructure and applications using statistical analysis. Lo, ¶ 7. In preferred embodiments, Lo describes simultaneously manages multiple, flexible groupings of software and infrastructure components based on real time deviations from an expected normative behavioral pattern (Footprint) ("real-time status indications for operational integrity"). *Id.*, ¶ 47. Lo further describes a system of deploying modified software into an operating environment based on

deviations in its expected operating behavior. After identifying and establishing a baseline for the operating behavioral patterns (Footprint) for a group of software and infrastructure components and comparing the Footprints of the modified state with that of the original state, the system provides IT operators with a statistical metric that indicates the extent to which the new modified system matches the expected original normal patterns as defined by the baseline Footprint. *Id*., ¶ 156. Based on these outputs from the system, the IT operator is able to make a software release decision based on a statistical measure of confidence that the modified application behaves as expected. *Id*., ¶ 157. A POSITA would have found it inherent or obvious that the various system output to the IT operators would be on dashboards of a computing environment, like the ones disclosed in Njemanze. Njemanze describes displaying, on a GUI for console 16, security events and a status bar that describes the current state of the time-based events. Njemanze, ¶ 49. Specifically, the GUI for console 16 comprises a views panel that allows a user to view security events from heterogeneous sources, a navigation panel that allows a user to view system assets (as well as dashboards) in an organized manner, and a status bar that describes the current state of the time-based events. *Id*. Because Lo's system is configured to "monitor[] performance of network infrastructure and applications by automatically identifying system variables or combinations constructed from such variables that dominate variance of system performance," Lo, ¶ 26, a POSITA would have been motivated to use Njemanze's GUI comprising different panels to display the various system output to the IT operators from the computer system of Lo. A POSITA would have readily appreciated that the functionalities of Njemanze's GUI could readily and beneficially be implemented in Lo's system to better monitor performance of network infrastructure and applications by facilitating automatically identifying system variables or combinations constructed from such variables.

- For the Asserted Claims of the ’948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose “monitoring... one or more of network dialogs of the application;…resource utilization by the application” or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Lo is deemed to lack sufficient disclosure of “monitoring... one or more of network dialogs of the application;…resource utilization by the application” in Claim 1, a POSITA would have found it obvious to combine the teachings of Lo with the teachings of Marvasti. Both Lo and Marvasti disclose monitoring computer networks. Lo describes “monitoring performance of network infrastructure and applications by automatically identifying system variables or combinations constructed from such variables that dominate variance of system performance” and “monitor[ing] performance of executing software applications and execution infrastructure components to detect deviations in performance.” *Id*., Abstract. Marvasti describes choosing metrics to monitor chosen variables within each node. Marvasti, 4:59-5:3. “For example, in a network environment, suitable metrics for a router may be packets received, packets sent, packets dropped, and collisions. Similarly, a workstation may monitor CPU load, outgoing packets, and incoming packets, in the processor and NIC, respectively.” *Id*., 4:59-5:3. Because Lo’s system is configured to “monitor[] performance of network infrastructure and applications by automatically identifying system variables or combinations constructed from such variables that dominate variance of system performance,” Lo, ¶ 26, a POSITA would have been motivated to use Marvasti’s method or system of monitoring packets received, sent, and dropped, and collisions and CPU load in a network to monitor performance of network

infrastructure and applications in Lo's system. A POSITA would have readily appreciated that the method of monitoring packets received, sent, and dropped, and collisions and CPU load as taught by Marvasti could readily and beneficially facilitate monitoring performance of network infrastructure and applications in Lo.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "the scans [being] performed as per a schedule, and wherein the schedule is one or more of hourly, daily, weekly, bi-weekly, or monthly" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Lo is deemed to lack sufficient disclosure of "the scans [being] performed as per a schedule, and wherein the schedule is one or more of hourly, daily, weekly, bi-weekly, or monthly" in Claim 4, a POSITA would have found it obvious to combine the teachings of Lo with the teachings of Marvasti. Lo discloses performing data collections per a baseline period and using open source systems management tool for periodic data input. *Id*., ¶¶ 79, 106. Marvasti discloses that the dynamic thresholding engine 402 collects, evaluates, and determines new thresholds at midnight each night, or resets the dynamic thresholds at other time intervals such as weekly, biweekly, or monthly. *See* Marvasti, 8:17-27. Because Lo's system is configured to "monitor[] performance of network infrastructure and applications by automatically identifying system variables or combinations constructed from such variables that dominate variance of system performance," Lo, ¶ 26, a POSITA would have been motivated to use Marvasti's method or system of monitoring a network on a daily, weekly, biweeky, or monthly schedule to perform data collections per a baseline period or receive periodic data input. A POSITA would have readily appreciated that the method of

daily, weekly, biweekly, or monthly monitoring as taught by Marvasti could readily and beneficially be implemented to "monitor[] performance of network infrastructure and applications" for the network of Lo.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "the event and behavior correlation engine receives one or more of: ...system events pertaining to … memory, and storage resource consumption; and system events pertaining to image profiles" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Lo is deemed to lack sufficient disclosure that "the event and behavior correlation engine receives one or more of: ...system events pertaining to … memory, and storage resource consumption; and system events pertaining to image profiles" in Claim 3, it would have been obvious to a POSITA that the system disclosed in Lo could have been implemented such that the Engine of Lo receives the above information when receiving data related to the performance of the network. Lo describes an Engine correlates various moving window metrics and the baseline values of those metrics and determining if the composite different metric is within or exceeding the threshold. Lo, ¶ 89. A POSITA would have found it obvious that the Engine of Lo receives different types of events related to the system or the network and server. It would have been obvious to a POSITA that the functions of receiving different types of events can be performed by different components/sensors of the Engine. It would have been obvious to a POSITA that the above functions are for separate components, and it would have been a design choice to separate them out or have them integrated. Lo describes that the Engine receives data that describes the performance behavior of the

Managed Unit or performance of network infrastructure and applications by automatically identifying system variables, combinations constructed from such variables that dominate variance of system performance, or performance of executing software applications and execution infrastructure components to detect deviations in performance ("system events and infection profiles from a network activity sensor"), *id*., Abstract, ¶ 50; outcomes data describing historical performance and availability of the systems being managed, performance of network infrastructure and applications by automatically identifying system variables or combinations constructed from such variables that dominate variance of system performance, or performance of executing software applications and execution infrastructure components to detect deviations in performance ("integrity measurement… of scans from a system configuration sensor"), *id*., Abstract, ¶ 52; data of performance of executing software applications and execution infrastructure components to detect deviations in performance ("verification reports of scans from a system configuration sensor"), *id*., Abstract; data describing the performance behavior of the Managed Unit or the Footprint calculation of step 400 performed in the run-time of the system ("system events pertaining to processor"), *id*., ¶ 52; performance of network infrastructure and applications by automatically identifying system variables or combinations constructed from such variables that dominate variance of system performance ("system events pertaining to … network"), *id*., Abstract; and business process and application specific data as quantifiable metrics describing a particular end-user process ("system events pertaining to …local execution context from an application integrity sensor"). *Id*., ¶ 51. A POSITA would have found it inherent or obvious that the Engine also receives data pertaining to "memory, and storage resource consumption" and "pertaining to image profiles" by monitoring performance of network infrastructure and applications by automatically identifying system variables or combinations constructed from such

variables that dominate variance of system performance. *Id*., Abstract. A POSITA would have found it obvious that one of Lo's components/sensors of the Engine detects these events. Because Lo's system is configured to "monitor[] performance of network infrastructure and applications by automatically identifying system variables or combinations constructed from such variables that dominate variance of system performance," *id*., ¶ 26, a POSITA would have found it inherent or obvious that detecting system events pertaining to a network, memory, storage resource consumption, and image profiles with one of Lo's components/sensors of the Engine would facilitate monitoring network infrastructure and applications in the system of Lo.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "the event and behavior correlation engine receives one or more of: ...system events pertaining to … memory, and storage resource consumption; and system events pertaining to image profiles" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Lo is deemed to lack sufficient disclosure that "the event and behavior correlation engine receives one or more of: ...system events pertaining to … memory, and storage resource consumption; and system events pertaining to image profiles" in Claim 3, a POSITA would have been motivated to combine Lo with the teachings of Wilson such that the Engine of Lo receives the above information when receiving data related to the performance of the network. Both Lo and Wilson disclose ensuring security of computer networks. As discussed above, it would have been obvious to a POSITA that Lo's functions of receiving different types of events can be performed by different components/sensors of the Engine. It would have been

obvious to a POSITA that the above functions are for separate components, and it would have been a design choice to separate them out or have them integrated in Lo's system. As also discussed above, a POSITA would have found it inherent or obvious that Lo's Engine receiving data pertaining to "memory, and storage resource consumption" and "pertaining to image profiles" by monitoring performance of network infrastructure and applications by automatically identifying system variables or combinations constructed from such variables that dominate variance of system performance, as disclosed in Wilson, would facilitate monitoring network infrastructure and applications in the system of Lo. Particularly, Wilson describes receiving system events pertaining to a memory. For example, Wilson describes that CoPilot measures the memory image of a running system and that "[t]he ability to inspect the runtime memory of the target is an improvement over file system hashes because it enables decisions about runtime state." Wilson, 4:57-65. Wilson describes receiving system events pertaining to storage resource consumption. For example, Wilson describes that a measurement agent (MA) collects measurement data about a target of measurement (T) by accessing to T's resources and be able to hold the measurement data until needed. *Id*., 2:4-11. Wilson describes receiving system events pertaining to image profiles. For example, Wilson describes that "[u]sing Tripwire enables complete integrity measurement of file system objects such as executable images or configuration files," *id*., 3:58-60, "measuring and verifying the integrity of a computer program," *id*., 6:5-6, and measuring "the Linux Kernel's process image may include a traversal of structures in the virtual file system (VFS)." *Id*., 13:39-41. Because Lo's system is configured to "monitor[] performance of network infrastructure and applications by automatically identifying system variables or combinations constructed from such variables that dominate variance of system performance." Lo, ¶ 26, a POSITA would have been motivated to use Wilson's teachings of inspecting runtime memory of the target,

collecting measurement data about a target of measurement (T) by accessing to T's resources, and measuring file system objects such as executable images or configuration files using one of Lo's components/sensors of the Engine to facilitate monitoring network infrastructure and applications in the system of Lo.

- For the Asserted Claims of the '948 patent, to the extent the preamble of Claim 1 is found to be limiting, and to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "a native computing environment" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent U.S. Patent No. 9,286,469 ("Uner") is deemed to lack sufficient disclosure of "a native computing environment" in Claim 1, a POSITA would have also been motivated to combine the teachings of Uner with the teachings of Njemanze, which describes a native computing environment including a native device that receives the event information from the associated network devices or applications in the devices' or applications' native formats and convert (or normalize) the information to a common schema. *See* Njemanze, ¶¶ 23, 29. Both Uner and Njemanze disclose preventing security attack of computer networks. Uner describes monitoring and taking actions, by a secure supervisory control and a system security monitor, with respect to untrusted applications and the real-time operating system of Secure SCADA elements (SSEs) on the operating system of SSEs that "comprise[s] any suitable OS, for example, Linux, Windows®, a Real-time Operating System (RTOS), or a simple execution loop." Uner, Abstract, 5:61-63. Uner is configured to resist malicious code from tampering with or otherwise exploiting a SCADA system to ensure a secure SCADA system. *Id*., 1:13-17. Njemanze is related to "provid[ing] a historical view of the event traffic" including that of

the security events of computer network devices, and its method is applicable to "any computer-based or network-based security system." Njemanze, ¶¶ 18-19, 49. A POSITA would have been motivated to use Njemanze's teachings of applying a native computing environment to monitor security events to ensure a secured SCADA system in Uner.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "scans [being] performed … by a third party collaboration service to score the severity of an application based on the vulnerability, compliance, and patch status to affirm positive presence of a threat" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Uner is deemed to lack sufficient disclosure of "scans [being] performed … by a third party collaboration service to score the severity of an application based on the vulnerability, compliance, and patch status to affirm positive presence of a threat" in Claim 4, a POSITA would have also been motivated to combine the teachings of Uner with the teachings of Njemanze. Particularly, Uner describes scoring the application, process or task as "trusted" if close monitoring with no suspicious activity over a period of time, Uner, 5:37-39, generating higher assurance level for trusted applications, *id*., 5:52-55, or creating interpret policies using a threat library. *Id*., 7:63-8:2. A POSITA would have found it obvious to incorporate a third-party service to monitor the application, process or task to detect malicious attacks or vulnerabilities. Njemanze discloses that knowledge base 28 contains reference documents (e.g., in the form of web pages and/or downloadable documents) from third-party sources such as user-authored articles, third-party articles, and/or security vendors' reference material. Njemanze, ¶ 35. These documents provide a description of the threat,

recommended solutions, reference information, company procedures and/or links to additional resources. *Id*. Because Uner is configured to resist malicious code from tampering with or otherwise exploiting a SCADA system to ensure a secure SCADA system. Uner, 1:13-17, a POSITA would have been motivated to use Njemanze's teachings of third-party sources documents in Uner's systems in order to ensure a secured SCADA system.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "the plurality of runtime dashboards are a plurality of application runtime reputation score dashboards that comprise one or more of: a global view dashboard; a system configuration dashboard; a resource utilization dashboard; an application integrity dashboard; and a network activity dashboard" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Uner is deemed to lack disclosure that "the plurality of runtime dashboards are a plurality of application runtime reputation score dashboards that comprise one or more of: a global view dashboard; a system configuration dashboard; a resource utilization dashboard; an application integrity dashboard; and a network activity dashboard" in Claim 5, a POSITA would have been motivated to combine the teachings of Uner with the teachings of Njemanze. Uner describes that a Human-Machine Interface ("HMI") of SCADA systems presents process data to a human operator. *Id*., 1:38-48. A POSITA would have found it inherent or obvious that various process data would be displayed in a plurality of runtime dashboards of a console, like the ones disclosed in Njemanze. Njemanze describes displaying, on a GUI for console 16, security events and a status bar that describes the

current state of the time-based events. Njemanze, ¶ 49. Specifically, the GUI for console 16 comprises a views panel that allows a user to view security events from heterogeneous sources, a navigation panel that allows a user to view system assets (as well as dashboards) in an organized manner, and a status bar that describes the current state of the time-based events. *Id*. Uner is configured to resist malicious code from tampering with or otherwise exploiting a SCADA system to ensure a secure SCADA system. Uner, 1:13-17. A POSITA would have been motivated to use Njemanze's GUI comprising different panels to display security events to better facilitate Uner's displaying of various process data to a human operator. A POSITA would have readily appreciated that the functionalities of Njemanze's GUI could readily and beneficially be implemented in Uner's SCADA system to ensure security.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "monitoring... one or more of network dialogs of the application; … a runtime configuration of the application, resource utilization by the application" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Uner is deemed to lack sufficient disclosure of "monitoring... one or more of network dialogs of the application; … a runtime configuration of the application, resource utilization by the application" in Claim 1, a POSITA would have found it obvious to combine the teachings of Uner with the teachings of Marvasti. Both Uner and Marvasti disclose preventing security attack of computer networks. Uner describes a system security monitor that "monitors and takes appropriate action with respect to untrusted applications that may impact an SSE." *Id*., Abstract. Uner further describes that the system security

monitor "may observe the operations of the application, process or task for any activity that violates any of the security rules in the monitor." Uner, 4:49-53. Marvasti describes choosing metrics to monitor chosen variables within each node. Marvasti, 4:59-5:3. In a network environment, "suitable metrics for a router may be packets received, packets sent, packets dropped, and collisions. *Id*. Similarly, a workstation may monitor CPU load, outgoing packets, and incoming packets, in the processor and NIC, respectively. *Id*. The monitoring includes capturing and weighing fingerprints of the relevant nodes, including software, within a complex system at various time intervals prior to the occurrence of the abnormality. *Id*., Abstract. Because Uner's system is configured to resist malicious code from tampering with or otherwise exploiting a SCADA system to ensure a secure SCADA system, Uner, 1:13-17, a POSITA would have been motivated to use Marvasti's method or system of monitoring packets received, packets sent, packets dropped, and collisions and CPU load as well as capturing and weighing fingerprints of the relevant nodes, including software in a network to monitor security events to ensure a secured SCADA system in Uner.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "the scans [being] performed as per a schedule, and wherein the schedule is one or more of hourly, daily, weekly, bi-weekly, or monthly" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Uner is deemed to lack sufficient disclosure of "the scans [being] performed as per a schedule, and wherein the schedule is one or more of hourly, daily, weekly, bi-weekly, or monthly" in Claim 4, a POSITA would have found it obvious to combine the teachings of

Uner with the teachings of Marvasti. Uner discloses close monitoring of suspicious activity over a period of time and no suspicious activities during the monitored time period can lead to the application, process or task becoming "trusted." *Id*., 5:37-39. Marvasti discloses that the dynamic thresholding engine 402 collects, evaluates, and determines new thresholds at midnight each night, or resets the dynamic thresholds at other time intervals such as weekly, biweekly, or monthly. *See* Marvasti, 8:17-27. Because Uner's system is configured to resist malicious code from tampering with or otherwise exploiting a SCADA system to ensure a secure SCADA system, Uner, 1:13-17, a POSITA would have been motivated to use Marvasti's method or system of monitoring on a daily, weekly, biweeky, or monthly schedule to closely monitor suspicious activity in the application, process or task over a period of time. A POSITA would have readily appreciated that the method of daily, weekly, biweekly, or monthly monitoring as taught by Marvasti could readily and beneficially be implemented to monitor security events to ensure a secured SCADA system in Uner.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "correlating… threat classifications based on the temporal sequence of" the events in Claim 1 or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Uner is deemed to lack sufficient disclosure of "correlating… threat classifications based on the temporal sequence of" the events in Claim 1, a POSITA would have found it obvious to combine the teachings of Uner with the teachings of Farley. Both Uner and Farley disclose ensuring security of computer networks. Uner describes that the system security server 220 may include a threat

library 226 containing a list of known threats and a link to an appropriate policy or policies in the policy library 224. Uner, 7:63-8:2. Uner further maps the violation to POSIX or C Standard Library errors. *Id*., 9:62-67. Farley describes using correlation rules with the raw event classification database to identify event patterns that indicate either malicious or nonmalicious activity. *See* Farley, 19:65-20:1. Farley further describes that a fusion engine can utilize the raw event classification database and the knowledge database to assess risks and determine ranks of real-time raw events. Particularly, Farley's raw event classification database can permit the fusion engine to classify raw computer events while the knowledge database can permit the fusion engine to rank and evaluate the risk of a raw computer event based upon the context of the raw computer event. *Id*., 4:44-58. Farley also correlates threat classifications based on temporal sequence of the generated event. For example, Farley determines whether the raw event (integrity attack) being considered as an inbound attack is an integrity attack and occurs earlier than at least one event in the outbound attacks list of the current correlation event. Farley, 30:27-42. Farley generates an Attack From Attacked Host event when an Integrity attack is seen against a host followed by a Confidentiality, Integrity, or Availability attack originating from that host. *Id*. When the controller 655 loads a correlation rule 620 such as the AFAH rule during system initialization, it associates the rule with all of the event types included in the event categories of interest to the rule. *Id*. Because Uner's system is configured to resist malicious code from tampering with or otherwise exploiting a SCADA system to ensure a secure SCADA system, Uner, 1:13-17, a POSITA would have been motivated to use Farley's method or system to identify event patterns that indicate either malicious or nonmalicious activity using correlation rules and the raw event classification database. A POSITA would have readily appreciated that Farley's method could readily and beneficially be implemented in Uner's monitoring system to ensure a secured SCADA system in Uner.

- For the Asserted Claims of the ’948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose “the real-time operations of applications are monitored continuously”” or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Uner is deemed to lack disclosure of “the real-time operations of applications are monitored continuously” in Claim 2, a POSITA would have been motivated to combine Uner with Datta Ray’s teachings on monitoring. Uner describes close monitoring suspicious activity over a period of time, and no such an activity can lead to the application, process or task becoming “trusted.” *Id*., 5:37-39. It would have been obvious to a POSITA that monitoring over a period of time is performed continuously to determining that there is no suspicious activity. Datta Ray describes continuously evolving threats and vulnerabilities are monitored or state information is obtained in real-time periodically. Datta Ray, 2:39-44, 19:66-20:1; *see also* Datta Ray Provisional, pp.1, 7. Thus, a POSITA would readily use Datta Ray’s method or system of continuously monitoring or periodically obtaining state information to achieve Uner’s objective.
- For the Asserted Claims of the ’948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose “scans [being] performed … by a third party collaboration service to score the severity of an application” or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Uner is deemed to lack sufficient disclosure of “scans [being] performed … by a third party collaboration service to score

the severity of an application" in Claim 4, a POSITA would have also been motivated to combine the teachings of Uner with the teachings of Datta Ray. Both Uner and Datta Ray disclose monitoring and identifying malicious attacks or vulnerabilities of computer networks. Uner describes scoring the application, process or task as "trusted" if close monitoring with no suspicious activity over a period of time, Uner, 5:37-39, generating higher assurance level for trusted applications, *id*., 5:52-55, or creating interpret policies using a threat library. *Id*., 7:63-8:2. A POSITA would have found it obvious to incorporate a third-party service to monitor the application, process or task to detect malicious attacks or vulnerabilities. Datta Ray discloses such a service such as risk assessments and decisions with government agencies and third party collaborations. *See* Datta Ray, 1:54-58; *see also* Datta Ray Provisional, p.1. Because Uner is configured to resist malicious code from tampering with or otherwise exploiting a SCADA system to ensure a secure SCADA system, Uner, 1:13-17, a POSITA would have been motivated to use Datta Ray's teachings of government agencies and third party collaborations in Uner's systems in order to ensure a secured SCADA system.

- For the Asserted Claims of the '948 patent, to the extent the preamble of Claim 1 is found to be limiting, and to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "operational integrity of an application" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent U.S. Publication No. 2009/0328222 ("Helman") is deemed to lack sufficient disclosure of "operational integrity of an application" in Claim 1, it would have been obvious to a POSITA that the system disclosed in Helman could have been implemented to provide "real-time operational integrity of an

application." Helman discloses the "integrity of a host computer" Helman, ¶ 103. It would have been obvious to a POSITA that Helman's "host" can be a client device. *See* Helman, Fig. 1 (item 108). And it would have been obvious to a POSITA that one or more applications are running on the host 108 of Helman, and the integrity of a host includes the integrity of the one or more applications running on that host.

- For the Asserted Claims of the '948 patent, to the extent the preamble of Claim 1 is found to be limiting, and to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "a native computing environment" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Helman is deemed to lack sufficient disclosure of "a native computing environment" in Claim 1, a POSITA would have also been motivated to combine the teachings of Helman with the teachings of U.S. Patent Publication No. 2011/0145711 ("Njemanze"), which describes a native computing environment including a native device that receives the event information from the associated network devices or applications in the devices' or applications' native formats and convert (or normalize) the information to a common schema. *See* Njemanze, ¶¶ 23, 29. Both Helman and Njemanze disclose monitoring and analyzing security events of computer networks. Helman is related to an enterprise security assessment sharing ("ESAS") system that detects and responds to attacks on an enterprise network and security incidents. Helman, Abstract. Helman describes collecting information about an object of interest in the environment such as a computer, user, service (e.g., one that is accessed via a website), data, or the enterprise as a whole. *Id.*, ¶ 33. A POSITA would have understood that an enterprise computing environment can be a native computing environment such as the one

disclosed by Njemanze. Njemanze is related to "provid[ing] a historical view of the event traffic" including that of the security events of computer network devices and its method is applicable to "any computer-based or network-based security system." Njemanze, ¶¶ 18-19, 49. Njemanze describes that agents 12 can collect events from any source that produces event logs or messages and can operate at the native device. *Id*., ¶ 23. The agents 12 also receive the event information from the associated network devices or applications in the devices' or applications' native formats and convert (or normalize) the information to a common schema. *Id*., ¶ 29. Because Helman is configured to detect security incidents and correlate enterprise-wide security data and to provide effective common management across all the enterprise security product islands, Helman, ¶ 3, a POSITA would have been motivated to use Njemanze's teachings of applying a native computing environment to monitor the security events of a network of Helman in order to correlate enterprise-wide security data and provide effective common management across all the network of Helman's.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "scans [being] performed … by a third party collaboration service to score the severity of an application based on the vulnerability, compliance, and patch status to affirm positive presence of a threat" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Helman is deemed to lack sufficient disclosure of "scans [being] performed … by a third party collaboration service to score the severity of an application based on the vulnerability, compliance, and patch status to affirm positive presence of a threat" in Claim 4, a POSITA would have also been

motivated to combine the teachings of Helman with the teachings of Njemanze. Particularly, Helman discusses that the Microsoft products are just examples of approaches for implementing the noted software update services endpoints and operations manager endpoints. *Id*., ¶ 50. A POSITA would have found it obvious that such software update services endpoints and operations manager endpoints can be implemented using third-party products like those in Njemanze. Njemanze discloses that knowledge base 28 contains reference documents (e.g., in the form of web pages and/or downloadable documents) from third-party sources such as user-authored articles, third-party articles, and/or security vendors' reference material. Njemanze, ¶ 35. These documents provide a description of the threat, recommended solutions, reference information, company procedures and/or links to additional resources. *Id*. Because Helman is configured to detect security incidents and correlate enterprise-wide security data and to provide effective common management across all the enterprise security product islands, Helman, ¶ 3, a POSITA would have been motivated to use Njemanze's teachings of third-party sources documents in Helman's systems in order to better detect security incidents and correlate enterprise-wide security data and to provide effective common management across its network.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "the plurality of runtime dashboards are a plurality of application runtime reputation score dashboards that comprise one or more of: a global view dashboard; a system configuration dashboard; a resource utilization dashboard; an application integrity dashboard; and a network activity dashboard" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Helman is deemed

to lack disclosure that "the plurality of runtime dashboards are a plurality of application runtime reputation score dashboards that comprise one or more of: a global view dashboard; a system configuration dashboard; a resource utilization dashboard; an application integrity dashboard; and a network activity dashboard" in Claim 5, a POSITA would have been motivated to combine the teachings of Helman with the teachings of Njemanze. Helman describes that the central server can provide a GUI that gives an administrator a comprehensive view of the history and current status of an enterprise as a whole, and of each endpoint. *Id.*, ¶¶ 59, 123, Figs. 11-18. A POSITA would have found it obvious that the information presentation forms such as the graphical displays would be in the form of dashboards, like the ones disclosed in Njemanze. Njemanze describes that the GUI for console 16 comprises a views panel that allows a user to view security events from heterogeneous sources, a navigation panel that allows a user to view system assets (as well as dashboards) in an organized manner, and a status bar that describes the current state of the time-based events. Njemanze, ¶ 49. Because Helman is configured to detect security incidents and correlate enterprise-wide security data and to provide effective common management across all the enterprise security product islands, Helman, ¶ 3, a POSITA would have been motivated to use Njemanze's GUI comprising different panels to display the comprehensive view on the GUI of Helman. A POSITA would have readily appreciated that the functionalities of Njemanze's GUI could readily and beneficially be implemented in Helman's system in order to better provide effective common management across its network.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "the scans [being] performed as per a schedule, and wherein the schedule is one or more of hourly, daily, weekly, bi-weekly, or monthly" or similar limitations, it would have been obvious to combine the teachings of

any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Helman is deemed to lack sufficient disclosure of "the scans [being] performed as per a schedule, and wherein the schedule is one or more of hourly, daily, weekly, bi-weekly, or monthly" in Claim 4, a POSITA would have found it obvious to combine the teachings of Helman with the teachings of U.S. 7,467,067 ("Marvasti"). Both Helman and Marvasti disclose monitoring and alerting security events of computer networks. Helman discloses that endpoints can operate, e.g., antivirus scan, according to a schedule. *Id*., ¶ 96. Marvasti discloses that the dynamic thresholding engine 402 collects, evaluates, and determines new thresholds at midnight each night, or resets the dynamic thresholds at other time intervals such as weekly, biweekly, or monthly. *See* Marvasti, 8:17-27. Because Helman is configured to detect security incidents and correlate enterprise-wide security data and to provide effective common management across all the enterprise security product islands, Helman, ¶ 3, a POSITA would have been motivated to use Marvasti's method or system of monitoring on a daily, weekly, biweeky, or monthly schedule to better monitor and correlate enterprise-wide security data and provide effective common management across all the network of Helman's. A POSITA would have readily appreciated that the method of daily, weekly, biweekly, or monthly monitoring as taught by Marvasti could readily and beneficially be implemented in Helman's security system.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose "correlating… threat classifications based on the temporal sequence of" the events in Claim 1 or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the

teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Helman is deemed to lack sufficient disclosure of "correlating… threat classifications based on the temporal sequence of" the events in Claim 1, a POSITA would have found it obvious to combine the teachings of Helman with the teachings of Farley. Both Helman and Farley disclose ensuring security of computer networks. Helman explains that security product endpoints can log data about events that occur in their respective areas of interest. *Id*., ¶¶ 67-68, Fig. 4. Further, Helman explains that correlation rules and complexity reduction operations can be applied to this logged data to generate security assessments. *Id*., ¶ 60. Also, Helman discloses that these security assessments can contain fields including "category" and "severity." *Id*., ¶¶ 64-65; *see also id*., Table 2 ("Category" and "Severity" fields of a security assessment). Further, Helman explains that the logged data can include events such as failed logins, and by indicating that a security assessment can contextualize such an event in time (e.g., that such a failed login is the sixth in the last ten minutes). *Id*., ¶ 60, Fig. 3. Farley describes using correlation rules with the raw event classification database to identify event patterns that indicate either malicious or nonmalicious activity. *See* Farley, 19:65-20:1. Farley further describes that a fusion engine can utilize the raw event classification database and the knowledge database to assess risks and determine ranks of real-time raw events. Particularly, Farley's raw event classification database can permit the fusion engine to classify raw computer events while the knowledge database can permit the fusion engine to rank and evaluate the risk of a raw computer event based upon the context of the raw computer event. *Id*., 4:44-58. Farley also correlates threat classifications based on temporal sequence of the generated event. For example, Farley determines whether the raw event (integrity attack) being considered as an inbound attack is an integrity attack and occurs earlier than at least one event in the outbound attacks list of the current correlation

event. Farley, 30:27-42. Farley generates an Attack From Attacked Host event when an Integrity attack is seen against a host followed by a Confidentiality, Integrity, or Availability attack originating from that host. *Id*. When the controller 655 loads a correlation rule 620 such as the AFAH rule during system initialization, it associates the rule with all of the event types included in the event categories of interest to the rule. *Id*. Because Helman is configured to detect security incidents and correlate enterprise-wide security data and to provide effective common management across all the enterprise security product islands, Helman, ¶ 3, a POSITA would have been motivated to use Farley's method or system to categorize security assessments which contextualizes events in time. A POSITA would have readily appreciated that Farley's method could readily and beneficially be implemented in Helman's system to provide effective common management.

- For the Asserted Claims of the '948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose a "native computing environment which includes a network analyzer, an integrity processor, an event correlation matrix, a risk correlation matrix, and a trust supervisor" or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Pan, which discloses a "network analyzer" (e.g. Performance Evaluator), an "integrity processor" (e.g. Tracer Module), an event correlation matrix (e.g. Comparison Module), a risk correlation matrix (e.g. Comparison Module), and a trust supervisor (e.g. Root Cause Output Module), as well as Datta Ray, Farley, Gill, Helman, Kraemer, Lo, Marvasti, Njemanze, Uner, Wilson, Das, Overcash II, Sreenivas, Leone, Williams, Bose, OfficeScan. A person of skill in the art would be

motivated to combine these references because they each disclose a native computing environment with components to monitor operational integrity of that environment and are thus in the same general field. It is the nature of those of skill in the art of computer operational integrity to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems accomplish integrity monitoring and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how other solutions in this field utilize components of a native computing environment. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose “correlating”, or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Williams which expressly discloses correlating threat classifications, as well as Datta Ray, Farley, Gill, Helman, Kraemer, Lo, Marvasti, Njemanze, Uner, Wilson, Das, Overcash II, Sreenivas, Leone, Williams, Bose, OfficeScan. A person of skill in the art would be motivated to combine these references because they each disclose methods of determining operational integrity and are thus in the same general field. It is the nature of those of skill in the art of integrity monitoring to

monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems accomplish integrity monitoring and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how other solutions in this field analyze events. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’948 patent, to the extent the references charted in any of the charts in Exhibit F do not expressly disclose “behavior based events”, or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Pan which discloses generating real-time behavior based events (e.g. “fingerprints” generated from a trace file) for determining real-time operational integrity of the application executing on the native computing environment, as well as well as Datta Ray, Farley, Gill, Helman, Kraemer, Lo, Marvasti, Njemanze, Uner, Wilson, Das, Overcash II, Sreenivas, Leone, Williams, Bose, OfficeScan. A person of skill in the art would be motivated to combine these references because they each disclose methods of determining operational integrity and are thus in the same general field. It is the nature of those of skill in the art of integrity monitoring to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products

and systems accomplish integrity monitoring and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how other solutions in this field analyze events. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

Consistent with the exemplary motivations to combine each of the references discussed above, the below table provides a summary of the exemplary combinations of references rendering obvious to a POSA, with or without a POSA's knowledge (including the AAPA and general state of the art), skill and/or creativity, the Asserted Claims of the '948 Patent.

Reference	**Reference Alone or in Combination With One or More of the Following**
Datta Ray	Farley, Gill, Helman, Kraemer, Lo, Marvasti, Njemanze, Uner, Wilson, Das, Overcash, Sreenivas, Leone, Pan, Williams, Bose, OfficeScan
Farley	Datta Ray, Gill, Helman, Kraemer, Lo, Marvasti, Njemanze, Uner, Wilson, Das, Overcash, Sreenivas, Leone, Pan, Williams, Bose, OfficeScan
Gill	Datta Ray, Farley, Helman, Kraemer, Lo, Marvasti, Njemanze, Uner, Wilson, Das, Overcash II, Sreenivas, Leone, Pan, Williams, Bose, OfficeScan
Helman	Datta Ray, Farley, Gill, Kraemer, Lo, Marvasti, Njemanze, Uner, Wilson, Das, Overcash II, Sreenivas, Leone, Pan, Williams, Bose, OfficeScan
Kraemer	Datta Ray, Farley, Gill, Helman, Lo, Marvasti, Njemanze, Uner, Wilson, Das, Overcash II, Sreenivas, Leone, Pan, Williams, Bose, OfficeScan
Lo	Datta Ray, Farley, Gill, Helman, Kraemer, Marvasti, Njemanze, Uner, Wilson, Das, Overcash II, Sreenivas, Leone, Pan, Williams, Bose, OfficeScan
Marvasti	Datta Ray, Farley, Gill, Helman, Kraemer, Lo, Njemanze, Uner, Wilson, Das, Overcash II, Sreenivas, Leone, Pan, Williams, Bose, OfficeScan

Reference	Reference Alone or in Combination With One or More of the Following
Njemanze	Datta Ray, Farley, Gill, Helman, Kraemer, Lo, Marvasti, Uner, Wilson, Das, Overcash II, Sreenivas, Leone, Pan, Williams, Bose, OfficeScan
Uner	Datta Ray, Farley, Gill, Helman, Kraemer, Lo, Marvasti, Njemanze, Wilson, Das, Overcash II, Sreenivas, Leone, Pan, Williams, Bose, OfficeScan
Wilson	Datta Ray, Farley, Gill, Helman, Kraemer, Lo, Marvasti, Njemanze, Uner, Das, Overcash II, Sreenivas, Leone, Pan, Williams, Bose, OfficeScan
Das	Datta Ray, Farley, Gill, Helman, Kraemer, Lo, Marvasti, Njemanze, Uner, Wilson, Overcash II, Sreenivas, Leone, Pan, Williams, Bose, OfficeScan
Overcash II	Datta Ray, Farley, Gill, Helman, Kraemer, Lo, Marvasti, Njemanze, Uner, Wilson, Das, Sreenivas, Leone, Pan, Williams, Bose, OfficeScan
Sreenivas	Datta Ray, Farley, Gill, Helman, Kraemer, Lo, Marvasti, Njemanze, Uner, Wilson, Das, Overcash II, Leone, Pan, Williams, Bose, OfficeScan
Leone	Datta Ray, Farley, Gill, Helman, Kraemer, Lo, Marvasti, Njemanze, Uner, Wilson, Das, Overcash II, Sreenivas, Pan, Williams, Bose, OfficeScan
Pan	Datta Ray, Farley, Gill, Helman, Kraemer, Lo, Marvasti, Njemanze, Uner, Wilson, Das, Overcash II, Sreenivas, Leone, Williams, Bose, OfficeScan
Williams	Datta Ray, Farley, Gill, Helman, Kraemer, Lo, Marvasti, Njemanze, Uner, Wilson, Das, Overcash II, Sreenivas, Leone, Pan, Bose, OfficeScan
Bose	Datta Ray, Farley, Gill, Helman, Kraemer, Lo, Marvasti, Njemanze, Uner, Wilson, Das, Overcash II, Sreenivas, Leone, Pan, Williams, OfficeScan
OfficeScan	Datta Ray, Farley, Gill, Helman, Kraemer, Lo, Marvasti, Njemanze, Uner, Wilson, Das, Overcash II, Sreenivas, Leone, Pan, Williams, Bose

Additionally, Defendant incorporates by reference each of the obviousness combinations and motivations to combine presented in the petition for the following *inter partes* review proceeding(s) (IPRs) and any additional petitions for IPRs that may be filed challenging the patentability of the ’948 patent:

- IPR2023-00705
- IPR2023-00800

7. ’616 Patent

To the extent that Plaintiff contends that any of the references identified as anticipatory references fail to disclose one or more elements, those anticipatory references render the Asserted Claims obvious, either alone or in combination with other references. Defendant contends that one of skill in the art, at the time the alleged inventions were made, would have been motivated to combine the references disclosed herein in such a way as to reach the alleged inventions, as described in further detail below. Defendant presently believes a reasonable basis exists that each of the claims asserted against it would have been obvious to one of ordinary skill in the art at the time of the alleged invention.

Below, Defendant identifies their current bases for obviousness. As seen below, Defendant presently asserts certain combinations of prior art reference(s) with or without the knowledge of one of ordinary skill in the art. Defendant offers these combinations without conceding that the reference(s) standing alone are insufficient to invalidate the relevant claims. Defendant specifically reserves their rights to combine any of its anticipatory references or any combinations of references with the knowledge of ordinary skill in the art to rebut a contention of Plaintiff that such reference(s) standing alone do not disclose every element of the Asserted Claims.

Defendant lists below prior art references and combinations of prior art references now known to Defendant, which Defendant contends render at least obvious the ’616 Asserted Claims. The corresponding charts in Exhibit G for each prior art reference identifies where and how each alleged item of prior art meets one or more limitations of the Asserted Claims. Pursuant to Patent Local Rule 3-6, Defendant expressly reserves the right to supplement these charts to provide additional details about obviousness should the Court’s claim construction require, or should Plaintiff serve adequate Infringement Contentions that better serve to notify Defendant which elements of their products allegedly correspond with the claim elements. Defendant’s contention that the references in this section in various combinations render obvious the ’616 Asserted Claims

under 35 U.S.C. § 103 is in no way an admission or suggestion that each reference does not independently anticipate or render obvious the ’616 Asserted Claims under 35 U.S.C. § 102. These combinations would have been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field.

It would been obvious to combine each of the references discussed below because they are analogous art generally directed to computer network security. The technology described in the ’616 Patent was widely known before the alleged priority date of the ’616 Patent and provided numerous motivations to combine the prior art references and systems as exemplified below. The ’616 Patent combines these components according to their known, predictable and intended uses, just as they were and/or would have been combined by a POSA in the prior art. None of the combinations presents an unexpected result, a counterintuitive combination, or a technological challenge; and no such description exists in the common specification of the asserted patents.

Accordingly, the prior art cited herein would have been readily combined according to methods known to those of ordinary skill within the field of computer network security to yield predictable results. The substitution of one element in this system could have been predictably achieved by one of ordinary skill at the time of the alleged invention. One of ordinary skill in the art would have been able to select appropriate attributes of one for inclusion in another. Those of ordinary skill in the art could have employed known techniques to improve similar prior art devices in the same way as claimed in the ’616 Patent. Thus, the combinations all have well defined, predictable, expectation of success.

- For the Asserted Claims of the ’616 patent, to the extent the references charted in any of the charts in Exhibit G do not expressly disclose sending and receiving “a dynamic context including … applications executing on the monitored device at runtime”, it would have been obvious to combine the teachings of any of these charted references with the

knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A For example, to the extent U.S. Patent No. 9,111,088 ("Ghai") is deemed to lack sufficient disclosure of sending and receiving "a dynamic context including … applications executing on the monitored device at runtime" in Claim 1, a POSITA would have also been motivated to combine the teachings of Ghai with the teachings of U.S. Patent No. 7,934,253 ("Overcash"). During prosecution, Applicants stated that Claim 1 of the '616 patent is directed to analyses of actions of an application executing on a monitored device at runtime which the prior art does not allegedly disclose. RCE (April 14, 2014) at 15. Applicants also stated that Claim 1 requires that an endpoint trust agent on the monitored device send a dynamic context and that a trust orchestration server receive the sent dynamic context. *Id*. at 16. Applicants also characterized Claim 1 as configuring to detect infected systems based on observed malicious behaviors of a monitored endpoint device and applications running thereon and perform remediation actions. Response to Office Action (Oct. 1, 2013) at 21-23. The '616 Patent describes that with the trend that many enterprises and organizations are moving their IT infrastructures to cloud computing environments, integrity of enterprise applications and mission critical applications executed in the cloud is critical. '616 Pat, 1:45-2:5. The patent suggests that the applications be executed in the clouds that include end point devices. *Id*., 1:55-56. The patent acknowledges that the proliferation of applications (business, social networking and gaming software) in Enterprise and cloud computing ecosystems make the applications more vulnerable to attacks and that "[a]pplication runtime operational integrity is a critical factor in building end-to-end trust from device to service." *Id*., 2:14-19. Thus, monitoring the device operating in the cloud computing environment and the applications running thereon were known in the industry. The '616 Patent confirms this by describing the current

security appliances such as network firewalls, intrusion detection/prevention systems, and antivirus and network-based integrity measurement and verification services for the cloud and the application running thereon. *Id*., 2:24-30. Indeed, sending and receiving a dynamic context including endpoint events and actions of the monitored device and applications executing on the monitored device were well known. For example, Overcash describes that a group of enterprises within a common industry segment, such as financial enterprise ("monitored device"), may agree to send security events including the execution of web applications within their computer networks to a central security manager, Overcash, 51:58-52:2, and a central security manager of a computer network 106 receives security events within enterprises' computer networks. *Id*. In this way, a collaborative detection module 206 in the computer network 106 uses this information to detect attacks against a web application, particularly monitoring and identifying abnormal activities against the web application. *Id*., 9:24-39; Fig. 1. Datta Ray also describes that monitored and controlled elements (MCE) are monitored and analyzed. Datta Ray, 15:1-7; *see also* Datta Ray Provisional, p. 10. Particularly, the MCE sends dynamic context including messages relevant to that MCE to all subscribing MCEs within the security system. *Id*. Each of the MCE has its own security and business risk analysis engine (SBRAE), and the SBRAE receives the dynamic context from the monitored MCE. *Id*. Jaeger also describes "analytic integrity" software of a monitored computer system sending runtime information for application 22A, and loaded program code for application 22A, to remote computer system 24. Jaeger, Fig. 1, 11:37-57. Both Ghai and Overcash disclose facilitating computer network security. Ghai describes a "policy orchestration framework layer within block 404 work[ing] to pass the physical security related data (for example, personnel credentials and access privileges) and events (for example, alerts) from physical security systems, network and IT applications to the system for processing." Ghai, 12:10-14. Overcash describes that

a "collaborative detection module 206 uses the input from the dynamic profiling module 204 to detect attacks against a Web application," particularly monitoring and identifying abnormal activities against the Web application. Overcash, 9:24-39; Fig. 1. Because Ghai is configured to provide "systems and methods for integration and normalization of physical security data, states and events from disparate physical security systems to maintain in real-time rules based policy state information to enforce physical security policies uniformly across network and information technology (IT) systems," Ghai, 3:44-49, a POSITA would have been motivated to use Overcash's system or method of monitoring and identifying attacks against the Web application along with Ghai's system or method of collecting data and events to facilitate enforcing physical security policies uniformly across network and IT systems in Ghai. A POSITA would have readily appreciated that the value of sending and receiving "a dynamic context" including abnormal activities against Web applications as taught by Overcash could readily and beneficially be implemented in Ghai's computer network security system.

- For the Asserted Claims of the '616 patent, to the extent the references charted in any of the charts in Exhibit G do not expressly disclose "generating, by the trust orchestration server, temporal events based at least in part on analyzing the third party network endpoint assessments", it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A For example, to the extent Ghai is deemed to lack sufficient disclosure of "generating, by the trust orchestration server, temporal events based at least in part on analyzing the third party network endpoint assessments" in Claim 1, a POSITA would have been motivated to combine the teachings of Ghai with the teachings of U.S. Publication No. 2005/0005169 ("Kelekar"). Using a third-party service

to assess network endpoints is well known in the art. As acknowledged by the ’616 Patent, it is a “recent trend in computing” that enterprise software which normally perform assessment of network endpoints is no longer owned by the customer, but instead the enterprise’s Information Technology (IT) infrastructure can be provided by a third party and enterprise software applications are sold as service offerings. ’616 Pat, 1:20-25. And many enterprises and organizations are moving their IT infrastructures from self-managed on-premise data centers to service-provider managed outsourced virtual data centers, wherein which third parties may provide shared computing resources and applications running on those resources. *Id*., 1:45-50. Indeed, applying third party services to assess applications was well known. For example, Kelekar describes that a server can receive from third-party vulnerability assessment (VA) tools (e.g., Nessus) VA test results for services for which indications of service start or stop have been received. Kelekar, ¶ 156. Cooper describes that a policy-based monitor system 100 receives vulnerability assessment from third-party network scanner that is not a SecurVantage™ Scanner. Cooper, 14:14-23. Datta Ray describes that OT, IT, and ST systems receive security and risk assessments of their own from sub-domain specific expertise of individuals, ad hoc decisions from personal experience, and guidelines and alerts issued by government agencies and third parties. Datta Ray, 1:49-53; *see also* Datta Ray Provisional, p. 1. Dowd describes that vulnerabilities database 103 receives vulnerability information from a variety of resources, from a network configuration tool, or from software and hardware companies, hacker web sites, security monitoring services and the like. Dowd, 3:54-65, 4:14-25, 4:66-5:2. Similarly, generating temporal events based on analyzing third-party network endpoint assessments was also well known. For example, Kelekar describes that a server can concatenate multiple VA test results received from third-party VA tools. Kelekar, ¶¶ 156, 179. Cooper describes generating special policy events based on new and resolved

vulnerabilities. Cooper, 6:15-19. Datta Ray describes generating temporal hierarchy for events to handle phenomena of different time scales and asynchronous events. Datta Ray, 7:63-8:7; *see also* Datta Ray Provisional, p. 5. Both Ghai and Kelekar disclose vulnerability assessment of computer networks. Ghai describes that the PSS module of the integrated physical security system 118 executed by the server 116 receives events from networking systems from Operations Support Systems (OSS). *Id*., 15:9-60. Ghai also describes that the rules engine creates actionable alarms and alerts based at least in part on physical access changes as well as correlating security events. *Id*., 8:38-42. Kelekar describes that a server can concatenate multiple VA test results received from the third-party VA tools. Kelekar describes using various third-party VA tools to detect vulnerabilities; that is, they can be used to do vulnerability tests only on particular services that are listening on various ports. An example is given using the open-source tool Nessus. Kelekar, ¶ 156. Kelekar further describes a system that can perform incremental vulnerability assessment. In one variation, instead of running a complete VA test on learning that the status of port/interfaces on the host/device has changed, the server runs tests to find services running on the open ports that now have listening services on them. And the server runs tests on only those services and comes up with an incremental VA results. Then, the server can come up with the overall vulnerability assessment report for the host/device by concatenating the vulnerability assessment results for the services running on various ports. *Id*., ¶ 179. Because Ghai is configured to provide "systems and methods for integration and normalization of physical security data, states and events from disparate physical security systems to maintain in real-time rules based policy state information to enforce physical security policies uniformly across network and information technology (IT) systems," Ghai, 3:44-49, a POSITA would have been motivated to use Kelekar's generation of concatenated VA test results received from the third-party VA tools in the system of Ghai

to facilitate enforcing physical security policies uniformly across network and IT systems in Ghai. A POSITA would have readily appreciated that Kelekar's teachings could readily and beneficially be implemented in Ghai's computer network security system.

- For the Asserted Claims of the '616 patent, to the extent the references charted in any of the charts in Exhibit G do not expressly disclose "receiving, by the trust orchestration server, integrity measurement and verification reports", it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A For example, to the extent Ghai is deemed to lack sufficient disclosure of "receiving, by the trust orchestration server, integrity measurement and verification reports" in Claim 3, a POSITA would have been motivated to combine the teachings of Ghai with the teachings of U.S. Patent No. 7,451,488 ("Cooper"). The '616 Patent acknowledges that it was well known in the art that "[a]pplication runtime operational integrity is a critical factor in building end-to-end trust from device to service." '616 Pat, 2:18-20. And the patent describes that the industry had antivirus and network-based integrity measurement and verification services. *Id*., 2:18-31. Indeed, receiving integrity measurement and verification reports by a network or from collaboration services was well known. For example, Cooper describes using commercial products for vulnerability scanning. Cooper, 1:54-61. Thielamay describes that the Threat Analysis Module (TAM) 19 is capable of generating trouble-tickets via the internal Trouble Ticketing Module (TTM) 15 or other third party Trouble Ticketing system, and the trouble tickets are received by the administrator of the framework. Thielamay, ¶¶ 45-46. Both Ghai and Cooper are related to ensuring computer network security. Ghai describes that the defined policies can connect to systems of record for data verification and rules enforcement in real-time. Ghai, 8:42-47. Cooper describes

using commercial products for vulnerability scanning. Cooper, 1:54-61. For example, the Nessus remote security scanner is used for performing the vulnerability scans. *Id*., 10:53-56. Cooper further provides a live data page including a special row(s) for reporting on vulnerability scanners. *Id*., 4:58-61. Such page allows access to all vulnerability events within the reporting timeframe, irrespective of volume." *Id*. Because Ghai is configured to provide "systems and methods for integration and normalization of physical security data, states and events from disparate physical security systems to maintain in real-time rules based policy state information to enforce physical security policies uniformly across network and information technology (IT) systems," Ghai, 3:44-49, a POSITA would have been motivated to use Cooper's vulnerability scans and the live data page reporting on vulnerability scanners in the system of Ghai to facilitate scanning and reporting security across network and IT systems in Ghai. A POSITA would have readily appreciated that Cooper's teachings could readily and beneficially be implemented in Ghai's computer network system.

- For the Asserted Claims of the '616 patent, to the extent the references charted in any of the charts in Exhibit G do not expressly disclose "performing, by a malware analyzer, detection of code obfuscation and evasion techniques in reverse engineered code of the applications", it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A For example, to the extent DiFalco is deemed to lack sufficient disclosure of "performing, by a malware analyzer, detection of code obfuscation and evasion techniques in reverse engineered code of the applications" in Claim 7, a POSITA would have been motivated to combine the teachings of Ghai with the teachings of U.S. Patent No. 9,628,501 ("Datta Ray") or its provisional application U.S.

Provisional Patent Application No. 61/547,610 ("Datta Ray Provisional"). The '616 patent describes the existing security appliances such as network firewalls, intrusion detection/prevention systems, and antivirus and network-based integrity measurement and verification services that scan and audit the soft-core that comprises business critical systems, services and high value data silos. '616 Pat, 2:18-31. The patent also describes the existing attestation systems that permit reliable statements of evidence about those systems to be conveyed to remote parties, including other computers, evaluate the identity and integrity of components of a system (i.e., a target system being evaluated), and produce evidence that the target system will not engage in defined classes of misbehaviors. *Id*., 3:49-55. Indeed, malware analyzers were well known in the art. For example, Datta Ray describes a unified risk model performing adaptive orchestrations of situational awareness and domain knowledge including malware intelligence, or a third party performing security and risk assessments of OT, IT, and ST systems. Datta Ray, 1:49-53, 3:9-16, 3:26-31; *see also* Datta Ray Provisional, pp. 1, 2. Kelekar describes a server detecting vulnerabilities in software executables and that it is well known to detect vulnerabilities in software executables via execution trace checking. Kelekar, ¶¶ 23, 156. Overcash describes a collaborative detection module 206 or a correlation and analysis module 208 analyzing abnormal behavior of web applications and user behaviors and detecting attacks across multiple enterprises and security events from the collaborative detection module 206. Overcash, 9:30-52, 9:48-52, 51:58-52:2. Detecting obfuscation and evasion techniques was also well known. For example, Datta Ray describes a security and business risk analysis engine (SBRAE) that receives information system state of monitored MCEs including significant events or exogenous security and vulnerabilities of applications. Datta Ray, 12:32-38, 20:56-59; *see also* Datta Ray Provisional, pp. 7, 14. Overcash describes a policy that specifies whether the data matching BreachMark is to be masked (i.e. obfuscated) in

the logs, events sent to the console, and in reports. Overcash, 11:22-34. Both Ghai and Datta Ray are related to ensuring computer network security. Ghai is directed to systems and methods for integration and normalization of physical security data, states and events to and from disparate physical security systems to maintain in real-time rules based policy state information to enforce physical security policies uniformly across network and information technology (IT) systems." Ghai, Abstract. Ghai describes examining and processing all physical security data and events. Ghai, 1:59-65, 10:16-24; FIG. 3. It would have been obvious to a POSITA that examining and processing physical security data and events include malware analysis or detection of code obfuscation and evasion techniques. Datta Ray describes using a unified risk model to perform adaptive orchestrations of situational awareness, domain knowledge including malware intelligence, or using third party to perform security and risk assessments of OT, IT, and ST systems. Datta Ray, 1:49-53, 3:9-16, 3:26-31; *see also* Datta Ray Provisional, pp. 1, 2. Because Ghai is configured to provide "systems and methods for integration and normalization of physical security data, states and events from disparate physical security systems to maintain in real-time rules based policy state information to enforce physical security policies uniformly across network and information technology (IT) systems," Ghai, 3:44-49, a POSITA would have been motivated to use Datta Ray's malware analyzer in the system of Ghai to facilitate scanning and reporting security across network and IT systems in Ghai. A POSITA would have readily appreciated that Datta Ray's teachings could readily and beneficially be implemented in Ghai's computer network system.

- For the Asserted Claims of the '616 patent, to the extent the references charted in any of the charts in Exhibit G do not expressly disclose "the code obfuscation and evasion techniques including at least one of: multiple packing, anti-debug, anti-trace, anti-memory, and anti-emulation", it would have been obvious to combine the teachings of any of these

charted references with the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A For example, to the extent Ghai and Datta Ray are deemed to lack sufficient disclosure of "the code obfuscation and evasion techniques including at least one of: multiple packing, anti-debug, anti-trace, anti-memory, and anti-emulation" in Claim 7, a POSITA would have been motivated to combine the teachings of Ghai and Datta Ray with the teachings of Overcash. Ghai, Datta Ray, and Overcash are related to facilitating computer network security. Overcash discloses that collaborative detection module 206 of the Web application protection module 128 or the correlation and analysis module 208 of the Web application protection module 128 analyzes abnormal behavior of web applications and user behaviors, detecting attacks across multiple enterprises and security events from the collaborative detection module 206. Overcash, 9:30-52, 9:48-52, 51:58-52:2. Overcash further describes a policy that specify whether the data matching BreachMark is to be masked (i.e. obfuscated) in the logs, events sent to the console, and in reports. *Id*., 11:22-34. A POSITA would have been motivated to use Overcash's collaborative detection module 206 of the Web application protection module 128 or the correlation and analysis module 208 of the Web application protection module 128 and the related policy in the system of Ghai to facilitate scanning and reporting security across network and IT systems in Ghai. A POSITA would have readily appreciated that Overcash's teachings could readily and beneficially be implemented in Ghai's computer network system.

- For the Asserted Claims of the '616 patent, to the extent the references charted in any of the charts in Exhibit G do not expressly disclose "generating, by the trust orchestration server, temporal events based at least in part on analyzing the third party network endpoint assessments", it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary

skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A For example, to the extent Datta Ray is deemed to lack sufficient disclosure of "generating, by the trust orchestration server, temporal events based at least in part on analyzing the third party network endpoint assessments" in Claim 1, a POSITA would have been motivated to combine the teachings of Datta Ray with the teachings of Kelekar. Both Datta Ray and Kelekar are related to ensuring computer network security. Datta Ray discloses that the OT, IT, and ST systems receive security and risk assessments such as guidelines and alerts issued by government agencies and third parties. Datta Ray, 1:49-53; *see also* Datta Ray Provisional, p. 1. Datta Ray describes generating temporal hierarchy for events to handle phenomena of different time scales and asynchronous events. *Id*., 7:63-8:7; *see also* Datta Ray Provisional, p. 5. Kelekar describes that a server can concatenate multiple VA test results received from the third-party VA tools. Kelekar describes using various third-party VA tools to detect vulnerabilities; that is, they can be used to do vulnerability tests only on particular services that are listening on various ports. An example is given using the open-source tool Nessus. Kelekar, ¶ 156. Kelekar further describes a system that can perform incremental vulnerability assessment. In one variation, instead of running a complete VA test on learning that the status of port/interfaces on the host/device has changed, the server runs tests to find services running on the open ports that now have listening services on them. And the server runs tests on only those services and comes up with an incremental VA results. Then, the server can come up with the overall vulnerability assessment report for the host/device by concatenating the vulnerability assessment results for the services running on various ports. *Id*., ¶ 179. Datta Ray is configured to "analyze and mitigate business risks and enhance cyber security throughout enterprise-wide OT, IT, and ST networks to maintain a high level of integrity and security, even after the conventional control measures in the network are

compromised." Datta Ray, 10:46-50; *see also* Datta Ray Provisional, p. 6. To achieve this, Datta Ray provides "risk and security monitoring and control capabilities that adapt to existing security and control postures of target systems, subsystems, and elements at various hierarchical levels of the network based upon the real-time conditions on the network elements, as well as the elements of the underlying enterprise business processes that could be affected." *Id*., 10:51-58; *see also* Datta Ray Provisional, p. 6. A POSITA would have been motivated to use Kelekar's generation of concatenated VA test results received from the third-party VA tools in the system of Datta Ray to assist with temporal hierarchical organization of events to facilitate risk and security monitoring and control capabilities throughout the networks in Datta Ray. A POSITA would have readily appreciated that Kelekar's teachings could readily and beneficially be implemented in Datta Ray's networks to analyze and mitigate business risks.

- For the Asserted Claims of the '616 patent, to the extent the references charted in any of the charts in Exhibit G do not expressly disclose "sending, by the endpoint trust agent on a monitored device, a dynamic context", it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A For example, to the extent U.S. Patent No. 8,850,565 ("Patrick") is deemed to lack sufficient disclosure of "sending, by the endpoint trust agent on a monitored device, a dynamic context" in Claim 1, it would have been obvious to a POSITA that the system disclosed in Patrick could have been implemented so that the endpoint trust agent on a monitored device sends a dynamic context. Patrick discloses collecting and accepting data specific to a given attack such as "Attack Event Targeting Information (AETI)" and "security capabilities and topology data for different and varied communities of network-attached devices and

applications" from a plurality of input sources. Patrick, 2:50-58. Patrick also describes that the Incident Response Action System (IRAS) 101 combines with primary components such as "input handlers 102 to accept Attack Event Targeting Information (AETI) from multiple input sources." *Id*., 4:1-6. It would have been obvious to a POSITA that such a plurality of input sources send data specific to a given attack such as "Attack Event Targeting Information (AETI)" and "security capabilities and topology data for different and varied communities of network-attached devices and applications" to input handlers 102.

- For the Asserted Claims of the '616 patent, to the extent the references charted in any of the charts in Exhibit G do not expressly disclose "generating, by the trust orchestration server, temporal events based at least in part on analyzing the third party network endpoint assessments", it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A For example, to the extent Patrick is deemed to lack sufficient disclosure of "generating, by the trust orchestration server, temporal events based at least in part on analyzing the third party network endpoint assessments" in Claim 1, a POSITA would have been motivated to combine the teachings of Patrick with the teachings of Kelekar. Both Patrick and Kelekar are related to ensuring computer network security. Patrick describes that input handlers 102 of the Incident Response Action System (IRAS) 101 accepts Attack Event Targeting Information (AETI) from multiple input sources. *Id*., 4:1-6. And the IRAS generates the Contingency Access Points (CAP) 504 candidate coordination list through a review process that considers each device and application identified by the SPI function. *Id*., 12:57-60. Kelekar describes that a server can concatenate multiple VA test results received from the third-party VA tools. Kelekar describes using various third-party VA tools to detect vulnerabilities; that is, they can be

used to do vulnerability tests only on particular services that are listening on various ports. An example is given using the open-source tool Nessus. Kelekar, ¶ 156. Kelekar further describes a system that can perform incremental vulnerability assessment. In one variation, instead of running a complete VA test on learning that the status of port/interfaces on the host/device has changed, the server runs tests to find services running on the open ports that now have listening services on them. And the server runs tests on only those services and comes up with an incremental VA results. Then, the server can come up with the overall vulnerability assessment report for the host/device by concatenating the vulnerability assessment results for the services running on various ports. *Id*., ¶ 179. Patrick is configured to provide an IRAS that can "facilitate and track incident response workflows and coordinate countermeasures in near-real-time to quarantine and/or mitigate and survive dynamic events affecting the security of a computer network." Patrick, 2:43-49. This involves collecting and accepting from a plurality of input sources AETI, which is the data specific to a given attack, and security capabilities and topology data for different and varied communities of network-attached devices and applications. *Id*., 2:50-58. This also involves "identify[ing] optimal countermeasures and/or contingency courses of action in a non-deterministic manner specific to AETI weighted against security capabilities and topology data known for varied communities of network-attached devices and applications." *Id*., 2:59-64. A POSITA would have been motivated to use Kelekar's generation of concatenated VA test results received from the third-party VA tools in the system of Patrick to assist with collecting and accepting data specific to a given attack and security capabilities and topology data related to network-attached devices and applications from a plurality of input sources. A POSITA would have readily appreciated that Kelekar's teachings could readily and beneficially be implemented in Patrick's networks to facilitate and track incident response workflows and coordinate countermeasures in near-real-time.

- For the Asserted Claims of the ’616 patent, to the extent the references charted in any of the charts in Exhibit G do not expressly disclose “correlating, by the trust orchestration server, the received endpoint events and the generated temporal events”, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A For example, to the extent Patrick is deemed to lack sufficient disclosure of “correlating, by the trust orchestration server, the received endpoint events and the generated temporal events” in Claim 1, a POSITA would have been motivated to combine the teachings of Patrick with the teachings of Kelekar and Ghai. Correlating the received endpoint events and generated temporal events from third party network endpoint assessments were well known. As discussed above, generating temporal events based on analyzing the third-party network endpoint assessments was well known in the art. Correlating the received endpoint events and generated temporal events was also well known. The ’616 patent acknowledges that the industry had adopted multi-factor authentication technologies for increased protection from malicious activities. ’616 Pat, 2:54-59. Numerous prior art had described this technique. Ghai describes a rules engine creating actionable alarms and alerts based at least in part on physical access changes and correlating security events, Ghai, 8:38-42, particularly, a physical security state engine 326 within the PSS module 302 correlating “physical security system data and events to access permissions for individuals, groups, corporate, geographical or other defined domains on the network, applications or other secured IT or physical assets.” *Id*., 10:16-54. Kelekar describes a server generating reports that correlate event/action information received from device agents (e.g., port number where the event/action is the start of a service) and concatenation of VA test results received from the third-party VA tools (e.g., including

assessments/recommendations). Kelekar, ¶¶ 156, 179, 314, FIGS. 12-15. Thielamay describes a Correlation Engine Module of a framework correlating security alerts and events from separate systems to provide a global view of IT security status that identifies threat patterns as they develop." Thielamay, ¶¶ 18-21. Overcash describes a correlation and analysis module 208 analyzing events reported by the collaborative detection module 206 to determine if an attack is taking place. Overcash, 9:48-52. As discussed above, a POSITA would have been motivated to combine the teachings of Patrick with the teachings of Kelekar for the disclosure of "generating, by the trust orchestration server, temporal events based at least in part on analyzing the third party network endpoint assessments." Ghai describes an integrated physical security management process 118 executed by the server 116 receiving data (e.g., personnel credentials and access privileges) and events (e.g., alerts) ("the dynamic context including the endpoint events"). Ghai, 11:49-67. Ghai also describes a rules engine that uses security events including events from networking systems from Operations Support Systems (OSS), *id*., 15:9-60, to generate "actionable alarms and alerts for management applications, supervisory staff or physical security, network or IT systems" ("the generated temporal events"). *Id*., 8:38-42. Ghai further describes correlating rules-based event ("generated temporal events") and data that results in actionable state information ("the received endpoint events") for IT, network and other security systems. Ghai discloses a physical security state engine 326 within the PSS module 302 that correlates "physical security system data and events to access permissions for individuals, groups, corporate, geographical or other defined domains on the network, applications or other secured IT or physical assets." *Id*., 10:16-54. Patrick, Kelekar, and Ghai are related to ensuring computer network security. Patrick describes that input handlers 102 of the Incident Response Action System (IRAS) 101 receives alerts or data specific to a given attack. Patrick, 2:50-58. Kelekar describes that the server component of the server can

receive the noted indication that a network service has started or stopped. Kelekar, ¶¶ 161 –162, 246. A POSITA would have been motivated to have Ghai's server 116 to receive data and events related to the alerts or data specific to a given attack as in Patrick and the noted indication that a network service has started or stopped as in Kelekar. As discussed above, Patrick also describes that the IRAS 101 accepts Attack Event Targeting Information (AETI) from multiple input sources and generates the Contingency Access Points (CAP) 504 candidate coordination list. Patrick, 4:1-6, 12:57-60. Kelekar describes that a server can concatenate multiple VA test results received from the third-party VA tools. A POSITA would have been motivated to have Ghai's rules engine to generate "actionable alarms and alerts" based on security events including Patrick's AETI from multiple input sources and Kelekar's concatenated multiple VA test results received from the third-party VA tools. A POSITA would have further been motivated to correlate the above generated "actionable alarms and alerts" and received data and events for IT, network and other security systems such as the ones of Patrick and Kelekar. Patrick is configured to provide an IRAS that can "facilitate and track incident response workflows and coordinate countermeasures in near-real-time to quarantine and/or mitigate and survive dynamic events affecting the security of a computer network." *Id*., 2:43-49. Kelekar is configured to ensure real-time network-based vulnerability assessment of an enterprise/consumer host/device. Kelekar, ¶ 2. A POSITA would have been motivated to use Ghai's correlation of the generated "actionable alarms and alerts" and received data and events in the system of Patrick and Kelekar to assist with Patrick's facilitating and tracking incident response workflows and coordinate countermeasures in near-real-time.

- For the Asserted Claims of the '616 patent, to the extent the references charted in any of the charts in Exhibit G do not expressly disclose "generating, by the trust orchestration server, temporal events based at least in part on analyzing the third party network endpoint

assessments", it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A For example, to the extent Cooper is deemed to lack sufficient disclosure of "generating, by the trust orchestration server, temporal events based at least in part on analyzing the third party network endpoint assessments" in Claim 1, a POSITA would have been motivated to combine the teachings of Cooper with the teachings of Kelekar. Both Cooper and Kelekar are related to monitoring vulnerabilities of computer networks. Cooper describes using a scanner network object to represent a third party network scanner that is not a SecurVantage™ Scanner for vulnerability scanning for vulnerability assessment. Cooper, 14:14-23. Cooper also describes generating special policy events based on new and resolved vulnerabilities. *Id*., 6:15-19. A POSITA would have found it obvious that the new and resolved vulnerabilities are from a third-party vulnerability scans and the special policy events related to the new and resolved vulnerabilities are generated based on the third-party scans. Kelekar describes that a server can concatenate multiple VA test results received from the third-party VA tools. Kelekar describes using various third-party VA tools to detect vulnerabilities; that is, they can be used to do vulnerability tests only on particular services that are listening on various ports. An example is given using the open-source tool Nessus. Kelekar, ¶ 156. Kelekar further describes a system that can perform incremental vulnerability assessment. In one variation, instead of running a complete VA test on learning that the status of port/interfaces on the host/device has changed, the server runs tests to find services running on the open ports that now have listening services on them. And the server runs tests on only those services and comes up with an incremental VA results. Then, the server can come up with the overall vulnerability assessment report for the host/device by concatenating the

vulnerability assessment results for the services running on various ports. *Id*., ¶ 179. Because Cooper is configured to provide "[a] system and method for a vulnerability assessment mechanism that serves to actively scan for vulnerabilities on a continuous basis and interpret the resulting traffic in context of policy," Cooper, 2: 46-49, a POSITA would have been motivated to use Kelekar's generation of concatenated VA test results received from the third-party VA tools in the system of Patrick to assist with actively scan for vulnerabilities on a continuous basis from third party network scanners. A POSITA would have readily appreciated that Kelekar's teachings could readily and beneficially be implemented in Cooper's networks to facilitate generating special policy events based on new and resolved vulnerabilities.

- For the Asserted Claims of the '616 patent, to the extent the references charted in any of the charts in Exhibit G do not expressly disclose "generating, by the trust orchestration server, temporal events based at least in part on analyzing the third party network endpoint assessments", it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A For example, to the extent U.S. Patent No. 10,795,855 ("DiFalco") is deemed to lack sufficient disclosure of "generating, by the trust orchestration server, temporal events based at least in part on analyzing the third party network endpoint assessments" in Claim 1, a POSITA would have been motivated to combine the teachings of DiFalco with the teachings of Kelekar. Both DiFalco and Kelekar are related to monitoring vulnerabilities of computer networks. DiFalco describes that a compliance server 106 may receive or retrieve new or updated compliance policies 110 from a service or a compliance standards organization that defines industry standards or from another server, system, or storage. DiFalco, 5:15-20, 6:40-43. DiFalco also describes

that the compliance server 106 may generate an event to notify one or more event listeners of the compliance server 106 that data associated with a new change has been stored in the change database 108. *Id*., 2:47-54. A POSITA would have found it obvious that the compliance server 106 may generate an event based in part on analyzing the change data including new or updated compliance policies from a service or a compliance standards organization or analyze the change data from another server, system, or storage. Kelekar describes that a server can concatenate multiple VA test results received from the third-party VA tools. Kelekar describes using various third-party VA tools to detect vulnerabilities; that is, they can be used to do vulnerability tests only on particular services that are listening on various ports. An example is given using the open-source tool Nessus. Kelekar, ¶ 156. Kelekar further describes a system that can perform incremental vulnerability assessment. In one variation, instead of running a complete VA test on learning that the status of port/interfaces on the host/device has changed, the server runs tests to find services running on the open ports that now have listening services on them. And the server runs tests on only those services and comes up with an incremental VA results. Then, the server can come up with the overall vulnerability assessment report for the host/device by concatenating the vulnerability assessment results for the services running on various ports. *Id*., ¶ 179. Because DiFalco is configured to "perform[] continuous compliance assessment of target host data in response to changes on a target host," DiFalco, 1:18-21, a POSITA would have been motivated to use Kelekar's generation of concatenated VA test results received from the third-party VA tools in the system of DiFalco to facilitate generating an event based on analyzing the change data including those from another server, system, or storage. A POSITA would have readily appreciated that Kelekar's teachings could readily and beneficially be implemented in DiFalco's networks to perform continuous compliance assessment of target host data.

- For the Asserted Claims of the ’616 patent, to the extent the references charted in any of the charts in Exhibit G do not expressly disclose “a dynamic context including endpoint events and actions of the monitored device and applications executing on the monitored device at runtime”, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A For example, to the extent DiFalco is deemed to lack sufficient disclosure of sending and receiving “a dynamic context including endpoint events and actions of the monitored device and applications executing on the monitored device at runtime” in Claim 1, a POSITA would have also been motivated to combine the teachings of DiFalco with the teachings of Overcash. Both DiFalco and Overcash are related to facilitating computer network security. DiFalco describes “methods, systems, and articles for receiving, by a compliance server, change data associated with a change captured on a target host,” which “may have provided the change data in response to detecting the change, and the change data may include one or more rules, settings, and/or parameters.” *Id*., 1:59-65. Overcash describes that a central security manager of a computer network 106 receives security events within enterprises’ computer networks, Overcash, 51:58-52:2, and a collaborative detection module 206 of the Web application protection module 128 in a computer network 106 uses the input from the dynamic profiling module 204 to detect attacks against a Web application, particularly monitoring and identifying abnormal activities against the Web application. *Id*., 9:24-39; Fig. 1. Because DiFalco is configured to “perform[] continuous compliance assessment of target host data in response to changes on a target host,” DiFalco, 1:18-21, a POSITA would have been motivated to use Overcash’s system or method of receiving security events and monitoring and identifying attacks against the Web application to facilitate

receiving change data including rules, settings, and/or parameters associated with a change captured on a target host. A POSITA would have readily appreciated that the value of sending and receiving such "a dynamic context" including security events and attacks against the Web application as taught by Overcash could readily and beneficially be implemented in DiFalco's networks to perform continuous compliance assessment of target host data.

- For the Asserted Claims of the '616 patent, to the extent the references charted in any of the charts in Exhibit G do not expressly disclose "receiving, by the trust orchestration server, integrity measurement and verification reports", it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A For example, to the extent DiFalco is deemed to lack sufficient disclosure of "receiving, by the trust orchestration server, integrity measurement and verification reports" in Claim 3, a POSITA would have been motivated to combine the teachings of DiFalco with the teachings of Cooper. Both DiFalco and Cooper are related to ensuring computer network security. DiFalco describes that the compliance server 106 may receive or retrieve new or updated compliance policies from a service or a compliance standards organization that defines industry standards or from another server, system, or storage. DiFalco, 5:15-20, 6:40-43. DiFalco also describes that a compliance server 106 receives change data associated with a change captured on a target host. *Id*., 1:59-65. Then, the compliance server 106 may generate an event to notify one or more event listeners of the compliance server 106 that data associated with a new change has been stored in the change database 108. *Id*., 2:47-54. A POSITA would have been motivated to incorporate the generated event for the change data and the received new or updated compliance policies in a "integrity

measurement and verification reports" like the ones in Cooper. Cooper describes using commercial products for vulnerability scanning. Cooper, 1:54-61. For example, the Nessus remote security scanner is used for performing the vulnerability scans. *Id*., 10:53-56. Cooper further provides a live data page that includes a special row(s) for reporting on vulnerability scanners. Such page allows access to all vulnerability events within the reporting timeframe, irrespective of volume. *Id*., 4:58-61. Because DiFalco is configured to "perform[] continuous compliance assessment of target host data in response to changes on a target host," DiFalco, 1:18-21, a POSITA would have been motivated to use Cooper's vulnerability scans and the live data page reporting on vulnerability scanners in the system of DiFalco to include the generated events related to the change data and the received new or updated compliance policies to facilitate continuous compliance assessment of target host data in DiFalco's system.

- For the Asserted Claims of the '616 patent, to the extent the references charted in any of the charts in Exhibit G do not expressly disclose "performing, by a malware analyzer, detection of code obfuscation and evasion techniques in reverse engineered code of the applications", it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A For example, to the extent DiFalco is deemed to lack sufficient disclosure of "performing, by a malware analyzer, detection of code obfuscation and evasion techniques in reverse engineered code of the applications" in Claim 7, a POSITA would have been motivated to combine the teachings of DiFalco with the teachings of Datta Ray. Both DiFalco and Datta Ray are related to ensuring computer network security. DiFalco describes that a compliance server 106 receives change data associated with a change captured on a target host. DiFalco, 1:59-65. The compliance

server 106 then looks up all compliance policies 110 that match collection policies or rules 104 specified in the received change data. *Id*., 2:54-63. The compliance server 106 may also receive or retrieve new or updated compliance policies from a service or a compliance standards organization that defines industry standards or from another server, system, or storage. *Id*., 5:15-20, 6:40-43. Then, the compliance server 106 may generate an event to notify one or more event listeners of the compliance server 106 that data associated with a new change has been stored in the change database 108. *Id*., 2:47-54. It would have been obvious to a POSITA that the change data include code obfuscation and evasion techniques. Datta Ray describes using a unified risk model to perform adaptive orchestrations of situational awareness, domain knowledge including malware intelligence, or using third party to perform security and risk assessments of OT, IT, and ST systems. Datta Ray, 1:49-53, 3:9-16, 3:26-31; *see also* Datta Ray Provisional, pp. 1, 2. Because DiFalco is configured to "perform[] continuous compliance assessment of target host data in response to changes on a target host," DiFalco, 1:18-21, a POSITA would have been motivated to use Datta Ray's malware analyzer in the system of DiFalco to analyze change data to facilitate continuous compliance assessment of target host data in DiFalco's system. To the extent DiFalco and Datta Ray are deemed to lack sufficient disclosure of "the code obfuscation and evasion techniques including at least one of: multiple packing, anti-debug, anti-trace, anti-memory, and anti-emulation" in Claim 7, a POSITA would have been motivated to combine the teachings of DiFalco and Datta Ray with the teachings of Overcash. DiFalco, Datta Ray, and Overcash are related to facilitating computer network security. Overcash discloses that collaborative detection module 206 of the Web application protection module 128 or the correlation and analysis module 208 of the Web application protection module 128 analyzes abnormal behavior of web applications and user behaviors, detecting attacks across multiple enterprises and security events from the

collaborative detection module 206. Overcash, 9:30-52, 9:48-52, 51:58-52:2. Overcash further describes a policy that specify whether the data matching BreachMark is to be masked (i.e. obfuscated) in the logs, events sent to the console, and in reports. *Id*., 11:22-34. A POSITA would have been motivated to use Overcash's collaborative detection module 206 of the Web application protection module 128 or the correlation and analysis module 208 of the Web application protection module 128 and the related policy in the system of DiFalco to facilitate continuous compliance assessment of target host data in DiFalco's system. A POSITA would have readily appreciated that Overcash's teachings could readily and beneficially be implemented in DiFalco's computer network system.

- For the Asserted Claims of the '616 patent, to the extent the references charted in any of the charts in Exhibit G do not expressly disclose "sending, by the endpoint trust agent on a monitored device, a dynamic context", it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A For example, to the extent U.S. Patent Publication No. 2006/0191007 ("Thielamay") is deemed to lack sufficient disclosure of "sending, by the endpoint trust agent on a monitored device, a dynamic context" in Claim 1, it would have been obvious to a POSITA that the system disclosed in Thielamay could have been implemented so that the endpoint trust agent on a monitored device sends a dynamic context. Thielamay discloses that "Central Management Center (CMC) 10 provides an administrator," "a system's security pertinent information is gathered via Syslog and Microsoft event viewer as well as other log files analysis methods," and "center 10 provides monitoring of CPU, Memory, Network Interfaces, Disk Statistics, System Processes, System Load and more." Thielamay, ¶¶ 43-44. It would have

been obvious to a POSITA that the CMC 10 collects and sends a system's security pertinent information including events, which is the "dynamic context," to the framework.

- For the Asserted Claims of the '616 patent, to the extent the references charted in any of the charts in Exhibit G do not expressly disclose "sending, by the endpoint trust agent on a monitored device", it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A For example, to the extent Thielamay is deemed to lack sufficient disclosure of "sending, by the endpoint trust agent on a monitored device" in Claim 1, a POSITA would have also been motivated to combine the teachings of Thielamay with the teachings of Overcash. Both Thielamay and Overcash disclose security monitoring and management of computer network. Thielamay describes that "Central Management Center (CMC) 10 provides an administrator," "a system's security pertinent information is gathered via Syslog and Microsoft event viewer as well as other log files analysis methods," and "center 10 provides monitoring of CPU, Memory, Network Interfaces, Disk Statistics, System Processes, System Load and more." Thielamay, ¶¶ 43-44. As discussed above, it would have been obvious to a POSITA that the CMC 10 collects and sends a system's security pertinent information including events, which is the "dynamic context," to the framework. Overcash discloses that a group of enterprises, such as financial enterprises, that are monitored, "send security events within their computer networks to a central security manager," which may then "aggregate the events to identify if there is an attack against the industry segment and notify the enterprises so that they may take appropriate action, or the centralized security manager may coordinate the security actions across the enterprises." Overcash, 51:58-52:2. Thielamay's system is designed to provide "a framework for automating the manual process of security

monitoring and management." Thielamay, ¶ 4. Particularly, Thielamay aims to "provid[e] a framework that incorporates disparate IT security mechanisms into one cohesive system," that is "designed to International Organization for Standardization (ISO) standards and Request for Comments (RFC) protocols," "that correlates security alerts and events from separate systems to provide a global view of IT security status that identifies threat patterns as they develop," and "that maintains the security posture and integrity of all IT systems." *Id.*, ¶¶ 18-21. A POSITA would have been motivated to use Overcash's teachings of having a group of monitored enterprises send dynamic context such as security events in the system of Thielamay such that the CMC 10 sends a system's security events to the framework. A POSITA would have readily appreciated that Overcash's teachings of having a group of monitored devices send dynamic context could readily and beneficially be implemented to better achieve Thielamay's "framework for automating the manual process of security monitoring and management."

- For the Asserted Claims of the '616 patent, to the extent the references charted in any of the charts in Exhibit G do not expressly disclose "a dynamic context including endpoint events and actions of the monitored device and applications executing on the monitored device at runtime", it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A. For example, to the extent Thielamay is deemed to lack sufficient disclosure of "a dynamic context including endpoint events and actions of the monitored device and applications executing on the monitored device at runtime" in Claim 1, a POSITA would have also been motivated to combine the teachings of Thielamay with the teachings of Overcash. Thielamay describes that the CMC 10 monitors CPU, Memory, Network Interfaces, Disk Statistics, System Processes, System

Load and more, and the security posture module of the framework gathers hardware and software information related to services, versions, and revisions of software running in an environment of the monitored systems. *Id*., ¶¶ 43-44. Overcash describes that a central security manager of a computer network 106 receives security events within enterprises' computer networks, Overcash, 51:58-52:2, and a collaborative detection module 206 of the Web application protection module 128 in the computer network 106 uses the input from the dynamic profiling module 204 to detect attacks against a Web application, particularly monitoring and identifying abnormal activities against the Web application. *Id*., 9:24-39; Fig. 1. Because Thielamay's system is designed to provide "a framework for automating the manual process of security monitoring and management" as discussed above, a POSITA would have been motivated to use Overcash's system or method of receiving security events and monitoring and identifying attacks against the Web application to facilitate security monitoring and management in Thielamay's system.

- For the Asserted Claims of the '616 patent, to the extent the references charted in any of the charts in Exhibit G do not expressly disclose "receiving, by the trust orchestration server, integrity measurement and verification reports", it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A For example, to the extent Thielamay is deemed to lack sufficient disclosure of "receiving, by the trust orchestration server, integrity measurement and verification reports" in Claim 3, a POSITA would have been motivated to combine the teachings of Thielamay with the teachings of Cooper. Both Thielamay and Cooper are related to ensuring computer network security. Thielamay describes that the Threat Analysis Module (TAM) 19 is capable of generating trouble-tickets via the internal Trouble Ticketing Module (TTM) 15 or other

third party Trouble Ticketing system, and the trouble tickets are received by the administrator of the framework. Thielamay, ¶¶ 45-46. A POSITA would have been motivated to incorporate the trouble tickets in a "integrity measurement and verification reports" like the ones in Cooper. Cooper describes using commercial products for vulnerability scanning. Cooper, 1:54-61. For example, the Nessus remote security scanner is used for performing the vulnerability scans. *Id*., 10:53-56. Cooper further provides a live data page that includes a special row(s) for reporting on vulnerability scanners, and such a page allows access to all vulnerability events within the reporting timeframe, irrespective of volume. *Id*., 4:58-61. Because Thielamay's system is designed to provide "a framework for automating the manual process of security monitoring and management." Thielamay, ¶ 4. Particularly, Thielamay aims to "provid[e] a framework that incorporates disparate IT security mechanisms into one cohesive system," that is "designed to International Organization for Standardization (ISO) standards and Request for Comments (RFC) protocols," "that correlates security alerts and events from separate systems to provide a global view of IT security status that identifies threat patterns as they develop," and "that maintains the security posture and integrity of all IT systems." Thielamay, ¶¶ 18-21. A POSITA would have been motivated to use Cooper's vulnerability scans and the live data page reporting on vulnerability scanners in the system of Thielamay to facilitate security monitoring and management in Thielamay's system.

- For the Asserted Claims of the '616 patent, to the extent the references charted in any of the charts in Exhibit G do not expressly disclose "sending, by the endpoint trust agent on a monitored device", it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A For example, to the extent U.S. Patent

No. 7,315,801 ("Dowd") is deemed to lack sufficient disclosure of "sending, by the endpoint trust agent on a monitored device" in Claim 1, a POSITA would have also been motivated to combine the teachings of Dowd with the teachings of Overcash. Both Dowd and Overcash disclose computer network security monitoring and management. Dowd describes that a security modeling system 100 includes a simulator 101 and a network configuration module 105 in which the simulator 101 simulates networks based on network configuration data of the network configuration module 105. Dowd, 3:35-46; FIG. 1. Dowd also describes that the security modeling system 100 includes both a defender and an attacker interface receiving feedback from the security modeling system 100 based on current network status, time and resources expended and the interactions between the attacker and defender. *Id*., 5:9-25. It would have been obvious to a POSITA that the simulator 101 of the security modeling system 100 receives current network status, time and resources expended and the interactions between the attacker and defender, which is the "dynamic context," from the defender and attacker that are monitored. Overcash discloses that a group of enterprises, such as financial enterprises, that are monitored, "send security events within their computer networks to a central security manager. The central security manager may then aggregate the events to identify if there is an attack against the industry segment and notify the enterprises so that they may take appropriate action, or the centralized security manager may coordinate the security actions across the enterprises." Overcash, 51:58-52:2. Dowd is configured to provide "a modeling system which efficiently analyzes the vulnerability of a network using a network model." Dowd, 1:58-60. This is achieved by providing "a network security modeling system which simulates a network and analyzes security vulnerabilities of the network." *Id*., 2:1-3. A POSITA would have been motivated to use Overcash's teachings of having a group of monitored devices send dynamic context such as security events in the system of Dowd such that the defender and

attacker send security events to the simulator 101 of the security modeling system 100 for simulating the network and analyzes security vulnerabilities of the network. A POSITA would have readily appreciated that Overcash's teachings could readily and beneficially be implemented to better achieve Dowd's modeling system.

- For the Asserted Claims of the '616 patent, to the extent the references charted in any of the charts in Exhibit G do not expressly disclose "a dynamic context including … applications executing on the monitored device at runtime", it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A For example, To the extent Dowd is deemed to lack sufficient disclosure of "a dynamic context including … applications executing on the monitored device at runtime" in Claim 1, a POSITA would have also been motivated to combine the teachings of Dowd with the teachings of Overcash. Dowd describes that the security modeling system 100 receives current network status, time and resources expended and the interactions between the attacker and defender, *id*., 5:9-25, and network configuration data representing network components such as servers, workstations, routers and firewalls and the protocols and services that run on these components. *Id*., 3:35-46; FIG. 1. Overcash describes that a "collaborative detection module 206 uses the input from the dynamic profiling module 204 to detect attacks against a Web application," particularly monitoring and identifying abnormal activities against the Web application. Overcash, 9:24-39; Fig. 1. Because Dowd is configured to provide a "modeling system which efficiently analyzes the vulnerability of a network using a network model" as discussed above, a POSITA would have been motivated to use Overcash's system or method of monitoring and identifying attacks against the Web application in Dowd's system or method of receiving current network

status, time and resources expended, the interactions between the attacker and defender, and network configuration data to facilitate Dowd's objective of achieving its "modeling system."

- For the Asserted Claims of the '616 patent, to the extent the references charted in any of the charts in Exhibit G do not expressly disclose "generating … temporal events", it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A For example, To the extent Dowd is deemed to lack sufficient disclosure of "generating … temporal events" in Claim 1, a POSITA would have been motivated to combine the teachings of Dowd with the teachings of Datta Ray. Both Dowd and Datta Ray disclose computer network security. Dowd describes that a simulator 101 of the security modeling system 100 determines vulnerabilities through a simulation based on analyzing the vulnerability information from a variety of sources, data from a network configuration tool, or current vulnerability data provided by software and hardware companies, hacker web sites, security monitoring services and the like. Dowd, 4:14-25, 4:66-5:2. Datta Ray describes generating temporal hierarchy for events to handle phenomena of different time scales and asynchronous events. Datta Ray, 7:63-8:7; *see also* Datta Ray Provisional, p. 5. Dowd is configured to provide "a modeling system which efficiently analyzes the vulnerability of a network using a network model." Dowd, 1:58-60. This is achieved by providing "a network security modeling system which simulates a network and analyzes security vulnerabilities of the network." *Id*., 2:1-3. A POSITA would have been motivated to use Datta Ray's temporally hierarchically organized events in the system of Dowd to efficiently analyzes the vulnerability of a network. A POSITA would

have readily appreciated that Datta Ray's teachings could readily and beneficially be implemented to better achieve Dowd's objective.

- For the Asserted Claims of the '616 patent, to the extent the references charted in any of the charts in Exhibit G do not expressly disclose "correlating, by the trust orchestration server, the received endpoint events and the generated temporal events", it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A For example, to the extent Dowd is deemed to lack sufficient disclosure of "correlating, by the trust orchestration server, the received endpoint events and the generated temporal events" in Claim 1, a POSITA would have been motivated to combine the teachings of Dowd with the teachings of Ghai. Dowd and Ghai are related to ensuring computer network security. Dowd describes that the simulator 101 of a security modeling system receives current network status, time and resources expended and the interactions between the attacker and defender. *Id*., 3:35-46; 5:9-25; FIG. 1. Dowd also describes that the simulator 101 analyzes the vulnerability information from a variety of sources, data from a network configuration tool, or current vulnerability data provided by software and hardware companies, hacker web sites, security monitoring services and the like. *Id*., 3:54-65. Dowd further describes that the simulator 101 determines vulnerabilities of the simulated network based on the vulnerability information and network configuration data output by a third-party network configuration tool such as CyberCop, Cyberscanner or Axent Enterprise Management System or a combination of data output by a network configuration tool and data provided from the output of an objective network or from a user such as an administrator. *Id*., 4:14-25. The simulator 101 also looks at the interactions of the network components as well as the interior and the exposed face of the network and

determines security vulnerabilities. *Id*. Ghai describes an integrated physical security management process 118 executed by the server 116 receives data (for example, personnel credentials and access privileges) and events (for example, alerts), which are the "the dynamic context including the endpoint events." Ghai, 11:49-67. A POSITA would have been motivated to have Ghai's server 116 to receive data and events related to the current network status, time and resources expended and the interactions between the attacker and defender as Dowd discloses. Ghai also describes a rules engine that uses security events including events from networking systems from Operations Support Systems (OSS), *id*., 15:9-60, to generate "actionable alarms and alerts for management applications, supervisory staff or physical security, network or IT systems," which are the "the generated temporal events." *Id*., 8:38-42. A POSITA would have been motivated to have Ghai's rules engine to generate "actionable alarms and alerts" based on security events including Dowd's vulnerability information and network configuration data output by a third-party network configuration tool. Ghai further describes correlating rules-based event ("generated temporal events") and data that results in actionable state information ("the received endpoint events") for IT, network and other security systems. Ghai discloses a physical security state engine 326 within the PSS module 302 that correlates "physical security system data and events to access permissions for individuals, groups, corporate, geographical or other defined domains on the network, applications or other secured IT or physical assets." *Id*., 10:16-54. A POSITA would further be motivated to correlate the above generated "actionable alarms and alerts" and received data and events for IT, network and other security systems such as the one of Dowd that looks at the interactions of the network components along with the interior and the exposed face of the network to determines security vulnerabilities. Dowd is configured to provide "a modeling system which efficiently analyzes the vulnerability of a network using a network model." Dowd,

1:58-60. This is achieved by providing "a network security modeling system which simulates a network and analyzes security vulnerabilities of the network." *Id*., 2:1-3. A POSITA would have been motivated to use Ghai's correlation of the generated "actionable alarms and alerts" and received data and events in the system of Dowd to assist with efficiently analyzing the vulnerability of a network. A POSITA would have readily appreciated that Ghai's teachings could readily and beneficially be implemented to better achieve the above Dowd's objective.

- For the Asserted Claims of the '616 patent, to the extent the references charted in any of the charts in Exhibit G do not expressly disclose "a dynamic context including endpoint events and actions of the monitored device and applications executing on the monitored device at runtime," or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Schrecker which expressly discloses a "dynamic context including endpoint events and actions of the monitored device and applications executing on the monitored device at runtime" (e.g. asset configuration data), and/or Jaeger which describes "analytic integrity" software of a monitored computer system sending a set of trusted subjects for application 22A, runtime information for application 22A, and loaded program code for application 22A, to remote computer system 24. Jaeger, Fig. 1, 11:37-57. Also including also including Cooper, Datta Ray, DiFalco, Dowd, Ghai, Kelekar, Overcash, Patrick, Thielamay, Wootton, Das, Overcash II, Sreenivas, Leone, Singla, Bose, OfficeScan. A person of skill in the art would be motivated to combine these references because they each disclose methods of providing attestation services in the field of computer security. It is the nature of those of skill in the art of computer security to monitor developments in

competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems attest to operational integrity of a computing environment and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how other solutions in this field attest to operational integrity, including what information they use to do so. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’616 patent, to the extent the references charted in any of the charts in Exhibit G do not expressly disclose “receiving, by the trust orchestration server, third party network endpoint assessments”, or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Schrecker which expressly discloses “receiving, by the trust orchestration server, third party network endpoint assessments” (e.g. McaFee ePolicy Orchestrator, Active Directory, etc.), also including Cooper, Datta Ray, DiFalco, Dowd, Ghai, Jaeger, Kelekar, Overcash, Patrick, Thielamay, Wootton, Das, Overcash II, Sreenivas, Leone, Singla, Bose, OfficeScan. A person of skill in the art would be motivated to combine these references because they each disclose methods of providing attestation services in the field of computer security. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine

relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems attest to operational integrity of a computing environment and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how other solutions in this field attest to operational integrity, including what information they use to do so. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results

- For the Asserted Claims of the ’616 patent, to the extent the references charted in any of the charts in Exhibit G do not expressly disclose “a computing platform comprising a network trust agent, an endpoint trust agent, and a trust orchestration server”, or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any of the other charted references, the AAPA, and/or of those references identified in Section III.A, including, for example, Singla, which discloses a method of providing an attestation service for providing runtime operational integrity of a system (e.g., generating real-time asset information based on network activity) using a computing platform (e.g., a network asset information management system), an endpoint trust agent (e.g., data sources), and a trust orchestration server (e.g., network asset information management system), also including also including Cooper, Datta Ray, DiFalco, Dowd, Ghai, Jaeger, Kelekar, Overcash, Patrick, Thielamay, Wootton, Das, Overcash II, Sreenivas, Leone, Schrecker, Bose, OfficeScan. A person of skill in the art

would be motivated to combine these references because they each disclose methods of providing attestation services in the field of computer security. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems attest to operational integrity of a computing environment and mimic those same techniques. For example, it would have been obvious to a person of skill to look at how other solutions in this field attest to operational integrity, including what information and components they use to do so. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

Consistent with the exemplary motivations to combine each of the references discussed above, the below table provides a summary of the exemplary combinations of references rendering obvious to a POSA, with or without a POSA's knowledge (including the AAPA and general state of the art), skill and/or creativity, the Asserted Claims of the '616 Patent.

Reference	**Reference Alone or in Combination With One or More of the Following**
Cooper	Datta Ray, DiFalco, Dowd, Ghai, Jaeger, Kelekar, Overcash, Patrick, Thielamay, Wootton, Das, Overcash II, Sreenivas, Leone, Schrecker, Singla, Bose, OfficeScan
Datta Ray	Cooper, DiFalco, Dowd, Ghai, Jaeger, Kelekar, Overcash, Patrick, Thielamay, Wootton, Das, Overcash II, Sreenivas, Leone, Schrecker, Singla, Bose, OfficeScan
DiFalco	Cooper, Datta Ray, Dowd, Ghai, Jaeger, Kelekar, Overcash, Patrick, Thielamay, Wootton, Das, Overcash II, Sreenivas, Leone, Schrecker, Singla, Bose, OfficeScan

Reference	Reference Alone or in Combination With One or More of the Following
Dowd	Cooper, Datta Ray, DiFalco, Ghai, Jaeger, Kelekar, Overcash, Patrick, Thielamay, Wootton, Das, Overcash II, Sreenivas, Leone, Schrecker, Singla, Bose, OfficeScan
Ghai	Cooper, Datta Ray, DiFalco, Dowd, Jaeger, Kelekar, Overcash, Patrick, Thielamay, Wootton, Das, Overcash II, Sreenivas, Leone, Schrecker, Singla, Bose, OfficeScan
Jaeger	Cooper, Datta Ray, DiFalco, Dowd, Ghai, Kelekar, Overcash, Patrick, Thielamay, Wootton, Das, Overcash II, Sreenivas, Leone, Schrecker, Singla, Bose, OfficeScan
Kelekar	Cooper, Datta Ray, DiFalco, Dowd, Ghai, Jaeger, Overcash, Patrick, Thielamay, Wootton, Das, Overcash II, Sreenivas, Leone, Schrecker, Singla, Bose, OfficeScan
Overcash	Cooper, Datta Ray, DiFalco, Dowd, Ghai, Jaeger, Kelekar, Patrick, Thielamay, Wootton, Das, Overcash II, Sreenivas, Leone, Schrecker, Singla, Bose, OfficeScan
Patrick	Cooper, Datta Ray, DiFalco, Dowd, Ghai, Jaeger, Kelekar, Overcash, Thielamay, Wootton, Das, Overcash II, Sreenivas, Leone, Schrecker, Singla, Bose, OfficeScan
Thielamay	Cooper, Datta Ray, DiFalco, Dowd, Ghai, Jaeger, Kelekar, Overcash, Patrick, Wootton, Das, Overcash II, Sreenivas, Leone, Schrecker, Singla, Bose, OfficeScan
Wootton	Cooper, Datta Ray, DiFalco, Dowd, Ghai, Jaeger, Kelekar, Overcash, Patrick, Thielamay, Das, Overcash II, Sreenivas, Leone, Schrecker, Singla, Bose, OfficeScan
Das	Cooper, Datta Ray, DiFalco, Dowd, Ghai, Jaeger, Kelekar, Overcash, Patrick, Thielamay, Wootton, Overcash II, Sreenivas, Leone, Schrecker, Singla, Bose, OfficeScan
Overcash II	Cooper, Datta Ray, DiFalco, Dowd, Ghai, Jaeger, Kelekar, Overcash, Patrick, Thielamay, Wootton, Das, Sreenivas, Leone, Schrecker, Singla, Bose, OfficeScan
Sreenivas	Cooper, Datta Ray, DiFalco, Dowd, Ghai, Jaeger, Kelekar, Overcash, Patrick, Thielamay, Wootton, Das, Overcash II, Leone, Schrecker, Singla, Bose, OfficeScan
Leone	Cooper, Datta Ray, DiFalco, Dowd, Ghai, Jaeger, Kelekar, Overcash, Patrick, Thielamay, Wootton, Das, Overcash II, Sreenivas, Schrecker, Singla, Bose, OfficeScan

Reference	Reference Alone or in Combination With One or More of the Following
Schrecker	Cooper, Datta Ray, DiFalco, Dowd, Ghai, Jaeger, Kelekar, Overcash, Patrick, Thielamay, Wootton, Das, Overcash II, Sreenivas, Leone, Singla, Bose, OfficeScan
Singla	Cooper, Datta Ray, DiFalco, Dowd, Ghai, Jaeger, Kelekar, Overcash, Patrick, Thielamay, Wootton, Das, Overcash II, Sreenivas, Leone, Schrecker, Bose, OfficeScan
Bose	Cooper, Datta Ray, DiFalco, Dowd, Ghai, Jaeger, Kelekar, Overcash, Patrick, Thielamay, Wootton, Das, Overcash II, Sreenivas, Leone, Schrecker, Singla, OfficeScan
OfficeScan	Cooper, Datta Ray, DiFalco, Dowd, Ghai, Jaeger, Kelekar, Overcash, Patrick, Thielamay, Wootton, Das, Overcash II, Sreenivas, Leone, Schrecker, Singla, Bose

Additionally, Defendant incorporates by reference each of the obviousness combinations and motivations to combine presented in the petition for the following *inter partes* review proceeding(s) (IPRs) and any additional petitions for IPRs that may be filed challenging the patentability of the ’616 patent:

- IPR2023-00706
- IPR2023-00799

8. ’356 Patent

To the extent that Plaintiff contends that any of the references identified as anticipatory references fail to disclose one or more elements, those anticipatory references render the Asserted Claims obvious, either alone or in combination with other references. Defendant contends that one of skill in the art, at the time the alleged inventions were made, would have been motivated to combine the references disclosed herein in such a way as to reach the alleged inventions, as described in further detail below. Defendant presently believes a reasonable basis exists that each of the claims asserted against it would have been obvious to one of ordinary skill in the art at the time of the alleged invention.

Below, Defendant identifies their current bases for obviousness. As seen below, Defendant presently asserts certain combinations of prior art reference(s) with or without the knowledge of one of ordinary skill in the art. Defendant offers these combinations without conceding that the reference(s) standing alone are insufficient to invalidate the relevant claims. Defendant specifically reserves their rights to combine any of its anticipatory references or any combinations of references with the knowledge of ordinary skill in the art to rebut a contention of Plaintiff that such reference(s) standing alone do not disclose every element of the Asserted Claims.

Defendant lists below prior art references and combinations of prior art references now known to Defendant, which Defendant contends render at least obvious the '356 Asserted Claims. The corresponding charts in Exhibit P for each prior art reference identifies where and how each alleged item of prior art meets one or more limitations of the Asserted Claims. Pursuant to Patent Local Rule 3-6, Defendant expressly reserves the right to supplement these charts to provide additional details about obviousness should the Court's claim construction require, or should Plaintiff serve adequate Infringement Contentions that better serve to notify Defendant which elements of their products allegedly correspond with the claim elements. Defendant's contention that the references in this section in various combinations render obvious the '356 Asserted Claims under 35 U.S.C. § 103 is in no way an admission or suggestion that each reference does not independently anticipate or render obvious the '356 Asserted Claims under 35 U.S.C. § 102. These combinations would have been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field.

It would been obvious to combine each of the references discussed below because they are analogous art generally directed to computer network security. The technology described in the '356 Patent was widely known before the alleged priority date of the '356 Patent and provided numerous motivations to combine the prior art references and systems as exemplified below. The

’356 Patent combines these components according to their known, predictable and intended uses, just as they were and/or would have been combined by a POSA in the prior art. None of the combinations presents an unexpected result, a counterintuitive combination, or a technological challenge; and no such description exists in the common specification of the asserted patents.

Accordingly, the prior art cited herein would have been readily combined according to methods known to those of ordinary skill within the field of computer network security to yield predictable results. The substitution of one element in this system could have been predictably achieved by one of ordinary skill at the time of the alleged invention. One of ordinary skill in the art would have been able to select appropriate attributes of one for inclusion in another. Those of ordinary skill in the art could have employed known techniques to improve similar prior art devices in the same way as claimed in the ’356 Patent. Thus, the combinations all have well defined, predictable, expectation of success.

- For the Asserted Claims of the ’356 patent, to the extent the references charted in any of the charts in Exhibit P do not expressly disclose “program instructions to determine if the packet is a known exploit,” or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the AAPA and with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any other charted references and/or of those references identified in Section III.A, including, for example: Apap, Copeland, Desai, Fretwell, Gassen, Gluck, Gupta, Jerrim, Mualem, Stute, Triulzi. For example, Desai at [0052] (“After the normalization process, a search for a match may be conducted against a Known Offender or attack signature database.”). A person of skill in the art would be motivated to combine these references because they each disclose methods, apparatuses, and techniques for detecting intrusions and/or anomalous behavior in order to improve computer security. It is the nature of those of skill in the art of computer security to monitor developments in competing products and

combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems determine if a packet is a known exploit, particularly because security programs commonly consider and attempt to identify known exploits to protect against known attacks that risk harming the computer and/or the user. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’356 patent, to the extent the references charted in any of the charts in Exhibit P do not expressly disclose “program instructions to determine if the packet is addressed to a broadcast IP address of a network,” or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the AAPA and with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any other charted references and/or of those references identified in Section III.A, including, for example: Cooper, Copeland, Desai, Fretwell, Gassen, Gupta, Jerrim, Mualem, Roux, Stute, Triulzi. For example, Cooper at [0314]-[0315] (“Another intradomain IP rule is used to segregate typical broadcast and multicast traffic within an enterprise network.”). A person of skill in the art would be motivated to combine these references because they each disclose methods, apparatuses, and techniques for detecting intrusions and/or anomalous behavior in order to improve computer security. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems detect the type of

IP address to which a packet is directed because security programs commonly consider the intended target of a packet in assessing the risk of malicious activity, because, for example, the target address may be known to correspond to either malicious or not malicious traffic. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’356 patent, to the extent the references charted in any of the charts in Exhibit P do not expressly disclose “program instructions to determine if the packet is network administration traffic,” or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the AAPA and with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any other charted references and/or of those references identified in Section III.A, including, for example: Apap, Carpenter, Cooper, Copeland, Desai, Fretwell, Gassen, Gupta, Jerrim, Mualem, Roux, Stute, Triulzi. For example, Carpenter at 6:46-7:6 (“Step 102 preferably includes determining whether some portion of the administrative packets in the aggregate flow meets or exceeds the corresponding maximum bandwidth limit.”). A person of skill in the art would be motivated to combine these references because they each disclose methods, apparatuses, and techniques for detecting intrusions and/or anomalous behavior in order to improve computer security. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems detect network administration traffic because

security programs commonly consider the type of traffic in assessing the risk of malicious activity, because, for example, certain types of traffic may be known to be either malicious or not malicious traffic. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’356 patent, to the extent the references charted in any of the charts in Exhibit P do not expressly disclose “program instructions, responsive to [any of multiple conditions] to determine that the packet is not a new, exploit candidate,” or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the AAPA and with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any other charted references and/or of those references identified in Section III.A, including, for example: Afek, Apap, Copeland, Desai, Eichstaedt, Fretwell, Gassen, Gluck, Gupta, Jerrim, Mualem, Roux, Stute, Triulzi. For example, Copeland at 8:48-54 (“The port profiling engine 155 analyzes the flow data to distinguish legitimate flows from probes.”). A person of skill in the art would be motivated to combine these references because they each disclose methods, apparatuses, and techniques for detecting intrusions and/or anomalous behavior in order to improve computer security. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems detect new exploit candidates because security programs commonly consider whether something is a known or new exploit in assessing security risk

and in maintaining an up-to-date database that can protect against known attacks in the future and on other devices. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’356 patent, to the extent the references charted in any of the charts in Exhibit P do not expressly disclose “program instructions, responsive to the packet [not meeting each of multiple conditions] to determine and report that the packet is a new, exploit candidate,” or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the AAPA and with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any other charted references and/or of those references identified in Section III.A, including, for example: Afek, Apap, Copeland, Desai, Eichstaedt, Fretwell, Gassen, Gluck, Gupta, Jerrim, Mualem, Roux, Stute, Triulzi. For example, Copeland at 9:54-10:25 (“Any Point outside of the profile will generate an alarm while operating in Mode 4.”). A person of skill in the art would be motivated to combine these references because they each disclose methods, apparatuses, and techniques for detecting intrusions and/or anomalous behavior in order to improve computer security. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems detect new exploit candidates because security programs commonly consider whether something is a known or new exploit in assessing security risk and in maintaining an up-to-date database that can protect against known attacks in the

future and on other devices. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’356 patent, to the extent the references charted in any of the charts in Exhibit P do not expressly disclose “program instructions are stored on the computer-readable tangible storage device,” or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the AAPA and with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any other charted references and/or of those references identified in Section III.A, including, for example: Afek, Apap, Carpenter, Cooper, Copeland, Desai, Eichstaedt, Fretwell, Gassen, Gluck, Gupta, Jerrim, Mualem, Roux, Stute, Triulzi. For example, Afek at 11:64-12:7 (“These functions can be implemented in software, firmware or hardware within a stand-alone networked digital data processing device, or as part of (or otherwise in connection with) a switch, router, personal computer, workstation, server, or other digital data processing apparatus.”). A person of skill in the art would be motivated to combine these references because they each disclose methods, apparatuses, and techniques for detecting intrusions and/or anomalous behavior in order to improve computer security. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would understand security programs are commonly implemented using code stored on tangible storage devices, such as hard drives or other memory devices. These combinations would have also been obvious to try given that there were a finite number of

identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’356 patent, to the extent the references charted in any of the charts in Exhibit P do not expressly disclose “program instructions to determine if the packet is web crawler traffic,” or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the AAPA and with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any other charted references and/or of those references identified in Section III.A, including, for example: Afek, Copeland, Desai, Eichstaedt, Fretwell, Gassen, Jerrim. For example, Afek at 4:18-33 (“Victim and potential victim traffic can be classified, according to related aspects of the invention, by their source types, e.g., whether they arise from individual users, plural users sharing host or proxy, web crawlers, monitoring servers, and so forth.”). A person of skill in the art would be motivated to combine these references because they each disclose methods, apparatuses, and techniques for detecting intrusions and/or anomalous behavior in order to improve computer security. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems detect the type of traffic, such as web crawler traffic, because security programs commonly consider the type of traffic in assessing the security risk because, for example, certain types of traffic may be known to be either malicious or not malicious traffic. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art,

and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’356 patent, to the extent the references charted in any of the charts in Exhibit P do not expressly disclose “determine if the packet is a known exploit by searching the packet for a known signature of a known exploit,” or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the AAPA and with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any other charted references and/or of those references identified in Section III.A, including, for example: Apap, Copeland, Desai, Gluck, Gupta, Jerrim, Mualem, Stute, Triulzi. For example, Desai at [0052] (“After the normalization process, a search for a match may be conducted against a Known Offender or attack signature database. As the name implies, the attack signature database contains “known” signatures from prior and previously encountered attacks.”). A person of skill in the art would be motivated to combine these references because they each disclose methods, apparatuses, and techniques for detecting intrusions and/or anomalous behavior in order to improve computer security. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems determine if a packet is a known exploit using signatures, particularly because security programs commonly consider and attempt to identify known exploits by their signature to protect against known attacks, and variants that are close enough to match the signature, that risk harming the computer and/or the user. These combinations would have also been obvious to try given that there were a finite number of identified, predictable

solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’356 patent, to the extent the references charted in any of the charts in Exhibit P do not expressly disclose “program instructions determine if the packet is a known exploit by comparing an identity of the packet to one or more identities, sent by an intrusion detection system, of respective packet(s) which the intrusion detection system determined to contain a known exploit,” or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the AAPA and with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any other charted references and/or of those references identified in Section III.A, including, for example: Apap, Copeland, Desai, Gassen, Gluck, Gupta, Jerrim, Mualem, Stute, Triulzi. For example, Desai at [0052] (“After the normalization process, a search for a match may be conducted against a Known Offender or attack signature database. As the name implies, the attack signature database contains “known” signatures from prior and previously encountered attacks.”). A person of skill in the art would be motivated to combine these references because they each disclose methods, apparatuses, and techniques for detecting intrusions and/or anomalous behavior in order to improve computer security. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems determine if a packet is a known exploit by comparing to identities from an intrusion detection systems, particularly because security programs commonly consider and attempt to identify known exploits by their identities (including in conjunction with

known security systems such as intrusion detection systems) to protect against known attacks, and variants that are close enough to match the identities, that risk harming the computer and/or the user. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’356 patent, to the extent the references charted in any of the charts in Exhibit P do not expressly disclose “program instructions, responsive to the fifth program instructions determining that the packet is a new exploit candidate, to determine a signature of the packet [or a sequence of packets including the first packet], and report the new exploit candidate and the signature to an administrator,” or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the AAPA and with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any other charted references and/or of those references identified in Section III.A, including, for example: Apap, Cooper, Copeland, Desai, Fretwell, Gassen, Gluck, Gupta, Jerrim, Mualem, Stute, Triulzi. For example, Copeland at 9:54-10:25 (“Mode 4. Like Mode 3, except an alarm (by email, email to beeper, or SNMP Trap packet) is issued as soon as an Out of Profile operation is observed.”). A person of skill in the art would be motivated to combine these references because they each disclose methods, apparatuses, and techniques for detecting intrusions and/or anomalous behavior in order to improve computer security. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated

to look at how other products and systems identify the signature of a new exploit, particularly because security programs commonly keep a database or other record of known exploits that they update with newly-discovered exploits in order to provide security against new attacks. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’356 patent, to the extent the references charted in any of the charts in Exhibit P do not expressly disclose “program instructions determine if the packet is addressed to a broadcast IP address of the network by comparing a destination IP address of the packet to a gateway IP address of the network and a netmask of the network which identifies a broadcast IP address of the network,” or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the AAPA and with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any other charted references and/or of those references identified in Section III.A, including, for example: Cooper, Copeland, Desai, Fretwell, Gassen, Gupta, Jerrim, Mualem, Roux, Stute, Triulzi. For example, Jerrim at 14:5-11 (“The port profile engine 155 stores in a host structure 160 the zone locking data 340 for each zone. As shown, Zone B 314 has been assigned in a known manner the subnet IP address 192.168.1.0 (locnetaddr) 342 with a corresponding subnet mask 255.255.255.0 (locnetmask) 344.”). A person of skill in the art would be motivated to combine these references because they each disclose methods, apparatuses, and techniques for detecting intrusions and/or anomalous behavior in order to improve computer security. It is the

nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems identify the address to which traffic is directed (including using well-known features of IP communications such as IP addresses, gateway IP addresses, and netmasks) particularly because security programs consider the intended target in assessing the security risk, because, for example, certain targets may correspond to either malicious or not malicious traffic. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’356 patent, to the extent the references charted in any of the charts in Exhibit P do not expressly disclose “program instructions also determine if the packet has a protocol listed in a list of protocols previously determined to be harmless network broadcast traffic,” or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the AAPA and with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any other charted references and/or of those references identified in Section III.A, including, for example: Carpenter, Cooper, Copeland, Desai, Fretwell, Gassen, Gupta, Jerrim, Mualem, Roux, Stute, Triulzi. For example, Copeland at 8:66-9:45 (“Port profiling entails keeping two lists for each of the hosts: 1) a profile list by Protocol and Port Number (0 to 65,535) (for TCP and UDP), and simply by Protocol (number) for other Transport Layer protocols, and type of operation (as a Client or Server) for of all allowed operations

(the "Port Profile").”). A person of skill in the art would be motivated to combine these references because they each disclose methods, apparatuses, and techniques for detecting intrusions and/or anomalous behavior in order to improve computer security. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems identify the type of traffic, including by identifying the protocol used to deliver the packet, particularly because security programs consider the type of traffic and protocol used in assessing the security risk, because, for example, certain protocols may be known to be correspond to either malicious or not malicious traffic. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the ’356 patent, to the extent the references charted in any of the charts in Exhibit P do not expressly disclose “program instructions determine if the packet is network administration traffic by comparing an IP protocol and IP address of the packet to a list of combinations of IP protocols and IP addresses previously determined to be network administration traffic,” or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the AAPA and with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any other charted references and/or of those references identified in Section III.A, including, for example: Cooper, Copeland, Desai, Fretwell, Gassen, Gupta, Jerrim, Roux, Stute, Triulzi. For example, Copeland at 8:66-9:45 (“Port profiling entails keeping

two lists for each of the hosts: 1) a profile list by Protocol and Port Number (0 to 65,535) (for TCP and UDP), and simply by Protocol (number) for other Transport Layer protocols, and type of operation (as a Client or Server) for of all allowed operations (the "Port Profile")."). A person of skill in the art would be motivated to combine these references because they each disclose methods, apparatuses, and techniques for detecting intrusions and/or anomalous behavior in order to improve computer security. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems identify the type of traffic, including by identifying the protocol and address used to deliver the packet, particularly because security programs consider the protocol and address in determining the security risk, because, for example, certain protocols may be known to be correspond to either malicious or not malicious traffic. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the '356 patent, to the extent the references charted in any of the charts in Exhibit P do not expressly disclose "program instructions, responsive to the packet not being a known exploit … to identify a sequence of packets including the first said packet, the sequence of packets being a new, exploit candidate," or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the AAPA and with the knowledge, common sense, and/or creativity of a person of ordinary skill in the art, the teachings of any other charted references and/or of those

references identified in Section III.A, including, for example: Apap, Cooper, Copeland, Desai, Fretwell, Gassen, Gluck, Gupta, Jerrim, Mualem, Stute, Triulzi. For example, Gluck at 7:46-59 ("In step 240, the processor copies the new virus signature data from the disk to a memory buffer (not shown) of the system 10. The processor 20 in step 246 updates the virus signature files 110 with the new virus signature data from the disk 170."). A person of skill in the art would be motivated to combine these references because they each disclose methods, apparatuses, and techniques for detecting intrusions and/or anomalous behavior in order to improve computer security. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems identify a sequence of packets as a new exploit candidates, particularly because security programs commonly keep a database or other record of known exploits that they update with newly-discovered exploits in order to provide security against new attacks. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

- For the Asserted Claims of the '356 patent, to the extent the references charted in any of the charts in Exhibit P do not expressly disclose "wherein the packet was received by a honeypot computing device at an unused IP address, and the first, second, third, fourth and fifth program instructions are executed at the honeypot computing device," or similar limitations, it would have been obvious to combine the teachings of any of these charted references with the AAPA and with the knowledge, common sense, and/or creativity of a

person of ordinary skill in the art, the teachings of any other charted references and/or of those references identified in Section III.A, including, for example: Roux, Triulzi. For example, Roux at Abstract ("The hacker may be redirected to a honeypot server (406) arranged to resemble the target computer, to deceive the hacker into thinking the attempted access was successful."). A person of skill in the art would be motivated to combine these references because they each disclose methods, apparatuses, and techniques for detecting intrusions and/or anomalous behavior in order to improve computer security. It is the nature of those of skill in the art of computer security to monitor developments in competing products and combine relevant features and functionalities. A person of skill would have been motivated to look at how other products and systems use a honeypot with unused IP addresses to provide computer security particularly because honeypots are known to provide network security by, for example, having an unused IP address such that any attempt to communicate with the IP address may indicate suspicious or dangerous network traffic. These combinations would have also been obvious to try given that there were a finite number of identified, predictable solutions in the prior art, and there would have been a reasonable expectation of success to simply implement one prior art solution with concepts from prior art in the same field. Moreover, these combinations would have been the result of applying known techniques to a known method that was ready for improvement to yield predictable results.

Consistent with the exemplary motivations to combine each of the references discussed above, the below table provides a summary of the exemplary combinations of references rendering obvious to a POSA, with or without a POSA's knowledge (including the AAPA and general state of the art), skill and/or creativity, the Asserted Claims of the '356 Patent.

Reference	**Reference Alone or in Combination With One or More of the Following**
Afek	Apap, Carpenter, Cooper, Copeland, Desai, Eichstaedt, Fretwell, Gassen, Gluck, Gupta, Jerrim, Mualem, Roux, Stute, Triulzi, Wang

Reference	Reference Alone or in Combination With One or More of the Following
Apap	Afek, Carpenter, Cooper, Copeland, Desai, Eichstaedt, Fretwell, Gassen, Gluck, Gupta, Jerrim, Mualem, Roux, Stute, Triulzi, Wang
Carpenter	Afek, Apap, Cooper, Copeland, Desai, Eichstaedt, Fretwell, Gassen, Gluck, Gupta, Jerrim, Mualem, Roux, Stute, Triulzi, Wang
Cooper	Afek, Apap, Carpenter, Copeland, Desai, Eichstaedt, Fretwell, Gassen, Gluck, Gupta, Jerrim, Mualem, Roux, Stute, Triulzi, Wang
Copeland	Afek, Apap, Carpenter, Cooper, Desai, Eichstaedt, Fretwell, Gassen, Gluck, Gupta, Jerrim, Mualem, Roux, Stute, Triulzi, Wang
Desai	Afek, Apap, Carpenter, Cooper, Copeland, Eichstaedt, Fretwell, Gassen, Gluck, Gupta, Jerrim, Mualem, Roux, Stute, Triulzi, Wang
Eichstaedt	Afek, Apap, Carpenter, Cooper, Copeland, Desai, Fretwell, Gassen, Gluck, Gupta, Jerrim, Mualem, Roux, Stute, Triulzi, Wang
Fretwell	Afek, Apap, Carpenter, Cooper, Copeland, Desai, Eichstaedt, Gassen, Gluck, Gupta, Jerrim, Mualem, Roux, Stute, Triulzi, Wang
Gassen	Afek, Apap, Carpenter, Cooper, Copeland, Desai, Eichstaedt, Fretwell, Gluck, Gupta, Jerrim, Mualem, Roux, Stute, Triulzi, Wang
Gluck	Afek, Apap, Carpenter, Cooper, Copeland, Desai, Eichstaedt, Fretwell, Gassen, Gupta, Jerrim, Mualem, Roux, Stute, Triulzi, Wang
Gupta	Afek, Apap, Carpenter, Cooper, Copeland, Desai, Eichstaedt, Fretwell, Gassen, Gluck, Jerrim, Mualem, Roux, Stute, Triulzi, Wang
Jerrim	Afek, Apap, Carpenter, Cooper, Copeland, Desai, Eichstaedt, Fretwell, Gassen, Gluck, Gupta, Mualem, Roux, Stute, Triulzi, Wang
Mualem	Afek, Apap, Carpenter, Cooper, Copeland, Desai, Eichstaedt, Fretwell, Gassen, Gluck, Gupta, Jerrim, Roux, Stute, Triulzi, Wang
Roux	Afek, Apap, Carpenter, Cooper, Copeland, Desai, Eichstaedt, Fretwell, Gassen, Gluck, Gupta, Jerrim, Mualem, Stute, Triulzi, Wang
Stute	Afek, Apap, Carpenter, Cooper, Copeland, Desai, Eichstaedt, Fretwell, Gassen, Gluck, Gupta, Jerrim, Mualem, Roux, Triulzi, Wang
Triulzi	Afek, Apap, Carpenter, Cooper, Copeland, Desai, Eichstaedt, Fretwell, Gassen, Gluck, Gupta, Jerrim, Mualem, Roux, Stute, Wang
Wang	Afek, Apap, Carpenter, Cooper, Copeland, Desai, Eichstaedt, Fretwell, Gassen, Gluck, Gupta, Jerrim, Mualem, Roux, Stute, Triulzi

V. PATENT L.R. 3-3(C)

Defendant provides herewith, as Exhibits A-P, claim charts identifying where and how specifically in each prior art reference each element of each asserted claim is found, including for each element that Defendant contends is governed by 35 U.S.C. § 112(6), the identity of the

structure(s), act(s), or material(s) in each item of prior art that performs the claimed function. Where Defendant cites to a particular figure in a prior art reference, the citation should be understood to encompass, in addition to the figure itself, the caption and description of the figure as well as any text relating to the figure. Conversely, where a cited portion of text refers to a figure, the citation should be understood to include the figure as well.

Defendant has endeavored to identify portions of the charted prior art that disclose each element of the Asserted Claims of the Patents-in-Suit as required by Patent L.R. 3-3(C). However, the prior art may contain additional disclosure for a particular claim element. To avoid excessive, cumulative citations, Defendant identifies portions of prior art references sufficient to show how the reference discloses the claimed feature. However, Defendant may point to additional evidence from the reference to support its contention that the cited passage discloses the claimed limitation. Persons of ordinary skill in the art of the Patents-in-Suit would determine what is described, disclosed, suggested, and taught by these items of prior art based on the prior art reference as a whole and in the context of relevant publications and literature in the art. Moreover, to understand and interpret any specific description, disclosure, or teaching of an item of prior art, such persons would rely on other information within the prior art item along with other prior art publications and their general scientific or engineering knowledge.

VI. PATENT L.R. 3-3(D)

A. Grounds of Invalidity Under 35 U.S.C. § 112

Defendant lists below the grounds upon which Defendant contends the Asserted Claims of the Patents-in-Suit are invalid for failure to meet one or more of the requirements of 35 U.S.C. § 112. Defendant has not yet taken any depositions related to these issues. Defendant specifically reserves the right to amend and/or supplement these Invalidity Contentions based on subsequently discovered information, and to rely on expert testimony in support of their defenses. Defendant's contentions related to 35 U.S.C. § 112 are informed by and premised on their understanding of

Plaintiff's current infringement contentions and theories and apparent constructions taken by Plaintiff therein.

Patent laws reward innovation with a temporary monopoly. *Festo Corp. v. Shoketsu Kinzoku Kogyo Kabushiki Co.,* 535 U.S. 722, 730 (2002). "The monopoly is a property right; and like any property right, its boundaries should be clear. This clarity is essential to promote progress, because it enables efficient investment in innovation." *Festo*, 535 U.S. at 731–32. As such, the "patent laws require inventors to describe their work in 'full, clear, concise, and exact terms.'" *Id*. at 731. All claim limitations must appear in the specification. *Lockwood v. Am. Airlines, Inc.*, 107 F.3d 1565, 1572 (Fed. Cir. 1997). Among other things, 35 U.S.C. § 112 defines these boundaries by setting forth separate requirements for the patent's written description, the patent's enablement, and the definiteness of the claims.

"The 'written description' requirement [of 35 U.S.C. § 112, ¶ 1] implements the principle that a patent must describe the technology that is sought to be patented; the requirement serves both to satisfy the inventor's obligation to disclose the technologic knowledge upon which the patent is based, and to demonstrate that the patentee was in possession of the invention that is claimed." *Capon v. Eshhar*, 418 F.3d 1349, 1357 (Fed. Cir. 2005). It is for this reason, that "[t]he written description doctrine prohibits new matter from entering into claim amendments, particularly during the continuation process." *Agilent Techs., Inc. v. Affymetrix, Inc.*, 567 F.3d 1366, 1379 (Fed. Cir. 2009). In other words, the written description serves "to prevent an applicant from later asserting that he invented that which he did not." *Amgen Inc. v. Hoechst Marion Roussel, Inc*., 314 F.3d 1313, 1330 (Fed. Cir. 2003).

"The written description requirement is satisfied if the inventor conveys with reasonable clarity to those skilled in the art that, as of the filing date sought, he or she was in possession of the invention, and demonstrates that by disclosure in the specification of the patent." *Quake*, 928 F.3d at 1373. "The invention is, for purposes of the 'written description' inquiry, *whatever is now*

claimed." *Vas-Cath Inc. v. Mahurkar*, 935 F.2d 1555, 1563–64 (Fed. Cir. 1991). That test "requires an objective inquiry into the four corners of the specification from the perspective of a person of ordinary skill in the art." *Ariad Pharms., Inc. v. Eli Lilly & Co.,* 598 F.3d 1336, 1351 (Fed. Cir. 2010). Furthermore, to satisfy the written description requirement, the applicant need not exactly describe the subject matter claimed or use the same terms as used in the claims, but "the specification must contain an equivalent description of the claimed subject matter." *Lockwood*, 107 F.3d at 1572; *see also Vas-Cath*, 935 F.2d at 1563. "A description that merely renders the claimed invention obvious [to a POSITA] does not satisfy the [written description] requirement." *Ariad*, 598 F.3d at 1352 *citing Lockwood*, 107 F.3d at 1571-1572. Accordingly, it is not sufficient that the original disclosure, when combined with the knowledge of a POSITA, would lead a POSITA to speculate as to the modifications that the inventor might have envisioned, but failed to disclose. *D Three Enters., LLC v. SunModo Corp.*, 890 F.3d 1042, 1050-1052 (Fed. Cir. 2018). While the written description requirement does not require that the specification disclose the claimed invention in any particular way, pointing to an "amalgam of disclosures" from which a POSITA could have created the claimed invention does not satisfy this requirement. *Novozymes A/S v. DuPont Nutrition Biosciences APS*, 723 F.3d 1336, 1349 (Fed. Cir. 2013); *accord Ariad*, 598 F.3d at 1352. Instead, the specification must present each claim as an "integrated whole." *Novozymes*, 723 F.3d at 1349. Further, the patent's written description must "sufficiently demonstrate[] that the inventors possessed the full scope of the claimed invention." *Juno Therapeutics, Inc. v. Kite Pharma, Inc.*, 10 F.4d 1330, 1336 (Fed. Cir. 2021).

Defendant contends that Taasera's apparent claim constructions, to the extent understandable from its infringement contentions, render many terms and phrases of the asserted claims well beyond the scope of the purported inventions described in the Patents-in-Suit. Taasera's infringement contentions, to the extent understandable, attempt to construe the terms and phrases of the Patents-in-Suit in a manner that is inconsistent with the written specifications

and prosecution histories of the Patents-in-Suit as well as with the understanding of one of ordinary skill in the art at the time the applications that issued as the Patents-in-Suit were filed, as discussed specifically below. These constructions taken in the infringement contentions are inconsistent with the plain language of the claims, the supporting description, and the prosecution history. As set forth in more detail below, to the extent any of the Asserted Claims are found to cover one or more of the accused products identified in Taasera's infringement contentions, such Asserted Claims are also invalid under 35 U.S.C. § 112, ¶ 1.

The enablement requirement refers to the requirement of 35 U.S.C. § 112, ¶ 1 that the specification describe how to make and how to use the invention. The invention, that one skilled in the art must be enabled to make and use, is that defined by the claim(s) of the particular patent. The purpose of the enablement requirement is to ensure that the invention is communicated to the interested public in a meaningful way. The standard for determining whether the specification meets the enablement requirement was cast in the Supreme Court decision of *Minerals Separation Ltd. v. Hyde,* 242 U.S. 261, 270 (1916) which postured the question: is the experimentation needed to practice the invention undue or unreasonable? That standard is still the one to be applied. *In re Wands,* 858 F.2d 731, 737, 8 USPQ2d 1400, 1404 (Fed. Cir. 1988). Accordingly, even though the statute does not use the term "undue experimentation," it has been interpreted to require that the claimed invention be enabled so that any person skilled in the art can make and use the invention without undue experimentation. *In re Wands,* 858 F.2d at 737, 8 USPQ2d at 1404 (Fed. Cir. 1988); *see also United States v. Telectronics, Inc.,* 857 F.2d 778, 785, 8 USPQ2d 1217, 1223 (Fed. Cir. 1988). To satisfy the enablement requirement, a patent specification need not teach what is well known in the art. *In re Buchner,* 929 F.2d 660, 661, 18 USPQ2d 1331, 1332 (Fed. Cir. 1991); *Hybritech, Inc. v. Monoclonal Antibodies, Inc.,* 802 F.2d 1367, 1384, 231 USPQ 81, 94 (Fed. Cir. 1986), *cert. denied,* 480 U.S. 947 (1987); and *Lindemann Maschinenfabrik GMBH v. American Hoist & Derrick Co.,* 730 F.2d 1452, 1463, 221 USPQ 481, 489 (Fed. Cir. 1984). There are several

factors to be considered when determining whether there is sufficient evidence to support a determination that a patent disclosure does or does not satisfy the enablement requirement and whether any necessary experimentation is "undue." These factors include, but are not limited to: (a) The breadth of the claims; (b) The nature of the invention; (c) The state of the prior art; (d) The level of one of ordinary skill; (e) The level of predictability in the art; (f) The amount of direction provided by the inventor; (g) The existence of working examples; and (h) The quantity of experimentation needed to make or use the invention based on the content of the disclosure. *Wands*, 858 F.2d at 737.

Written description and enablement are separate requirements for patentability under 35 U.S.C. § 112, ¶ 1. *Ariad*, 598 F.3d at 1351. Written description is about whether a POSITA can recognize that what is claimed corresponds to what is described; it is ***not*** about whether the patentee has proven to a POSITA that the invention works, or how to make it work, which is an enablement question. *CenTrak, Inc. v. Sonitor Techs., Inc.*, 915 F.3d 1360, 1366 (Fed. Cir. 2019); *see also Alcon Research Ltd. v. Barr Labs., Inc.*, 745 F.3d 1180, 1191 (Fed. Cir. 2014).

The intent of 35 U.S.C. § 112, ¶ 2, and the "definiteness" requirement, is to preserve and protect the public notice function served by claims. In the seminal *Nautilus* decision, the Supreme Court set forth the relationship between the public notice function and the "definiteness" requirement:

> [A] patent must be precise enough to afford clear notice of what is claimed, thereby "appris[ing] the public of what is still open to them." … Otherwise there would be "[a] zone of uncertainty which enterprise and experimentation may enter only at the risk of infringement claims." … And absent a meaningful definiteness check, we are told, patent applicants face powerful incentives to inject ambiguity into their claims … Eliminating that temptation is in order, and "the patent drafter is in the best position to resolve the ambiguity in … patent claims." … Cognizant of the competing concerns, we read §112, ¶ 2 to require that a patent's claims, viewed in light of the specification and prosecution history, inform those skilled in the art about the scope of the invention with reasonable certainty … It cannot be sufficient that a court can ascribe *some* meaning to a patent's claims; the definiteness inquiry trains on the understanding of a skilled artisan at the time of the patent application, not that of a court viewing matters *post hoc*. To tolerate imprecision just short of that rendering a claim "insolubly

> ambiguous" would diminish the definiteness requirement's public-notice function and foster the innovation-discouraging "zone of uncertainty," … .

Nautilus, Inc. v. Biosig Instruments, Inc., 572 U.S. 898, 909-911 (2014) (emphasis in original) (internal citations omitted) (the Supreme Court also expressly overruling the Federal Circuit's prior "insolubly ambiguous" standard and expressly abrogating several district court decisions wrongly decided under this prior standard). A decision on whether a claim is indefinite under 35 U.S.C. 112(b) or pre-AIA 35 U.S.C. 112, second paragraph requires a determination of whether those skilled in the art would understand what is claimed when the claim is read in light of the specification. *Power-One, Inc. v. Artesyn Techs., Inc.,* 599 F.3d 1343, 1350, 94 USPQ2d 1241, 1245 (Fed. Cir. 2010).

As further described below, certain Asserted Claims are invalid because the specifications of the Patents-in-Suit do not include sufficient written description of the purported inventions allegedly claimed in the Patents-in-Suit, and the manner and process of making and using it, in such full, clear, concise and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use any of the allegedly claimed inventions, 35 U.S.C. § 112, ¶ 1. Moreover, the patent applications that issued as the Patents-in-Suit did not contain a written description of the purported invention sufficiently clear and complete to enable one of ordinary skill in the art to which the purported invention pertains to make and use the invention as claimed without undue experimentation. *Id.*; *see also, e.g.*, *Telectronics, Inc.*, 857 F.2d at 785. Certain Asserted Claims are also invalid under 35 U.S.C. § 112, ¶ 1 because their full scope is not enabled by the specifications of the relevant Patents-in-Suit. *Id*. Certain Asserted Claims are also invalid under 35 U.S.C. § 112, ¶ 1 because there is insufficient written description of the invention to demonstrate that purported inventor(s) of the Asserted Claims possessed the inventions set forth in the Asserted Claims. 35 U.S.C. § 112, ¶ 1; *see also, e.g.*, *Moba, B.V. v. Diamond Automation, Inc.,* 325 F.3d 1306, 1319, 66 USPQ2d 1429, 1438 (Fed. Cir. 2003). To the extent the below-listed limitations are definite (under 35 U.S.C. § 112, second paragraph), the

applications that became the Asserted Patents fail to sufficiently describe or enable them as required by 35 U.S.C. § 112, first paragraph:[32]

Certain Asserted Claims are also invalid because, under the apparent constructions and infringement theories set forth by Plaintiff, the claims do not particularly point out and distinctly claim the subject matter which the applicant(s) regards as the invention as required by 35 U.S.C. § 112, ¶ 2. *See, e.g.*, *Power-One*, 599 F.3d at 1350

1. '796 Patent

Term	Relevant Claim(s)
"[A method of/Apparatus for/An article of manufacture for use in] automatically processing an input sequence of data symbols"	All asserted claims
"[identify/identifying] at least one regularly identifiable expression in the input sequence of data symbols, wherein the at least one regularly identifiable expression represents a pattern that is matchable in accordance with a programming language that supports such a regularly identifiable expression"	All asserted claims
"[identify/identifying] at least a portion of information associated with the at least one regularly identifiable expression"	All asserted claims
"extract/extracting the portion of information"	All asserted claims
"comparing the input sequence of data symbols to one or more previously-stored regularly identifiable expressions to determine if there is a match between a portion of the input sequence and at least one of the previously-stored regularly identifiable expression"	2, 13
"the extracted portion of information is used to take a specified action"	7, 18
"wherein the input sequence of data symbols is representative of at least one of text data, transcribed spoken data, deoxyribonucleic acid sequence data, ribonucleic acid sequence data, amino-acid sequence data, audio sequence data, and video sequence data"	10, 21

[32] Listed claims also include all Asserted Claims dependent thereon, even if those claims are not individually listed.

a. "[A method of/Apparatus for/An article of manufacture for use in] automatically processing an input sequence of data symbols" (all asserted claims)

All asserted claims of the '796 Patent are invalid as indefinite because a person of ordinary skill in the art would not be able to ascertain with reasonable certainty what steps or elements of each claim need to be "automatic."

b. "[identify/identifying] at least one regularly identifiable expression in the input sequence of data symbols, wherein the at least one regularly identifiable expression represents a pattern that is matchable in accordance with a programming language that supports such a regularly identifiable expression" (all asserted claims)

All asserted claims of the '796 Patent are invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue experimentation; (2) the specification does not describe either the claim element individually or the claim as a whole in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

c. "[identify/identifying] at least one regularly identifiable expression in the input sequence of data symbols, wherein the at least one regularly identifiable expression represents a pattern that is matchable in accordance with a programming language that supports such a regularly identifiable expression" (all asserted claims)

All asserted claims of the '796 Patent are invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue experimentation; (2) the specification does not describe either the claim element individually or the claim as a whole in sufficient detail that one skilled in the art can reasonably

conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

d. "[identify/identifying] at least a portion of information associated with the at least one regularly identifiable expression" (all asserted claims)

All asserted claims of the '796 Patent are invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue experimentation; (2) the specification does not describe either the claim element individually or the claim as a whole in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

e. "[extract/extracting] the portion of information" (all asserted claims)

All asserted claims of the '796 Patent are invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue experimentation; (2) the specification does not describe either the claim element individually or the claim as a whole in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

f. "comparing the input sequence of data symbols to one or more previously-stored regularly identifiable expressions to determine if there is a match between a portion of the input sequence and at least one of the previously-stored regularly identifiable expression" (claims 2 and 13)

Claims 2 and 13 of the '796 Patent are invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue experimentation; (2) the specification does not describe either the claim element individually or the claim as a whole in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

g. "the extracted portion of information is used to take a specified action" (claims 7 and 18)

Claims 7 and 18 of the '796 Patent are invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue experimentation; (2) the specification does not describe either the claim element individually or the claim as a whole in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

h. "wherein the input sequence of data symbols is representative of at least one of text data, transcribed spoken data, deoxyribonucleic acid sequence data, ribonucleic acid sequence

data, amino-acid sequence data, audio sequence data, and video sequence data" (claims 10 and 21)

Claims 10 and 21 of the '796 Patent are invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification of the '796 Patent lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue experimentation; (2) the specification does not describe either the claim element individually or the claim as a whole in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

2. '137 Patent

Term	Relevant Claim(s)
"a validation module coupled to *the pre-execution monitor* operable for determining whether the program is valid" and "a detection module coupled to *the pre-execution monitor* operable for intercepting a trigger from the computing device's operating system."	1-4
"an execution module coupled to the detection module and operable for monitoring, at the operating system kernel of the computing device, the program in response to the trigger intercepted by the detection module"	1-4
"validation data"	3
"wherein the step of monitoring the new program while it executes is performed at the operating system kernel of the computing device"	6-13
"if the new program is validated, permitting the new program to continue loading and to execute in connection with the computing device"	6-13
"if the new program is not validated, monitoring the new program while it loads and executes in connection with the computing device, wherein the step of monitoring the new program while it executes is performed at the operating system kernel of the computing device"	6-13
"wherein the step of monitoring the new program while allowing it to execute is performed at the operating system kernel of the computing device"	14, 16-18, 22, 24-25, 29

Term	Relevant Claim(s)
"if the new program is the same as the allowed program, permitting the new program to execute on the computing device"	14, 16-18, 22, 24
"if the new program is not the same as the allowed program, monitoring the new program while allowing it to execute on the computing device"	14, 16-18, 22, 24
"if the new program is valid, permitting the new program to execute on the computing device"	25, 29
"if the new program is not valid, monitoring the new program while allowing it to execute on the computing device, wherein the step of monitoring the new program while allowing it to execute is performed at the operating system kernel of the computing device"	25, 29
"a pre-execution module operable for receiving notice from the computing device's operating system that a new program is being loaded onto the computing device"	1-4
"a validation module coupled to the pre-execution monitor operable for determining whether the program is valid"	1-4
"a detection module coupled to the pre-execution monitor operable for intercepting a trigger from the computing device's operating system"	1-4
"an execution module coupled to the detection module and operable for monitoring, at the operating system kernel of the computing device, the program in response to the trigger intercepted by the detection module"	1-4

a. "a validation module coupled to *the pre-execution monitor* operable for determining whether the program is valid" and "a detection module coupled to *the pre-execution monitor* operable for intercepting a trigger from the computing device's operating system." (claims 1-4)

Claims 1-4 of the '137 Patent are invalid as indefinite because claim 1 refers to "a validation module coupled to *the pre-execution monitor* operable for determining whether the program is valid" and "a detection module coupled to *the pre-execution monitor* operable for intercepting a trigger from the computing device's operating system," but there is no antecedent basis for "the pre-execution monitor," which is required to be coupled to the detection module and the validation module. Nor does the surrounding claim language provide context as to what "the

pre-execution monitor" does. A person of skill would thus be unable to ascertain the scope of the claim with reasonable certainty.

b. "an execution module coupled to the detection module and operable for monitoring, at the operating system kernel of the computing device, the program in response to the trigger intercepted by the detection module" (claims 1-4)

Claims 1-4 of the '137 Patent are invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue experimentation; (2) the specification does not describe either the claim element individually or the claim as a whole in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

c. "validation data" (claim 3)

Claim 3 is invalid as indefinite because the specification does not use the term "validation data" and a person of ordinary skill would be unable to discern the meaning of the term "validation data" with reasonable certainty.

d. "wherein the step of monitoring the new program while it executes is performed at the operating system kernel of the computing device" (claims 6-13)

Claims 6-13 of the '137 Patent are invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue experimentation; (2) the specification does not describe either the claim element individually or the claim as a whole in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

e. "if the new program is validated, permitting the new program to continue loading and to execute in connection with the computing device" (claims 6-13)

Claims 6-13 of the 137 Patent are invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue experimentation; (2) the specification does not describe either the claim element individually or the claim as a whole in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

f. "if the new program is not validated, monitoring the new program while it loads and executes in connection with the computing device, wherein the step of monitoring the new program while it executes is performed at the operating system kernel of the computing device" (claims 6-13)

Claims 6-13 of the '137 Patent are invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue experimentation; (2) the specification does not describe either the claim element individually or the claim as a whole in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

g. "wherein the step of monitoring the new program while allowing it to execute is performed at the operating system

kernel of the computing device" (claims 14, 16-18, 22, 24-25, and 29)

Claims 14, 16-18, 22, 24-25, and 29 of the '137 Patent are invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue experimentation; (2) the specification does not describe either the claim element individually or the claim as a whole in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

h. "if the new program is the same as the allowed program, permitting the new program to execute on the computing device" (claims 14, 16-18, 20, 22, and 24)

Claims 14, 16-18, 22, and 24 of the '137 Patent are invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue experimentation; (2) the specification does not describe either the claim element individually or the claim as a whole in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

i. "if the new program is not the same as the allowed program, monitoring the new program while allowing it to execute on the computing device" (claims 14, 16-18, 22, and 24)

Claims 14, 16-18, 22, and 24 of the '137 Patent are invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue experimentation; (2) the specification does not describe either the claim element

individually or the claim as a whole in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

j. "if the new program is not valid, monitoring the new program while allowing it to execute on the computing device, wherein the step of monitoring the new program while allowing it to execute is performed at the operating system kernel of the computing device" (claims 25 and 29)

Claims 25 amd 29 of the '137 Patent are invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue experimentation; (2) the specification does not describe either the claim element individually or the claim as a whole in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

k. "if the new program is valid, permitting the new program to execute on the computing device" (claims 25 and 29)

Claims 25 and 29 of the '137 Patent are invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue experimentation; (2) the specification does not describe either the claim element individually or the claim as a whole in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions,

a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

l. Indefinite Means-Plus-Function Claim Elements

Defendant reserves the right to argue that each of the following is a means-plus-function term that is indefinite because it does not disclose an adequate corresponding structure or algorithm for performing the claimed function:

- "a pre-execution module operable for receiving notice from the computing device's operating system that a new program is being loaded onto the computing device;"
- "a validation module coupled to the pre-execution monitor operable for determining whether the program is valid;"
- "a detection module coupled to the pre-execution monitor operable for intercepting a trigger from the computing device's operating system;"
- "an execution module coupled to the detection module and operable for monitoring, at the operating system kernel of the computing device, the program in response to the trigger intercepted by the detection module."

3. '441 Patent

Term	Relevant Claim(s)
"a runtime execution context indicating attributes of the application at runtime, wherein the attributes comprise one or more executable file binaries of the application and loaded components of the application"	1–7
"an execution analysis of the one or more executable file binaries and the loaded components"	1–7
"parent-child process associations of the application"	3
"a static, dynamic, or virtual analysis of the one or more executable file binaries and the loaded components"	4

a. "a runtime execution context indicating attributes of the application at runtime, wherein the attributes comprise one or

more executable file binaries of the application and loaded components of the application"

Claims 1-7 of the '441 Patent are invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue experimentation; (2) the specification does not describe either the claim element individually or the claim as a whole in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

b. "an execution analysis of the one or more executable file binaries and the loaded components"

Claims 1-7 of the '441 Patent are invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue experimentation; (2) the specification does not describe either the claim element individually or the claim as a whole in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

c. "parent-child process associations of the application" (claim 3)

Claim 3 of the '441 Patent is invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue

experimentation; (2) the specification does not describe either the claim element individually or the claim as a whole in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

d. "a static, dynamic, or virtual analysis of the one or more executable file binaries and the loaded components" (claim 4)

Claims 4 of the '441 Patent is invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue experimentation; (2) the specification does not describe either the claim element individually or the claim as a whole in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

4. '517 Patent

Term	Relevant Claim(s)
"identifying, using at least one assessment policy, a runtime risk for an application program that executes on a device"	All asserted claims
"wherein the identified runtime risk indicates a risk or threat of the identified action sequence of the application"	All asserted claims
"identifying, by a runtime monitor including a processing device, a behavior score for the application program that executes on the device based on the identified runtime risk" / "identifying a behavior score for the application program that executes on the device based on the identified runtime risk"	All asserted claims
"assessing, by the processing device, that the at least two performed actions of the action sequence describe a forensic chain of events based on a plurality of factors" / "the processing device assesses that the at least two performed actions	12, 24

of the action sequence describe a forensic chain of events based on a plurality of factors"	
Entire claim (mixing statutory subject matter)	13-17, 24

a. "identifying, using at least one assessment policy, a runtime risk for an application program that executes on a device" (all asserted claims)

All asserted claims of the '517 Patent are invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue experimentation; (2) the specification does not describe either the claim element individually or the claim as a whole in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

b. "wherein the identified runtime risk indicates a risk or threat of the identified action sequence of the application" (all asserted claims)

All asserted claims of the '517 Patent are invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue experimentation; (2) the specification does not describe either the claim element individually or the claim as a whole in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

c. "identifying, by a runtime monitor including a processing device, a behavior score for the application program that executes on the device based on the identified runtime risk" / "identifying a behavior score for the application program that executes on the device based on the identified runtime risk" (all asserted claims)

All asserted claims of the '517 Patent are invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue experimentation; (2) the specification does not describe either the claim element individually or the claim as a whole in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

d. "assessing, by the processing device, that the at least two performed actions of the action sequence describe a forensic chain of events based on a plurality of factors" / the processing device assesses that the at least two performed actions of the action sequence describe a forensic chain of events based on a plurality of factors" (claims 12 and 24)

Claims 12 and 24 of the '517 Patent are invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue experimentation; (2) the specification does not describe either the claim element individually or the claim as a whole in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

e. Mixing Statutory Subject Matter

Independent claim 13 of the ’517 Patent and its dependent claims are all invalid for indefiniteness because they mix the statutory subject matter of apparatus limitations and method steps into a single claim, and because they leave a person of ordinary skill in the art unable to determine when infringement would occur with reasonable certainty. Claim 13 recites as follows:

> 13. A system for assessing runtime risk for an application program that executes on a device, comprising:
>
> a rules database storing a plurality of rules, wherein each rule identifies an action sequence;
>
> a policy database storing a plurality of assessment policies, wherein each assessment policy includes at least one rule of the plurality of rules; and
>
> a runtime monitor including a processing device
>
> ***identifying***, using at least one assessment policy, a runtime risk for an application program that executes on a device, wherein the identified runtime risk indicates a risk or threat of the identified action sequence of the application, and
>
> ***identifying*** a behavior score for the application program that executes on the device based on the identified runtime risk, wherein
>
> the action sequence is a sequence of at least two performed actions, and
> each performed action is at least one of: a user action, an application action, and a system action.

As shown above, claim 13 recites “a system” but goes on to recite steps of “identifying, using at least one assessment policy, a runtime risk for an application program that executes on a device, wherein the identified runtime risk indicates a risk or threat of the identified action sequence of the application” and “identifying a behavior score for the application program that executes on the device based on the identified runtime risk.”

5. ’038 Patent

Term	Relevant Claim(s)

"identifying, from the plurality of policies, a plurality of operating conditions on the endpoint to monitor"	1-22
"configuring one or more software agents on the endpoint to monitor the plurality of operating conditions"	1-22
"one or more software agents on the endpoint configured to monitor a plurality of operating conditions identified in the plurality of policies"	23-33
"determining, by the computing system, a compliance state of the endpoint based on the status information"	1-22
"one or more hardware processors at the computing system configured to … determine a compliance state of the endpoint based on the status information"	23-33

a. "identifying, from the plurality of policies, a plurality of operating conditions on the endpoint to monitor" "configuring one or more software agents on the endpoint to monitor the plurality of operating conditions," and "one or more software agents on the endpoint configured to monitor a plurality of operating conditions identified in the plurality of policies"

Asserted claims 1-22 of the '038 Patent recite, among other limitations, "identifying, from the plurality of policies, a plurality of operating conditions on the endpoint to monitor" and "configuring one or more software agents on the endpoint to monitor the plurality of <u>operating</u> conditions", while claims 23-33 require "one or more software agents on the endpoint configured to monitor a plurality of operating conditions identified in the plurality of policies," but there is inadequate support in the specification for these claim terms. In particular, there is inadequate support in the specification of how a plurality of *operating conditions* on the endpoint to monitor are identified from the plurality of policies in the data store on the remote computing system. And, in particular, there is no support in the specification of how one or more software agents on the endpoint are configured to monitor the plurality of *operating conditions* on the endpoint that were identified from the plurality of policies in the data store on the remote computing system. During prosecution of the application leading to the '038 Patent, the Patent Office examiner rejected the then-pending claims as anticipated by U.S. Pat. Pub. No. 2005/0086511 to Balacheff *et al.* ("Balacheff"), and where the then-pending claims recited, among other things:

> identifying, from the plurality of policies, a plurality of *conditions* on the endpoint to monitor;
> configuring one or more software agents on the endpoint to monitor the plurality of *conditions*;

> receiving, across a network, at the computing system, status information about the plurality of *conditions* on the endpoint gathered by the one or more software agents;

'038 Patent File History at March 4, 2014 Non-Final Office Action at pp. 2-4, 7-8.

In response to this rejection, the Applicant distinguished these (and related) claim limitations from the cited prior art's (Balacheff's) disclosure by arguing:

> In contrast [to these limitations], the integrity metrics and logs of Balacheff are <u>not</u> taught as configurable, nor does Balacheff teach that the conditions monitored to create an integrity metric are configurable or are identified from the plurality of policies. *See, e.g.*, Balacheff at [0042-44]. Rather than teaching configurable software agents that assist in controlling the operation of endpoints, Balacheff teaches sending logs or integrity metrics upon request.

'038 Patent File History at June 4, 2014 Amendment at p. 8; March 4, 2014 Non-Final Office Action at pp. 3-4, 7-8; Balacheff at paras [0042]-[0044] (discussing that conditions monitored on the endpoint (computer 20) to create an "integrity metric" include "the BIOS", "current BIOS version", "identity and version number of each procedure, application, DLL and so on that is running"), FIG. 1.

Subsequent to receiving the Applicant's amendment and arguments, in an interview with the Applicant, the Patent Office examiner proposed an additional claim amendment, which required that the recited "plurality of conditions on the endpoint to monitor" limitation in the independent claims of the application leading to the '038 patent be narrowed to recite "*operating* conditions" (as shown below):

> identifying, from the plurality of policies, a plurality of <u>operating</u> conditions on the endpoint to monitor;
> configuring one or more software agents on the endpoint to monitor the plurality of <u>operating</u> conditions;
> receiving, across a network, at the computing system, status information about the plurality of <u>operating</u> conditions on the endpoint gathered by the one or more software agents;

'038 Patent File History at Aug. 28, 2014 Notice of Allowance at pp. 2-4; Summary of Examiner-Initiated Aug. 20, 2014 Interview with Applicant. The Applicant agreed to narrow these claim limitations to recite "*operating* conditions", instead of just conditions, in order to gain allowance of the claims. *Id*. In the resulting Notice of Allowance, the examiner stated that the claims of the application leading to the '038 Patent were allowed "in view of the examiner's amendment and

for reasons argued by applicant in pages 7-8 of the Remarks, filed on June 06, 2014." '038 Patent File History at Aug. 28, 2014 Notice of Allowance at pp. 4-6.

The specification of the '038 patent, however, (i) does not use the term "operating condition", (ii) describes "conditions 104F" that can be monitored by software agents within endpoint system 104, and (iii) includes a list of "illustrative conditions 104F" spanning 5+ columns without specifying which would constitute an "*operating* condition" vs. a "*non-operating* condition." *See, e.g.*, '038 Patent at FIG. 1, 7:64-8:3, 9:25-15:14. Thus, a POSITA would not understand the meaning or scope of the term "operating condition" nor how to perform "identifying, from the plurality of policies, a plurality of <u>operating</u> conditions on the endpoint to monitor", nor how to perform "configuring one or more software agents on the endpoint to monitor the plurality of <u>operating</u> conditions." Therefore, claims 1-22 of the '038 Patent, which each incorporate the "operating conditions" term, are invalid as indefinite, lacking enablement, and/or lacking written description in violation of 35 U.S.C. § 112. *See, e.g.*, *Moba*, 325 F.3d at 1319; *Telectronics*, 857 F.2d at 785; *Power-One*, 599 F.3d at 1350.

b. "determining, by the computing system, a compliance state of the endpoint based on the status information"

Asserted claims 1-22 recite, amongst other things, "determining, by the computing system, a compliance state of the endpoint based on the status information," but there is inadequate support in the specification for this term. In particular, there is inadequate support in the specification of how a remote computing system makes a determination of a compliance state of the endpoint based on the status information. Since there is not sufficient detail in the specification to support this limitation for the term "determining, by the computing system, a compliance state of the endpoint based on the status information" and its meaning cannot be sufficiently ascertained, claims 1-22 are invalid as indefinite, lacking enablement, and/or lacking written description in violation of 35 U.S.C. § 112.

c. "determine a compliance state of the endpoint based on the status information"

Asserted claims 23-33 of the '038 Patent recite, amongst other things, "one or more hardware processors at the computing system configured to … determine a compliance state of the endpoint based on the status information," but there is inadequate support in the specification for

this term. In particular, there is inadequate support in the specification of how a remote computing system makes a determination of a compliance state of the endpoint based on the status information. Since there is not sufficient detail in the specification to support this limitation for the term "determine a compliance state of the endpoint based on the status information", and its meaning cannot be sufficiently ascertained, claims 23-33 are invalid as indefinite, lacking enablement, and/or lacking written description in violation of 35 U.S.C. § 112.

Moreover, to the extent Defendant understands Taasera's infringement contentions, Taasera's apparent interpretation of the Asserted Claims of the '038 Patent is that they are broad enough to encompass, e.g., a server that simply collects data from an endpoint as "determin[ing . . .]a compliance state of the endpoint based on the status information and plurality of compliance policies in the data store"; an application on an endpoint taking actions on its own as "initiat[ing, by the computing system,] based on the compliance state, an action identified in at least one rule in the data store"; and blocking exfiltration attempts or a network connection as "controlling access of the endpoint to computing resources." Under these overly broad interpretations of the Asserted Claims, each of the Asserted Claims of the '038 Patent is invalid for lack of written description and/or is not enabled.

6. '918 Patent

Term	Relevant Claim(s)
"identifying, from the plurality of policies, a plurality of operating conditions on the endpoint to evaluate"	1-16
"configuring one or more software services provided by an operating system on the endpoint to monitor the plurality of operating conditions"	1-16
"one or more software services provided by an operating system on the endpoint configured to monitor a plurality of operating conditions identified in the plurality of policies"	17-24
"determining, by the computing system, a compliance state of the endpoint based on the user information and status information"	1-16
"determine, by the computing system, a compliance state of the endpoint based on the user information and status information"	17-24
"configuring one or more applications running on the endpoint on the endpoint to monitor at least a subset of the plurality of operating conditions"	5, 13, 21

a. "identifying, from the plurality of policies, a plurality of operating conditions on the endpoint to evaluate" and "configuring one or more software services provided by an

operating system on the endpoint to monitor the plurality of operating conditions"

Asserted claims 1-16 of the '918 Patent recite, among other limitations, "identifying, from the plurality of policies, a plurality of operating conditions on the endpoint to monitor" and "configuring one or more software services provided by an operating system on the endpoint to monitor the plurality of operating conditions", but there is inadequate support in the specification for these claim terms. In particular, there is inadequate support in the specification of how a plurality of *operating conditions* on the endpoint to monitor are identified from the plurality of policies in the data store on the remote computing system. And, in particular, there is inadequate support in the specification of how one or more software *services provided by an operating system* on the endpoint are configured to monitor the plurality of *operating conditions* on the endpoint that were identified from the plurality of policies in the data store on the remote computing system.

During prosecution of the application leading to the '918 Patent, the Applicant filed a terminal disclaimer to overcome the Patent Office examiner's non-statutory, obviousness-type double patenting rejection of the claims over claims 1-33 of the '038 Patent, and claims 1-30 of the '997 Patent. '918 Patent File History at June 14, 2017 Non-Final Rejection at p. 2; Aug. 21, 2017 Amendment and Response at p. 2; Aug. 21, 2017 Terminal Disclaimer; Aug. 21, 2017 Approval of Terminal Disclaimer; Nov. 8, 2017 Notice of Allowance at pp. 2-3. During prosecution of the application leading to the '038 Patent, to which the '918 and '997 Patents are terminally disclaimed, the Patent Office examiner rejected the then-pending claims as anticipated by U.S. Pat. Pub. No. 2005/0086511 to Balacheff *et al.* ("Balacheff"), and where the then-pending claims recited, among other things:

> identifying, from the plurality of policies, a plurality of *conditions* on the endpoint to monitor;
> configuring one or more software agents on the endpoint to monitor the plurality of *conditions*;
> receiving, across a network, at the computing system, status information about the plurality of *conditions* on the endpoint gathered by the one or more software agents;

'038 Patent File History at March 4, 2014 Non-Final Office Action at pp. 2-4, 7-8.

In response to this rejection, the Applicant distinguished these (and related) claim limitations from the cited prior art's (Balacheff's) disclosure by arguing:

> In contrast [to these limitations], the integrity metrics and logs of Balacheff are <u>not</u> taught as configurable, nor does Balacheff teach

> that the conditions monitored to create an integrity metric are configurable or are identified from the plurality of policies. *See, e.g.*, Balacheff at [0042-44]. Rather than teaching configurable software agents that assist in controlling the operation of endpoints, Balacheff teaches sending logs or integrity metrics upon request.

'038 Patent File History at June 4, 2014 Amendment at p. 8; March 4, 2014 Non-Final Office Action at pp. 3-4, 7-8; Balacheff at paras [0042]-[0044] (discussing that conditions monitored on the endpoint (computer 20) to create an "integrity metric" include "the BIOS", "current BIOS version", "identity and version number of each procedure, application, DLL and so on that is running"), FIG. 1.

Subsequent to receiving the Applicant's amendment and arguments, in an interview with the Applicant, the Patent Office examiner proposed an additional claim amendment, which required that the recited "plurality of conditions on the endpoint to monitor" limitation in the independent claims of the application leading to the '038 Patent be narrowed to recite "*operating* conditions" (as shown below):

> identifying, from the plurality of policies, a plurality of <u>operating</u> conditions on the endpoint to monitor;
> configuring one or more software agents on the endpoint to monitor the plurality of <u>operating</u> conditions;
> receiving, across a network, at the computing system, status information about the plurality of <u>operating</u> conditions on the endpoint gathered by the one or more software agents;

'038 Patent File History at Aug. 28, 2014 Notice of Allowance at pp. 2-4; Summary of Examiner-Initiated Aug. 20, 2014 Interview with Applicant. The Applicant agreed to narrow these claim limitations to recite "*operating* conditions", instead of just conditions, in order to gain allowance of the claims. *Id*. In the resulting Notice of Allowance, the examiner stated that the claims of the application leading to the '038 Patent were allowed "in view of the examiner's amendment and for reasons argued by applicant in pages 7-8 of the Remarks, filed on June 06, 2014." '038 Patent File History at Aug. 28, 2014 Notice of Allowance at pp. 4-6.

The shared specification of the '038 Patent, '997 Patent and '918 Patent, however, (i) does not use the term "operating condition", (ii) describes "conditions 104F" that can be monitored by software agents within endpoint system 104, and (iii) includes a list of "illustrative conditions 104F" spanning 5+ columns without specifying which would constitute an "*operating* condition" vs. a "*non-operating* condition." *See, e.g.*, '038 Patent at FIG. 1, 7:64-8:3, 9:25-15:14. Thus, a

POSITA would not understand the meaning or scope of the term "operating condition" nor how to perform "identifying, from the plurality of policies, a plurality of operating conditions on the endpoint to monitor", nor how to perform "configuring one or more software services provided by an operating system on the endpoint to monitor the plurality of operating conditions." Therefore, claims 1-16 of the '918 Patent, which each incorporate the "operating conditions" term, are invalid as indefinite, lacking enablement, and/or lacking written description in violation of 35 U.S.C. § 112. *See, e.g.*, *Moba*, 325 F.3d at 1319; *Telectronics*, 857 F.2d at 785; *Power-One*, 599 F.3d at 1350.

b. "one or more software services provided by an operating system on the endpoint configured to monitor a plurality of operating conditions identified in the plurality of policies"

Asserted claims 17-24 of the '918 Patent recite, among other limitations, "one or more software services provided by an operating system on the endpoint configured to monitor a plurality of operating conditions identified in the plurality of policies", but there is inadequate support in the specification for this claim term. In particular, there is inadequate support in the specification of a software *service provided by an operating system* on the endpoint being configured to monitor a plurality of *operating conditions* identified in the plurality of policies in the data store on the remote computing system.

During prosecution of the application leading to the '918 Patent, the Applicant filed a terminal disclaimer to overcome the Patent Office examiner's non-statutory, obviousness-type double patenting rejection of the claims over claims 1-33 of the '038 Patent, and claims 1-30 of the '997 Patent. '918 Patent File History at June 14, 2017 Non-Final Rejection at p. 2; Aug. 21, 2017 Amendment and Response at p. 2; Aug. 21, 2017 Terminal Disclaimer; Aug. 21, 2017 Approval of Terminal Disclaimer; Nov. 8, 2017 Notice of Allowance at pp. 2-3. During prosecution of the application leading to the '038 Patent, to which the '918 and '997 Patents are terminally disclaimed, the Patent Office examiner rejected the then-pending claims as anticipated by U.S. Pat. Pub. No. 2005/0086511 to Balacheff *et al.* ("Balacheff"), and where the then-pending claims recited, among other things, "one or more software agents on the endpoint configured to monitor a plurality of *conditions* identified in the plurality of policies." '038 Patent File History at March 4, 2014 Non-Final Office Action at pp. 10-11. In response to this rejection, the Applicant distinguished these (and related) claim limitations from the cited prior art's (Balacheff's) disclosure by arguing:

> In contrast [to these limitations], the integrity metrics and logs of Balacheff are not taught as configurable, nor does Balacheff teach that the conditions monitored to create an integrity metric are configurable or are identified from the plurality of policies. *See, e.g.*, Balacheff at [0042-44]. Rather than teaching configurable software agents that assist in controlling the operation of endpoints, Balacheff teaches sending logs or integrity metrics upon request.

'038 Patent File History at June 4, 2014 Amendment at p. 8; March 4, 2014 Non-Final Office Action at pp. 3-4; Balacheff at paras [0042]-[0044] (discussing that conditions monitored on the endpoint (computer 20) to create an "integrity metric" include "the BIOS", "current BIOS version", "identity and version number of each procedure, application, DLL and so on that is running"), FIG. 1.

Subsequent to receiving the Applicant's amendment and arguments, in an interview with the Applicant, the Patent Office examiner proposed an additional claim amendment, which required that the recited "plurality of conditions on the endpoint to monitor" limitation in the independent claims of the application leading to the '038 Patent be narrowed to recite "*operating* conditions" as follows: "one or more software agents on the endpoint configured to monitor a plurality of operating conditions identified in the plurality of policies." '038 Patent File History at Aug. 28, 2014 Notice of Allowance at pp. 2-4; Summary of Examiner-Initiated Aug. 20, 2014 Interview with Applicant. The Applicant agreed to narrow this claim limitation to recite "*operating* conditions", instead of just "conditions", in order to gain allowance of the claims. *Id*. In the resulting Notice of Allowance, the examiner stated that the claims of the application leading to the '038 Patent were allowed "in view of the examiner's amendment and for reasons argued by applicant in pages 7-8 of the Remarks, filed on June 06, 2014." '038 Patent File History at Aug. 28, 2014 Notice of Allowance at pp. 4-6.

The shared specification of the '038 Patent, '997 Patent and '918 Patent, however, (i) does not use the term "operating condition", (ii) describes "conditions 104F" that can be monitored by software agents within endpoint system 104, and (iii) includes a list of "illustrative conditions 104F" spanning 5+ columns without specifying which would constitute an "*operating* condition" vs. a "*non-operating* condition." *See, e.g.*, '038 Patent at FIG. 1, 7:64-8:3, 9:25-15:14. Thus, a POSITA would not understand the meaning or scope of the term "operating condition" nor the meaning or scope of the term "one or more software services provided by an operating system on

the endpoint configured to monitor a plurality of operating conditions identified in the plurality of policies." Therefore, claims 17-24 of the '918 Patent, which each incorporate the "operating conditions" term, are invalid as indefinite, lacking enablement, and/or lacking written description in violation of 35 U.S.C. § 112. *See, e.g.*, *Moba*, 325 F.3d at 1319; *Telectronics*, 857 F.2d at 785; *Power-One*, 599 F.3d at 1350.

c. "determining, by the computing system, a compliance state of the endpoint based on the user information and status information"

Asserted claims 1-16 of the '918 Patent recite, among other limitations, "determining, by the computing system, a compliance state of the endpoint based on the user information and status information," but there is inadequate support in the specification for this term. In particular, there is inadequate support in the specification of how a remote computing system makes a determination of a compliance state of the endpoint *based on the user information and status information*.

To satisfy the written description requirement, a patent specification must describe the claimed invention in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the claimed invention. *See, e.g.*, *Moba*, 325 F.3d at 1319; *see also, e.g.*, *Telectronics*, 857 F.2d at 785. A decision on whether a claim is indefinite under 35 U.S.C. 112(b) or pre-AIA 35 U.S.C. 112, second paragraph requires a determination of whether those skilled in the art would understand what is claimed when the claim is read in light of the specification. *See, e.g.*, *Power-One*, 599 F.3d at 1350. Since there is not sufficient detail in the specification to support this limitation for the term "[determining/determine], by the computing system, a compliance state of the endpoint based on the user information and status information", and its meaning cannot be sufficiently ascertained, claims 1-16 of the '918 Patent are invalid as indefinite, lacking enablement, and/or lacking written description in violation of 35 U.S.C. § 112.

d. "determine a compliance state of the endpoint based on the user information and status information"

Asserted claims 17-24 of the '918 Patent recite, among other limitations, "one or more hardware processors at the computing system configured to … determine a compliance state of the endpoint based on the user information and status information," but there is inadequate support in the specification for this term. In particular, there is inadequate support in the specification of how

a remote computing system makes a determination of a compliance state of the endpoint *based on the user information and status information*.

To satisfy the written description requirement, a patent specification must describe the claimed invention in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the claimed invention. *See, e.g.*, *Moba*, 325 F.3d at 1319; *see also, e.g.*, *Telectronics*, 857 F.2d at 785. A decision on whether a claim is indefinite under 35 U.S.C. 112(b) or pre-AIA 35 U.S.C. 112, second paragraph requires a determination of whether those skilled in the art would understand what is claimed when the claim is read in light of the specification. *See, e.g.*, *Power-One*, 599 F.3d at 1350. Since there is not sufficient detail in the specification to support this limitation for the term "determining a compliance state of the endpoint based on the status information" and its meaning cannot be sufficiently ascertained, claims 17-24 of the '918 Patent are invalid as indefinite, lacking enablement, and/or lacking written description in violation of 35 U.S.C. § 112.

e. "configuring one or more applications running on the endpoint on the endpoint to monitor at least a subset of the plurality of operating conditions"

Asserted claims 5, 13 and 21 of the '918 Patent recite, among other limitations, "configuring one or more applications running on the endpoint on the endpoint to monitor at least a subset of the plurality of operating conditions," but the meaning of this term lacks reasonable certainty to a POSITA, especially in view of the specification and the other claim language. In particular, the specification, and the rest of the claim language, do not provide reasonable certainty as to what this term relates to or means, nor how to configure one or more *applications* running *on the endpoint on the endpoint* to monitor at least a subset of the plurality of operating conditions that were identified from the plurality of policies in the data store on the remote computing system.

For example, the antecedent of this term is unclear. In particular, independent claims 1 and 9 recite "configuring one or more software services provided by an operating system on the endpoint to monitor the plurality of operating conditions." And, for example, independent claim 17 similarly recites "one or more software services provided by an operating system on the endpoint configured to evaluate a plurality of operating conditions identified in the plurality of policies." Therefore, the relationship between "one or more software services provided by an operating system on the endpoint" and "one or more applications running on the endpoint" lacks reasonable certainty. The specification does not resolve this ambiguity as, for example, the word

"application" is frequently used in the specification to refer to *monitored* software rather than *monitoring* software. *See, e.g.*, '997 Patent at 11:21-22, 11:60-12:31, 13:9-12, 17:44-18:26, 19:53-20:26, 28:31-35, 29:64-30:59.

Additionally, for example, it is not clear, from the as-issued claims, specification and file history, whether the recitation of "on the endpoint on the endpoint" in the disputed term was intentional. As discussed above, the specification does not resolve the ambiguity in this term. Moreover, the language of issued claims 5, 13 and 21 was identical throughout the entire duration of the prosecution history starting with the Applicant's preliminary amendment concurrently filed with the application leading to the '918 Patent (March 27, 2017).

Given this uncertainty, to the extent that Taasera argues that the Court should rewrite, as a matter of law, claims 5, 13 and 21 to recite "configuring one or more applications running on the endpoint ~~on the endpoint~~ to monitor at least a subset of the plurality of operating conditions," this request should be denied as there is at least a reasonable debate as to whether the second recitation of "on the endpoint" was intentional or unintended. *Novo*, 350 F.at 1354 (a court can correct a claim error "only if (1) the correction is not subject to reasonable debate based on consideration of the claim language and the specification and (2) the prosecution history does not suggest a different interpretation of the claims."). To the extent that Taasera argues that the second recitation of "on the endpoint" was an unintended mistake, the proper forum for correcting this alleged "mistake" is the Patent Office via its certificate of correction procedure, and where the certificate of correction would be effective as of the date the Patent Office issues it. 35 U.S.C. § 255; *Novo,* 350 F.3d at 1356-57. Any request by Taasera for the Court, instead of the Patent Office, to rewrite this claim limitation as a matter of law would be a transparent attempted end-around to this statutory procedure and restriction to change what is not clearly an error. Because the claims, specification, and file history do not resolve the ambiguity in this claim term, claims 6 and 16 are invalid as indefinite in violation of 35 U.S.C. § 112.

Moreover, to the extent Defendant understands Taasera's infringement contentions, Taasera's apparent interpretation of the Asserted Claims of the '918 Patent is that they are broad enough to encompass, e.g., software installed on an endpoint as "software services provided by an operating system on the endpoint"; and information received from software installed on an endpoint as "receiving . . . at the computing system, status information . . . gathered by the one or more software services"; a system that simply receives information from an endpoint as

"determin[e/ing], by the computing system, a compliance state of the endpoint based on the user information and status information, and a plurality of compliance policies in the data store"; and a remote system that can trigger an endpoint to block activity as "authoriz[e/ing] access by the endpoint to a computing resource on the network, authorization being determined by the remote computing system in response to the compliance state". Under these overly broad interpretations of the Asserted Claims, each of the Asserted Claims of the '918 Patent is invalid for lack of written description and/or is not enabled.

7. '948 Patent

Term	Relevant Claim(s)
"correlating, by the event and risk correlation matrix"	All asserted claims
"native computing environment"	All asserted claims
"real-time"	All asserted claims
"the event and behavior correlation engine"	3
"integrity measurement and verification reports"	3
"integrity processor"	All asserted claims

a. "correlating, by the event and risk correlation matrix" (all asserted claims)

All asserted claims of the '948 Patent are invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue experimentation; (2) the specification does not describe either the claim element individually or the claim as a whole in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions,

a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

b. "native computing environment" (all asserted claims)

All asserted claims of the '948 Patent are invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue experimentation; (2) the specification does not describe either the claim element individually or the claim as a whole in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

c. "real-time" (all asserted claims)

All asserted claims of the '948 Patent are invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue experimentation; (2) the specification does not describe either the claim element individually or the claim as a whole in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

d. "event and behavior correlation engine" (claim 3)

Claim 3 of the '948 Patent is invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue

experimentation; (2) the specification does not describe either the claim element individually or the claim as a whole element in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

e. "integrity processor" (all asserted claims)

All asserted claims of the '948 Patent are invalid as indefinite because a person of ordinary skill would be unable to discern the meaning of the term with reasonable certainty.

f. "integrity measurement and verification reports" (claim 3)

Claim 3 of the '948 Patent is invalid as indefinite because a person of ordinary skill would be unable to discern the meaning of the term with reasonable certainty.

8. '616 Patent

Term	Relevant Claim(s)
"a network trust agent, an endpoint trust agent, and a trust orchestration server"	All asserted claims
"sending, by the endpoint trust agent on a monitored device, a dynamic context including endpoint events and actions of the monitored device and applications executing on the monitored device at runtime"	All asserted claims
"sending, by the endpoint trust agent on a monitored device, a dynamic context"	All asserted claims
"correlating, by the trust orchestration server, the received endpoint events and the generated temporal events"	All asserted claims
"integrity measurement and verification reports"	3, 5

a. "a network trust agent, an endpoint trust agent, and a trust orchestration server." (all asserted claims)

All asserted claims of the '616 Patent are invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to

enable a person of ordinary skill in the art to practice the full scope of this claim element without undue experimentation; (2) the specification does not describe either the claim element individually or the claim as a whole in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

b. "sending, by the endpoint trust agent on a monitored device, a dynamic context including endpoint events and actions of the monitored device and applications executing on the monitored device at runtime" and "receiving, at the trust orchestration server, the dynamic context including the endpoint events of the monitored device and the applications executing on the monitored device at runtime" (all asserted claims)

All asserted claims of the '616 Patent are invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue experimentation; (2) the specification does not describe either the claim element individually or the claim as a whole in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

c. "correlating, by the trust orchestration server, the received endpoint events and the generated temporal events" (all asserted claims)

All asserted claims of the '616 Patent are invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue experimentation; (2) the specification does not describe either the claim element individually or the claim as a whole in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at

least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

d. "integrity measurement and verification reports" (claims 3 and 5)

Claims 3 and 5 of the '616 Patent and its dependent claims are invalid for indefiniteness because a person of ordinary skill would be unable to discern the meaning of the "integrity measurement and verification reports" with reasonable certainty.

9. '356 Patent

Term	Relevant Claim(s)
"fourth program instructions, responsive to the packet being a known exploit OR the packet being addressed to a broadcast IP address of a network OR the packet being network administration traffic, to determine that the packet is not a new, exploit candidate; and"	1, 9, 10, 17
"fifth program instructions, responsive to the packet not being a known exploit AND the packet not being addressed to a broadcast IP address of a network AND the packet not being network administration traffic AND the packet not being another type of traffic known to be benign, to determine and report that the packet is a new, exploit candidate; and wherein"	1, 9, 10, 17
"the fourth program instructions are responsive to the packet being a known exploit OR the packet being addressed to a broadcast IP address of a network OR the packet being network administration traffic OR the packet being web crawler traffic, to determine that the packet is not a new, exploit candidate; and"	2, 18
"the fifth program instructions are responsive to the packet not being a known exploit AND the packet not being addressed to a broadcast IP address of a network AND the packet not being network administration traffic AND the packet not being web crawler traffic, to determine that the packet is a new, exploit candidate; and"	2, 18
"the fourth program instructions are responsive to the packet being a known exploit OR the packet being addressed to a broadcast IP address of a network OR the packet being network administration traffic OR the packet having a protocol listed in a list of protocols previously determined to be harmless network broadcast traffic, to determine that the packet is not a new, exploit candidate; and"	9
"the fifth program instructions are responsive to the packet not being a known exploit AND the packet not being addressed to a broadcast IP address of a network AND the packet not being network administration traffic AND the packet not having a protocol listed in a list of protocols previously determined to be harmless network broadcast traffic, to determine and report that the packet is a new, exploit candidate."	9

Term	Relevant Claim(s)
"fifth program instructions, responsive to the packet being a known exploit OR the packet being addressed to a broadcast IP address of a network OR the packet being network administration traffic OR the packet having a protocol listed in a list of protocols previously determined to be harmless broadcast traffic, to determine that the packet is not a new, exploit candidate; and"	13
"sixth program instructions, responsive to the packet not being a known exploit AND the packet not being addressed to a broadcast IP address of a network AND the packet not being network administration traffic AND the packet not having a protocol listed in a list of protocols previously determined to be harmless broadcast traffic, to determine and report that the packet is a new, exploit candidate; and wherein"	13
"the fifth program instructions are responsive to the packet being a known exploit OR the packet being addressed to a broadcast IP address of a network OR the packet being network administration traffic OR the packet being web crawler traffic OR the packet having a protocol listed in a list of protocols previously determined to be harmless broadcast traffic, to determine that the packet is not a new, exploit candidate; and"	14
"the sixth program instructions are responsive to the packet not being a known exploit AND the packet not being addressed to a broadcast IP address of a network AND the packet not being network administration traffic AND the packet not being web crawler traffic AND the packet not being other traffic known to be benign AND the packet not having a protocol listed in a list of protocols previously determined to be harmless broadcast traffic, to determine that the packet is a new, exploit candidate; and"	14
"program instructions to determine if the packet is addressed to a broadcast IP address of a network"	1, 9, 10, 13, 17
"program instructions to determine if the packet is network administration traffic"	1, 9, 10, 13, 17
"the packet not being another type of traffic known to be benign" / "the packet not being other traffic known to be benign"	1, 9, 10, 12, 14, 17

a. "program instructions, responsive to [any of multiple conditions] to determine that the packet is not a new, exploit candidate" / "program instructions, responsive to the packet [not meeting each of multiple conditions] to determine and report that the packet is a new, exploit candidate"

The asserted claims recite, amongst other things, program instructions that determine a packet either "is not a new, exploit candidate" or "is a new, exploit candidate" depending on whether one or multiple criteria are met (the specific criteria of which vary by claim). The full scope of the claims, including these limitations as applied by Taasera in asserting infringement, are not supported by the specification of the '356 Patent. For example, Taasera appears to contend

that a system infringes if it identifies the packet as a new, exploit candidate regardless of the result of the various conditions. But the specification teaches, for example, "[i]n step 100, program 30 determines if the packet contains a portion of a known exploit with a known signature. If so, program 30 stores the packet but "disregards" it, i.e. does not consider it further as a new exploit candidate and will not send it to SOC 44 for further analysis (step 102). As explained in more detail below, program 30 may also disregard the other packets it receives in the same TCP sequence such that the entire message is disregarded (step 102). The reason for disregarding this packet or sequence of packets is because the function of honeypot 12 is to identify new exploits, for which a signature is not yet known." '356 Patent, 5:13-24. Thus, the full scope of the claims as applied by Taasera (i.e., that identifying a packet as a new, exploit candidate regardless of the conditions) is not supported by and actually conflicts with the specification (which teaches that if certain conditions are met, such as being a known exploit, the packet is disregarded and not considered for any further analysis). The claims therefore lack written description support, are not enabled, and are not what the inventors regarded as their invention, in violation of 35 U.S.C. § 112.

b. "program instructions to determine if the packet is addressed to a broadcast IP address of a network"

All asserted claims of the '356 Patent are invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue experimentation; (2) the specification of the '356 Patent also does not describe the claim element in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff's interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

c. "program instructions to determine if the packet is network administration traffic"

All asserted claims of the '356 Patent are invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without

undue experimentation; (2) the specification of the ’356 Patent also does not describe the claim element in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff’s interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

d. “the packet not being another type of traffic known to be benign” / “the packet not being other traffic known to be benign”

The asserted claims of the ’356 Patent reciting “the packet not being another type of traffic known to be benign” or “the packet not being other traffic known to be benign” are invalid as lacking enablement, lacking written description support, and indefinite because (1) the specification lacks sufficient disclosures to enable a person of ordinary skill in the art to practice the full scope of this claim element without undue experimentation; (2) the specification of the ’356 Patent also does not describe the claim element in sufficient detail that one skilled in the art can reasonably conclude that the inventor had possession of the full scope of the claimed invention; and (3) at least under Plaintiff’s interpretation of the term, as evidenced from its Infringement Contentions, a person of ordinary skill in the art would not be able to ascertain the scope and meaning of this term without reasonable certainty.

B. Grounds of Invalidity Under 35 U.S.C. § 101

Section 101 of the Patent Act states: “Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefore, subject to the conditions and requirements of this title.” Section 101 precludes the patenting of mental processes and abstract ideas. *See Mayo Collaborative Servs. v. Prometheus Labs., Inc.*, 556 U.S. 66, 70 (2012) (citation omitted). The Supreme Court has set forth a two-step process for distinguishing patents that claim laws of nature, natural phenomenon, and abstract ideas from those that claim patent-eligible applications of those concepts. *Alice Corp. Pty. Ltd. v. CLS Bank Int’l*, 134 S. Ct. 2347, 2350 (2014). Step one

is to determine whether the claims at issue are directed to a patent-ineligible concept. *Id.* at 2355. Step two is to "examine the elements of the claim to determine whether it contains an 'inventive concept' sufficient to 'transform' the claimed abstract idea into a patent-eligible application." *Id.* at 2357 (citing *Mayo*, 566 U.S. at 72, 78). To answer that question, the elements of each claim are considered both individually and "as an ordered combination" to determine whether the additional elements "transform the nature of the claim" into a patent-eligible application. *Id.* at 2355. The Court has described step two of this analysis as a search for an "'inventive concept'"—i.e., an element or combination of elements that is "sufficient to ensure that the patent in practice amounts to significantly more than a patent upon the [ineligible concept] itself." *Id.* (alteration in original)

Pursuant to the Court's Standing Order Regarding Subject Matter Eligibility Contentions Applicable to All Patent Infringement Cases Assigned to Chief District Judge Rodney Gilstrap, and as described below, Defendant asserts that certain Asserted Claims fails to satisfy the requirements of 35 U.S.C. § 101 due to their failure to claim patentable subject matter. Defendant hereby incorporates by reference its Motion for Partial Judgment on the Pleadings in Case No. 2:22-cv-00314 and 2:22-MD-03042-JRG, including all related briefing and oral argument, showing that the '796, '356, '517, '038, and '918 patent claims are patent-ineligible under 35 U.S.C. § 101. To the extent an Asserted Claim is invalid for any reason under 35 U.S.C. § 101, all claims depending therefrom are invalid unless they add something to the claim that transforms it into patentable subject matter. Accordingly, each of the asserted dependent claims incorporates by reference all § 101 invalidity contentions of the parent claim(s). Defendant reserves the right to rely upon expert testimony in support of their ineligibility contentions.

Additionally, the Asserted Claims may be invalid for statutory and/or obviousness-type double patenting over other patents owned by Plaintiff.

1. ’796 Patent

a. Chart identifying each exception to eligibility to which each challenged claim is directed

Asserted Claims	Representative Claim	Eligibility Exception
1, 2, 7, 10, 12, 13, 18, 21, 23-25	1	Abstract Idea

Asserted ’796 Patent Claim	Identification of Exception to Eligibility
1. A method of automatically processing an input sequence of data symbols, the method comprising the steps of: identifying at least one regularly identifiable expression in the input sequence of data symbols, wherein the at least one regularly identifiable expression represents a pattern that is matchable in accordance with a programming language that supports such a regularly identifiable expression; identifying at least a portion of information associated with the at least one regularly identifiable expression; and extracting the portion of information.	The ’796 patent, which predates *Alice*, claims nothing more than the abstract idea of identifying and extracting information by matching a pattern—tasks that humans routinely perform when reading a document and focusing on key words or phrases. Indeed, the patent describes “[t]he present invention” as “exploiting the readily-identifiable structure of language to explicitly identify portions of data in a document that a user seeks to be identified, e.g., relevant or important information.” Dkt. No. 1-1 at 1:52–55. To do so, the invention uses “regular expressions,” i.e., “regularly identifiable or stereotypical phrases that people commonly use to convey particular information.” *Id*. at 2:12–17. For example, one can identify a caller’s name (“Bob”) from the voice message, “Hi John, it’s Bob.” *Id*. at 2:3–6. The ’796 patent admits that “common programming languages” have “built-in support for regular expressions.” *Id*. at 2:11–15, 2:23–27. Claim 1, which is representative, recites a simple three-step method: (1) identifying a “regularly identifiable expression” that “represents a pattern that is matchable”; (2) identifying a portion of information associated with that expression; and (3) extracting the information. On its face, claim 1 is directed to the abstract idea of identifying and extracting information by matching a pattern. The Federal Circuit has invalidated analogous claims with steps for “1) extracting data from hard copy documents using an automated digitizing unit such as a scanner, 2) recognizing specific information from the extracted data, and 3) storing that information in a memory.” *Content Extraction & Transmission LLC v. Wells Fargo Bank, Nat’l Ass’n*, 776 F.3d 1343, 1345 (Fed. Cir. 2014). Those claims were “drawn to the abstract idea of 1) collecting data, 2) recognizing certain data within the collected data set, and 3) storing that recognized data in a memory.” *Id*. at 1347. The Federal Circuit held that such concepts are “undisputedly well-known,” and “humans have always performed” them. *Id*. For example, banks have long “reviewed checks, recognized relevant data such as the amount,

Asserted ’796 Patent Claim	Identification of Exception to Eligibility
	account number, and identity of account holder, and stored that information in their records.” *Id*. The claims here are no different. Apart from claim 1’s reference to a “programming language that supports such a regularly identifiable expression” (Dkt. No. 1-1 at claim 1)—a “generic computer-implemented” limitation—every other limitation refers to activities that can be performed by a human, mentally or with pen and paper.” *Intellectual Ventures I LLC v. Symantec Corp.*, 838 F.3d 1307, 1318 (Fed. Cir. 2016). “Identifying” and “extracting” is akin to a human recognizing key words like names in a letter and writing them down in a notepad, similar to data recognition steps that have been held abstract. *See id.* at 1313 (“determining . . . whether [input data] matches a characteristic”); *SAP Am., Inc. v. InvestPic, LLC*, 898 F.3d 1161, 1167 (Fed. Cir. 2018) (“selecting certain information” and “analyzing it using mathematical techniques”); *Elec. Power Grp.*, 830 F.3d at 1354 (“gathering and analyzing information of a specified content”). Claim 1’s “essentially result-focused, functional character” confirms it is “ineligible under § 101.” *Elec. Power Grp*., 830 F.3d at 1356. The claim recites “identifying at least one regularly identifiable expression,” “identifying at least a portion of information,” and “extracting the portion of information,” but it does not limit how to achieve these functions, which read on human activity. Claim 1 also fails to recite any “specific improvement to the way computers operate,” *Enfish, LLC v. Microsoft Corp*., 822 F.3d 1327, 1336 (Fed. Cir. 2016), or “overcome a problem specifically arising in the realm of computer networks,” *DDR Holdings, LLC v. Hotels.com, L.P.*, 773 F.3d 1245, 1257 (Fed. Cir. 2014). Although the specification alleges that “regular-expression technology enables extremely fast pattern matching and information extraction with highly optimized standard programs” and “without the expensive and time consuming step of gathering and annotating a large ‘training’ database” (Dkt. No. 1-1 at 2:28–33), this is merely the “improved speed or efficiency inherent with applying the abstract idea on a computer” (*Customedia Techs., LLC v. Dish Network Corp.*, 951 F.3d 1359, 1364–65 (Fed. Cir. 2020) (quotations omitted)).
2. The method of claim 1, wherein the regularly identifiable expression identifying step comprises comparing the input sequence of data symbols to one or more previously-	The same analysis applies to claim 2, which Taasera admits are represented by claim 1. Dkt. 15-2 (Feb. 21, 2023 Hr’g. Tr.) at 29:22–24 (“As counsel pointed out, we also treated claim 1 of each patent as representative. We don’t dispute that claim 1 is representative of the asserted claims in this case.”).

Asserted ’796 Patent Claim	Identification of Exception to Eligibility
stored regularly identifiable expressions to determine if there is a match between a portion of the input sequence and at least one of the previously-stored regularly identifiable expressions.	The other independent claims (12, 23, 24, 25) recite the same steps of identifying expressions, identifying information, and extraction. Dkt. No. 1-1. Two of these claims merely change the form of the invention from a “method” to an “apparatus” with a “memory . . . for storing” information (claim 12) and a “data capture device for obtaining the input sequence of data symbols” (claim 24). These same concepts were held abstract in *Content Extraction*. 776 F.3d at 1347 (“collecting” and “storing . . . data in a memory”). The other two independent claims reframe the same elements as an “article of manufacture” (claim 23) or add “comparing the input document to one or more previously-stored regular expressions” for a “match” (claim 25), which humans can also do. The dependent claims add nothing significant: the same mental step of “comparing” data to find a “match” (claims 2, 13); taking any “specified action” (claims 7, 18); and limiting data to categories including text (claims 10, 21). These elements do not make the claims less abstract and confirm that claim 1 is representative. *SAP*, 898 F.3d at 1167 (“collecting information, including when limited to particular content,” is abstract).
7. The method of claim 1, wherein the extracted portion of information is used to take a specified action.	
10. The method of claim 1, wherein the input sequence of data symbols is representative of at least one of text data, transcribed spoken data, deoxyribonucleic acid sequence data, ribonucleic acid sequence data, amino-acid sequence data, audio sequence data, and video sequence data.	
12. Apparatus for automatically processing an input sequence of data symbols, the apparatus comprising: at least one processor operative to: (i) identify at least one regularly identifiable expression in the input sequence of data symbols, wherein the at least one regularly identifiable expression represents a pattern that is matchable in accordance with a programming language that supports such a regularly identifiable expression; (ii) identify at least a portion of information associated with the at least one regularly identifiable expression; and	

Asserted ’796 Patent Claim	Identification of Exception to Eligibility
(iii) extract the portion of information; and memory, operatively coupled to the at least one processor, for storing at least a portion of results associated with the identifying and extracting operations.	
13. The apparatus of claim 12, wherein the regularly identifiable expression identifying operation comprises comparing the input sequence of data symbols to one or more regularly identifiable expressions, previously stored in the memory, to determine if there is a match between a portion of the input sequence and at least one of the previously-stored regularly identifiable expressions.	
18. The apparatus of claim 12, wherein the extracted portion of information is used to take a specified action.	
21. The apparatus of claim 12, wherein the input sequence of data symbols is representative of at least one of text data, transcribed spoken data, deoxyribonucleic acid sequence data, ribonucleic acid sequence data, amino-acid sequence data, audio sequence data, and video sequence data.	
23. An article of manufacture for use in automatically processing an	

Asserted ’796 Patent Claim	Identification of Exception to Eligibility
input sequence of data symbols, comprising a machine readable medium containing one or more programs which when executed implement the steps of: identifying at least one regularly identifiable expression in the input sequence of data symbols, wherein the at least one regularly identifiable expression represents a pattern that is matchable in accordance with a programming language that supports such a regularly identifiable expression; identifying at least a portion of information associated with the at least one regularly identifiable expression; and extracting the portion of information.	
24. Apparatus for automatically processing an input sequence of data symbols, the apparatus comprising: a data capture device for obtaining the input sequence of data symbols; at least one processor, operatively coupled to the data capture device, operative to: (i) identify at least one regularly identifiable expression in the input sequence of data symbols. wherein the at least one regularly identifiable expression represents a pattern that is matchable in	

Asserted ’796 Patent Claim	Identification of Exception to Eligibility
accordance with a programming language that supports such a regularly identifiable expression; (ii) identify at least a portion of information associated with the at least one regular expression; and (iii) extract the portion of information; memory, operatively coupled to the at least one processor, for storing at least a portion of results associated with the identifying and extracting operations; and a data output device, operatively coupled to the at least one processor, for presenting the extracted portion of information to a user.	
25. A method of automatically processing an input document, the method comprising the steps of: identifying one or more regular expressions in the input document by comparing the input document to one or more previously-stored regular expressions to determine if there is a match between a portion of the input document and at least one of the previously-stored regular expressions, wherein a regular expression represents a pattern that is matchable in accordance with a programming language that supports such a regular expression; and identifying at least a portion of information associated	

Asserted ’796 Patent Claim	Identification of Exception to Eligibility
with the at least one regular expression for extraction.	

b. Chart identifying a description of the industry at the relevant time and how each element of each challenged claim, individually and in combination with the other elements, was well understood, routine, and conventional

Claim	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
1. A method of automatically processing an input sequence of data symbols, the method comprising the steps of: identifying at least one regularly identifiable expression in the input sequence of data symbols, wherein the at least one regularly identifiable expression represents a pattern that is matchable in accordance with a programming language that supports such a regularly identifiable expression; identifying at least a portion of information associated with the at least one regularly identifiable expression; and extracting the portion of information.	The ’796 claims also fail Alice step two because “[t]hey add nothing outside the abstract realm.” *SAP*, 898 F.3d at 1169. At most, some claims recite a “programming language,” “processor,” “memory,” “data capture device,” or “data output device” (claims 12, 24) but “require no improved computer resources [that the patentee] claims to have invented.” *Id.* at 1169–70. Indeed, the ’796 patent admits that “‘processor’ . . . include[s] any processing device” (Dkt. No. 1-1 at 12:30–31); “‘memory’ . . . include[s] any memory devices associated with a processor” (id. at 12:35–37); “data capture device” can be a “digital scanner” or other “alternate forms” (id. at 11:49–65); and “data output device,” whose “structure and operations depend on the application,” can be any “display” (id. at 12:52–65). Again, “common programming languages” in the prior art had “built-in support for regular expressions” (id. at 2:25–27; see also id. at 3:22–31), and the claims “may be implemented in various forms of hardware, software, or combinations thereof,” including “general purpose digital computers” (id. at 12:66–13:5). Thus, all asserted claims fail both Alice steps. *Content Extraction*, 776 F.3d at 1348 (“recognizing and storing information from hard copy documents using a scanner and a computer” is not inventive). Defendant further incorporates by reference the above-identified Applicant Admitted Prior Art, wherein the ’796 patent itself admits that the elements of the asserted claims were well-understood, routine, and conventional. Further, while prior art references are unnecessary to conclude that the asserted claims fail *Alice* step two, Defendant incorporates by reference each of its invalidity charts for the ’796 patent (Exhibit A), which identify prior art teachings that further confirm that the elements of the asserted claims, individually or when combined, were well-understood, routine, and conventional in the industry.
2. The method of claim 1, wherein the regularly	*See* claim 1, above.

Claim	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
identifiable expression identifying step comprises comparing the input sequence of data symbols to one or more previously-stored regularly identifiable expressions to determine if there is a match between a portion of the input sequence and at least one of the previously-stored regularly identifiable expressions.	
7. The method of claim 1, wherein the extracted portion of information is used to take a specified action.	
10. The method of claim 1, wherein the input sequence of data symbols is representative of at least one of text data, transcribed spoken data, deoxyribonucleic acid sequence data, ribonucleic acid sequence data, amino-acid sequence data, audio sequence data, and video sequence data.	
12. Apparatus for automatically processing an input sequence of data symbols, the apparatus comprising: at least one processor operative to: (i) identify at least one regularly identifiable expression in the input sequence of data symbols, wherein the at least one regularly identifiable expression represents a pattern that is matchable in accordance with a programming language that supports such a regularly	

Claim	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
identifiable expression; (ii) identify at least a portion of information associated with the at least one regularly identifiable expression; and (iii) extract the portion of information; and memory, operatively coupled to the at least one processor, for storing at least a portion of results associated with the identifying and extracting operations.	
13. The apparatus of claim 12, wherein the regularly identifiable expression identifying operation comprises comparing the input sequence of data symbols to one or more regularly identifiable expressions, previously stored in the memory, to determine if there is a match between a portion of the input sequence and at least one of the previously-stored regularly identifiable expressions.	
18. The apparatus of claim 12, wherein the extracted portion of information is used to take a specified action.	
21. The apparatus of claim 12, wherein the input sequence of data symbols is representative of at least one of text data, transcribed spoken data, deoxyribonucleic acid sequence data, ribonucleic acid sequence data, amino-acid sequence data, audio sequence data, and video sequence data.	

Claim	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
23. An article of manufacture for use in automatically processing an input sequence of data symbols, comprising a machine readable medium containing one or more programs which when executed implement the steps of: identifying at least one regularly identifiable expression in the input sequence of data symbols, wherein the at least one regularly identifiable expression represents a pattern that is matchable in accordance with a programming language that supports such a regularly identifiable expression; identifying at least a portion of information associated with the at least one regularly identifiable expression; and extracting the portion of information.	
24. Apparatus for automatically processing an input sequence of data symbols, the apparatus comprising: a data capture device for obtaining the input sequence of data symbols; at least one processor, operatively coupled to the data capture device, operative to: (i) identify at least one regularly identifiable expression in the input sequence of data symbols. wherein the at least one regularly identifiable	

Claim	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
expression represents a pattern that is matchable in accordance with a programming language that supports such a regularly identifiable expression; (ii) identify at least a portion of information associated with the at least one regular expression; and (iii) extract the portion of information; memory, operatively coupled to the at least one processor, for storing at least a portion of results associated with the identifying and extracting operations; and a data output device, operatively coupled to the at least one processor, for presenting the extracted portion of information to a user.	
25. A method of automatically processing an input document, the method comprising the steps of: identifying one or more regular expressions in the input document by comparing the input document to one or more previously-stored regular expressions to determine if there is a match between a portion of the input document and at least one of the previously-stored regular expressions, wherein a regular expression represents a pattern that is matchable in accordance with a programming language that supports such a regular expression; and	

Claim	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
identifying at least a portion of information associated with the at least one regular expression for extraction.	

Defendant incorporates its Motion for Partial Judgment on the Pleadings in Case No. 2:22-cv-00314 and 2:22-MD-03042-JRG, including all related briefing and oral argument, showing the factual and legal basis that support that the challenged '796 patent claims are patent-ineligible under 35 U.S.C. § 101. Defendant further incorporates by reference the above-identified Applicant Admitted Prior Art, wherein the '796 patent itself admits that certain technologies related to the elements of the asserted claims were well-understood, routine, and conventional. Defendant also incorporates each of its invalidity charts for the '796 patent (Exhibit A), which may additionally identify, among others, how the elements of the asserted claims, individually or when combined, were well-understood, routine, and conventional in the industry at the relevant time frame.[33]

2. '137 Patent

All asserted claims of the '137 patent are directed to patent ineligible abstract ideas.

Asserted Claims	Representative Claim	Eligibility Exception
2-4, 6-14, 16-18, 22-25, 29	1	Abstract Idea
7-14, 16-18, 22-25, 29	6	

Below is a chart with a description of the industry, at the relevant time, in which the asserted claims are alleged to be well understood, routine, and conventional, and the factual and

[33] Defendants refer to its exemplary identification of motivations to combine and its invalidity charts solely for the purpose of complying with the Court's standing order. Defendant's reference to its exemplary identification of motivation to combine or its invalidity charts is not intended to be, and shall not be construed as, an admission that an analysis under 35 U.S.C. 102 (anticipation) or 103 (obviousness) is relevant or necessary to determine the patent eligibility of the asserted claims.

legal basis therefor; a description of how each element of each asserted claim, both individually and in combination with the other elements of that claim, was: (i) well understood; (ii) routine; and (iii) conventional, in the relevant industry at the relevant time, and the legal and factual basis therefor, and any other factual or legal basis for how the asserted claims are otherwise ineligible for patent protection. Defendant reserves the right to rely on expert testimony in support of their ineligibility defenses.

’137 Patent Claim	**Ineligibility Contentions**
1. A system for managing security of a computing device comprising: a pre-execution module operable for receiving notice from the computing device's operating system that a new program is being loaded onto the computing device; a validation module coupled to the pre-execution monitor operable for determining whether the program is valid; a detection module coupled to the pre-execution monitor operable for intercepting a trigger from the computing device's operating system; and an execution module coupled to the detection module and operable for monitoring, at the operating system kernel of the computing device, the program in response to the trigger intercepted by the detection module.	Like all asserted claims of the ’137 patent, claim 1 is directed to the abstract idea of trying to validate an actor’s identity before an activity is carried out, and if it cannot be validated, then monitoring the actor for suspicious activity as the activity is performed. This abstract idea had long been implemented by human beings in ordinary life. For example, it was ordinary human behavior to halt a visitor at a security gate, verify the visitor’s security clearance, and if the security clearance was unsatisfactory, to monitor the visitor. As a particular example, a prison visitor might be halted at security to present identification. If the visitor could not be identified as a prison guard, the visitor would be monitored closely (e.g., pat downs, visual monitoring, metal detectors, inspecting baggage, an escort, security cameras, interventions as needed), but a prison guard whose identity was verified would not be so monitored. As another particular example, an officer visitor might be halted at security to present identification. If the visitor could not be identified as an employee, the visitor would be monitored closely, but an employee whose identity was verified would not be so monitored. In the case of the ’137 patent claims, the identity to be validated is that of a software program executing on conventional computer components, and the activity of the software program is monitored at the operating system kernel if its identity is not validated, again with conventional computer components. The claims are thus directed to an abstract idea. The claim does not impose limitations that provide an inventive concept. The preamble is a non-limiting, generic, and abstract recitation of an intended use: “managing security of a computing device” (*e.g.*, security guard manages

’137 Patent Claim	Ineligibility Contentions
	security of a building). The body of the claim recites generic, functional modules, including: • A pre-execution module that receives notice from the computing system’s operating system that a new program is loaded (*e.g.*, security guard sees person approaching checkpoint); • A validation module that determines whether the new program is valid (*e.g.*, security guard checks identification); • A detection module that intercepts triggers from the computing device’s operating system and an execution module that performs “monitoring” at the operating system kernel in response to the trigger (*e.g.*, security guard is alerted and monitors suspicious actor whose identity has not been appropriately validated, takes action if needed). *See Elec. Power Grp., LLC v. Alstom SA,* 830 F.3d 1350, 1356 (Fed. Cir. 2016) (“the essentially result-focused, functional character of claim language has been a frequent feature of claims held ineligible under § 101”). The claimed elements were well known, routine, and conventional at the time of the alleged invention in the industry of security (or computer security), whether considered individually or as a combination. This is confirmed by the claim charts submitted with Defendant’s invalidity contentions for the ’137 patent, which show various instances of each element of each claim disclosed in the prior art individually and as a combination, as well as the obviousness of the claims. As to the pre-execution module, the ability to determine when a new program is loaded from an operating system is directed to an abstract idea and was well-known, routine, and conventional, generic computer activity. *SAP Am., Inc. v. INVESTPIC, LLC*, 898 F.3d 1161, 1168 (Fed. Cir. 2018) (“collecting information, including when limited to particular content (which does not change its character as information), is within the realm of abstract ideas”). Loading programs to an operating system was well-known, routine, and conventional. Other claims, which recite interrupting the program from loading, are again reciting a well-known,

’137 Patent Claim	Ineligibility Contentions
	routine and conventional activity of interrupting a software routine, and do no more than implement the abstract idea of halting an actor at a security checkpoint on conventional computer components. As to the claimed “validation module,” it determines “whether the program is valid,” which again is directed to the abstract idea with no limitation that provides an inventive concept. The specification admits that a “validation module” was well-known, routine, and conventional. According to the specification, validation may be accomplished by “any one of a variety of pieces of software that are used to verify that a program has not been tampered with. For example, the validation module can represent the MD5 software module commonly known to those in the art. The MD5 software module calculates a checksum for each allowed program and that checksum is stored for later comparison.” ’137 patent at 8:7-19. The alert and validation process adds nothing patent eligible to the abstract idea, as determining whether the identity of a program is valid using computer processes is admitted by the specification to be conventional. Moreover, a human being is capable of receiving an alert from an operating system that a new program is being loaded, calculating a hash such as an MD5 hash—which is a mathematical algorithm that can be done in the mind or on pen and paper that produces a specified output based on a specified input—and comparing the result to a list of valid hashes. As affirmed by the ’137 patent specification, the act of validating a software program by a hash value was conventional activity and does not add patentable significance to the abstract idea. *See also SAP Am., Inc. v. INVESTPIC, LLC*, 898 F.3d 1161, 1168 (Fed. Cir. 2018) (“collecting information, including when limited to particular content (which does not change its character as information), is within the realm of abstract ideas,” and so too “is analyzing information . . . by mathematical algorithms”). Claim 1 also refers to a “trigger from the computer device’s operating system” intercepted by a “detection module,” and in response, “an execution module” performs “monitoring” of the program at the operating system kernel. This states the abstract idea of monitoring the unverified actor for suspicious activity, couched in functional language and conventional computer componentry. *SAP Am., Inc. v. INVESTPIC, LLC*, 898 F.3d 1161, 1168 (Fed. Cir. 2018) (“collecting information, including when limited to particular content

’137 Patent Claim	Ineligibility Contentions
	(which does not change its character as information), is within the realm of abstract ideas”). The claimed “detection module” that intercepts triggers from the computing device’s operating system and the claimed “execution module” that performs “monitoring” at the operating system kernel level in response to the intercepted trigger “do not go beyond requiring the collection [and] analysis . . . of available information in a particular field, stating those functions in general terms, without limiting them to technical means for performing the functions that are arguably an advance over conventional computer and network technology.” *Elec. Power Group, LLC v. ALSTOM SA*, 830 F.3d 1350, 1351 (Fed. Cir. 2016). “[M]erely selecting information, by content or source, for collection, analysis, and display does nothing significant to differentiate a process from ordinary mental processes, whose implicit exclusion from § 101 undergirds the information-based category of abstract ideas.” *Id.* at 1355. The claims “merely call for performance of the claimed information collection [and] analysis . . . functions ‘on a set of generic computer components.’” *Id.*; *see also SAP Am., Inc. v. INVESTPIC, LLC*, 898 F.3d 1161, 1168 (Fed. Cir. 2018) (“collecting information, including when limited to particular content (which does not change its character as information), is within the realm of abstract ideas,” and so too “is analyzing information . . . by mathematical algorithms”); *see also Internet Patents Corp. v. Active Network, Inc.*, 790 F.3d 1343, 1349 (Fed. Cir. 2015) (limitations “represent merely generic data collection steps or siting the ineligible concept in a particular technological environment”). Moreover, a human being is capable of receiving triggers from an operating system and, in response, monitoring a program at the operating system kernel. The specification also acknowledges that “kernel space 107 refers to the central part of the operating system 180” and “typically represents that portion of the operating system 180 that directly accesses the hardware of the computing device 103,” whereas “user space 105 represents portions of the computing device 103 that interact with software and data received from outside the computing device 103.” 137 patent at 5:61-6:1. This further evidences that monitoring at the operating system kernel is conventional use of conventional components, and that it constitutes performing a generic, conventional analysis function on generic computer

’137 Patent Claim	Ineligibility Contentions
	components. Moreover, “merely selecting information, by content or source, for collection, analysis, and display does nothing significant to differentiate a process from ordinary mental processes, whose implicit exclusion from § 101 undergirds the information-based category of abstract ideas.” *Elec. Power Group, LLC v. ALSTOM SA*, 830 F.3d 1350, 1355 (Fed. Cir. 2016). The specification confirms that the claims are implemented on conventional computer components that add nothing to provide an inventive concept. The specification acknowledges that its description “is represented largely in terms of processes and symbolic representations of operations in a distributed computing environment by conventional computer components, such as database servers, application servers, routers, security devices, firewalls, clients, workstations, memory storage devices, display devices and input devices. Each of these conventional distributed computing components is accessible via a communications network, such as a wide area network or local area network.” ’137 patent at 5:12-20. The specification admits that the disclosure is provided in terms of abstract, functional language left up to the designer to implement. “The present invention also includes computer programs that embody the functions described herein and illustrated in the appended flow charts. 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. Further, a skilled programmer would be able to write such a computer program to implement the disclosed invention based on the flow charts and associated description in the application text, for example.” ’137 patent at 5:32-41. Even if Taasera could demonstrate that the claims do not completely preempt the abstract idea, a “narrow claim directed to an abstract idea, however, is not necessarily patent-eligible, for while preemption may signal patent ineligible subject matter, the absence of complete preemption does not demonstrate patent eligibility.” *Intellectual Ventures I LLC v. Symantec Corp.*, 838 F.3d 1307, 1320 (Fed. Cir. 2016) (cleaned up).

’137 Patent Claim	Ineligibility Contentions
	Even if Taasera could show that the asserted claims reflect novel, non-obvious, or even groundbreaking techniques (and it cannot), the claims would still be patent ineligible. A court “may assume that the techniques claimed are ‘groundbreaking, innovative, or even brilliant,’ but that is not enough for eligibility.” *Elec. Commc’n Techs., LLC v. ShoppersChoice.com, LLC*, 958 F.3d 1178, 1182-83 (Fed. Cir. 2020); *see also Synopsys, Inc. v. Mentor Graphics Corp.*, 839 F.3d 1138, 1151 (Fed. Cir. 2016) (“a claim for a new abstract idea is still an abstract idea”). Nor can Taasera argue patent eligibility based on components that “merely provide a generic environment in which to carry out the abstract idea.” *Ericsson v. TCL Commc’n Tech. Holdings*, 955 F.3d 1317, 1327 (Fed. Cir. 2020)
2. The system of claim 1, wherein the pre-execution module is further operable for suspending loading of the program onto the computing device.	Claim 2 adds nothing to take claim 1 from an abstract idea to an inventive concept. It recites generic computer activity of suspending the loading of a program. Furthermore, these activities were well known, routine, and conventional as shown by the accompanying claim charts. As reflected by the analysis of claim 1, it was ordinary human activity to halt an actor at a security gate so that a security process could be performed, and claim 2’s functional recitation of performing that activity on a computer for a computer program does not introduce patent eligible subject matter. *See also Ericsson v. TCL Commc’n Tech. Holdings*, 955 F.3d 1317, 1327 (Fed. Cir. 2020) (“Controlling access to resources is exactly the sort of process that can be performed in the human mind, or by a human using pen and paper, which we have repeatedly found unpatentable”)
3. The system of claim 1, wherein the pre-execution module is further operable for retrieving validation data.	Claim 3 adds nothing to take claim 1 from an abstract idea to an inventive concept. It recites generic computer activity of retrieving validation data. Furthermore, these activities were well known, routine, and conventional as shown by the accompanying claim charts. Claim 3 recites that the pre-execution module is further operable for retrieving validation data. As discussed for

’137 Patent Claim	Ineligibility Contentions
	claim 1, the specification states the validation module can use conventional hashing techniques such as an MD5 hash that generates a checksum for each allowed program, and the retrieval of validation data introduced in claim 3 thus does not transform non-eligible claim 1 into a patent eligible invention. *SAP Am., Inc. v. INVESTPIC, LLC*, 898 F.3d 1161, 1168 (Fed. Cir. 2018) (“collecting information, including when limited to particular content (which does not change its character as information), is within the realm of abstract ideas”)
4. The system of claim 1, wherein the execution module is further operable for deciding whether to terminate the trigger intercepted by the detection module.	Claim 4 adds nothing to take claim 1 from an abstract idea to an inventive concept. It recites generic computer activity of deciding whether to terminate a trigger. Furthermore, these activities were well known, routine, and conventional as shown by the accompanying claim charts. Claim 4 recites that the execution module is further operable for deciding whether to terminate the trigger intercepted by the detection module. This reflects an abstract idea, stopping suspicious activity if it is detected, implemented here on conventional computer components. *See also Ericsson v. TCL Commc’n Tech. Holdings*, 955 F.3d 1317, 1327 (Fed. Cir. 2020) (“Controlling access to resources is exactly the sort of process that can be performed in the human mind, or by a human using pen and paper, which we have repeatedly found unpatentable”).
6. A computer implemented method for implementing security for a computing device comprising the steps of: interrupting the loading of a new program for operation with the computing device; validating the new program; if the new program is validated, permitting the new program to continue loading	*See* claim 1. In addition, claim 6 recites functional steps, and thus any patent eligibility that is incorrectly alleged to be present due to the “module” limitations of claim 1 is not present in claim 6.

’137 Patent Claim	Ineligibility Contentions
and to execute in connection with the computing device; if the new program is not validated, monitoring the new program while it loads and executes in connection with the computing device, wherein the step of monitoring the new program while it executes is performed at the operating system kernel of the computing device.	
7. The method of claim 6, wherein the step of interrupting the loading of a new program comprises: intercepting a signal from the computing device's operating system that the new program is loading, and suspending the loading of the new program.	Claim 7 adds nothing to take claim 6 from an abstract idea to an inventive concept. It recites generic computer activity of intercepting a signal from the operating system that the new program is loading and suspending the loading of the new program. Furthermore, these activities were well known, routine, and conventional as shown by the accompanying claim charts. Claim 7 does “not go beyond requiring the collection [and] analysis . . . of available information in a particular field, stating those functions in general terms, without limiting them to technical means for performing the functions that are arguably an advance over conventional computer and network technology.” *Electric Power Group, LLC v. ALSTOM SA*, 830 F.3d 1350, 1351 (Fed. Cir. 2016). “[M]erely selecting information, by content or source, for collection, analysis, and display does nothing significant to differentiate a process from ordinary mental processes, whose implicit exclusion from § 101 undergirds the information-based category of abstract ideas.” *Id.* at 1355.
8. The method of claim 6, wherein the step of monitoring the new program comprises intercepting a signal from the computing device's operating system.	Claim 8 adds nothing to take claim 6 from an abstract idea to an inventive concept. It recites generic computer activity of intercepting a signal from the computing device’s operating system. Furthermore, these activities were well known, routine, and conventional as shown by the accompanying claim charts.

’137 Patent Claim	Ineligibility Contentions
	Claim 7 does “not go beyond requiring the collection [and] analysis . . . of available information in a particular field, stating those functions in general terms, without limiting them to technical means for performing the functions that are arguably an advance over conventional computer and network technology.” *Electric Power Group, LLC v. ALSTOM SA*, 830 F.3d 1350, 1351 (Fed. Cir. 2016). “[M]erely selecting information, by content or source, for collection, analysis, and display does nothing significant to differentiate a process from ordinary mental processes, whose implicit exclusion from § 101 undergirds the information-based category of abstract ideas.” *Id.* at 1355.
9. The method of claim 6, wherein the step of validating the new program comprises determining whether the new program corresponds with an approved program.	Claim 9 adds nothing to take the validation referred to in claim 6 from an abstract idea to an inventive concept. The specification affirms that validation may be implemented with well known, routine, and conventional computer software such as comparison of MD5 hashing. *See* claim 1.
10. The method of claim 6, wherein the step of validating the new program comprises comparing a checksum for the new program with a previously determined checksum.	An MD5 hash is a checksum. Thus, claim 10 adds nothing to take the validation referred to in claim 6 from an abstract idea to an inventive concept. The specification affirms that validation may be implemented with well known, routine, and conventional computer software such as comparison of MD5 hashing. *See* claim 1.
11. The method of claim 6, wherein the step of validating the new program comprises analyzing characteristics of the new program.	Claim 11 adds nothing to take the validation of claim 6 from an abstract idea to an inventive concept. A hash such as an MD5 hash is a checksum generated by analyzing characteristics of the new program, such as its binary. That is, an MD5 hash is generated by analyzing the program and generating a unique value that identifies the program. The specification affirms that validation may be implemented with well known, routine, and conventional computer software such as comparison of MD5 hashing. *See* claim 1. To the extent the term characteristics is alleged to concern something other than hashes, claim 11 likewise adds nothing to take the claim from an abstract idea to an inventive concept and recites only well known, routine, and conventional activity. For example, it was well known in the field to identify files (trusted and otherwise) based on their

’137 Patent Claim	Ineligibility Contentions
	file extensions, file names, file paths, and the like, as further demonstrated by the accompanying claim charts. Claim 11 does “not go beyond requiring the collection [and] analysis . . . of available information in a particular field, stating those functions in general terms, without limiting them to technical means for performing the functions that are arguably an advance over conventional computer and network technology.” *Electric Power Group, LLC v. ALSTOM SA*, 830 F.3d 1350, 1351 (Fed. Cir. 2016). “[M]erely selecting information, by content or source, for collection, analysis, and display does nothing significant to differentiate a process from ordinary mental processes, whose implicit exclusion from § 101 undergirds the information-based category of abstract ideas.” *Id.* at 1355.
12. The method of claim 6, wherein the step of monitoring the new program comprises controlling the files the new program attempts to access during execution of the new program.	Claim 12 adds nothing to take the monitoring of claim 6 from an abstract idea to an inventive concept. The idea of controlling the files a program attempts to access during execution is abstract and well known, routine, and conventional. *See also Ericsson v. TCL Commc’n Tech. Holdings*, 955 F.3d 1317, 1327 (Fed. Cir. 2020) (“Controlling access to resources is exactly the sort of process that can be performed in the human mind, or by a human using pen and paper, which we have repeatedly found unpatentable”). Human beings have long performed the abstract idea of controlling what information unauthorized users can access. *See also* accompanying claim charts.
13. A computer-readable medium having computer-executable instructions for performing the steps recited in claim 6.	Claim 13 recites a computer-readable medium having computer-executable instructions for performing the steps recited in claim 6. Because a claim that commands “do it on a computer” does not transform an otherwise patent-ineligible claim into patent eligible subject matter, claim 13 does not add any patent eligible matter to patent ineligible claim 6. *Alice Corp. v. CLS Bank Int’l*, 134 S.Ct. 2347, 2358 (2014).
14. A computer-implemented method for implementing security for a computing device, comprising the steps of:	Claim 14 is patent ineligible for substantially the same reasons as claims 1 and 6.

’137 Patent Claim	Ineligibility Contentions
identifying an allowed program that is permitted to execute on the computing device; receiving a signal that a new program is going to be executed on the computing device; suspending the execution of the new program on the computing device; determining whether the new program is the same as the allowed program; if the new program is the same as the allowed program, permitting the new program to execute on the computing device; and if the new program is not the same as the allowed program, monitoring the new program while allowing it to execute on the computing device, wherein the step of monitoring the new program while allowing it to execute is performed at the operating system kernel of the computing device.	
16. The method of claim 14, wherein the step of determining whether the new program is the same as the allowed program comprises: computing a checksum for the new program, and comparing the new program's checksum to a checksum for the allowed program.	Claim 16 is patent ineligible for substantially the same reasons as claim 10.
17. The method of claim 14, wherein the step of determining whether the new	Claim 17 is patent ineligible for substantially the same reasons as claim 10. As the specification states, the commonly known “MD5 software module calculates a

’137 Patent Claim	Ineligibility Contentions
program is the same as the allowed program comprises determining whether the new program has been modified.	checksum for each allowed program and that checksum is stored for later comparison.” ’137 patent at 8:13-18. An MD5 hash calculated for a program is used to determine if the program has been altered by comparing a hash for the program at issue to the authorized program. The claimed step of “determining whether the new program has been modified” thus reflects no more than the well-known, routine, and conventional use of hash comparisons that does not add any patentable significance to the abstract idea.
18. The method of claim 14, wherein the step of monitoring the new program while allowing it to execute comprises controlling the files the program attempts to access during execution of the new program on the computing device.	Claim 18 is patent ineligible for substantially the same reasons as claim 12. The idea controlling the files a program attempts to access during execution is abstract and well known, routine, and conventional. *See also Ericsson v. TCL Commc’n Tech. Holdings*, 955 F.3d 1317, 1327 (Fed. Cir. 2020) (“Controlling access to resources is exactly the sort of process that can be performed in the human mind, or by a human using pen and paper, which we have repeatedly found unpatentable”). Human beings have long performed the abstract idea of controlling what information unauthorized users can access. *See also* accompanying claim charts.
22. The method of claim 14, further comprising the step of terminating the execution of the new program if the program performs suspicious activities.	Claim 22 is patent ineligible for substantially the same reasons as claim 4. Terminating the execution of a new program if it performs suspicious activities recites generic computer functionality that adds no inventive concept to the abstract idea of stopping suspicious activities when they are observed (e.g., a security guard stops an actor who is performing suspicious activities). *See also Ericsson v. TCL Commc’n Tech. Holdings*, 955 F.3d 1317, 1327 (Fed. Cir. 2020) (“Controlling access to resources is exactly the sort of process that can be performed in the human mind, or by a human using pen and paper, which we have repeatedly found unpatentable”).
24. A computer-readable medium having computer-executable instructions for performing the steps recited in claim 14.	Claim 24 recites a computer-readable medium having computer-executable instructions for performing the steps recited in claim 14. Because a claim that commands “do it on a computer” does not transform an otherwise patent-ineligible invention into patent eligible subject matter, claim 24 does not add any patent eligible matter to claim 14.

’137 Patent Claim	Ineligibility Contentions
25. A computer-implemented method for performing security for a computer device during a pre-execution phrase comprising the steps of: identifying an allowed program that is permitted to execute with the computing device; receiving a signal that a new program is being loaded for execution with the computing device; suspending the loading of the new program; comparing the new program to the allowed program; and determining whether the new program is valid; if the new program is valid, permitting the new program to execute on the computing device; and if the new program is not valid, monitoring the new program while allowing it to execute on the computing device, wherein the step of monitoring the new program while allowing it to execute is performed at the operating	Claim 25 is patent ineligible for substantially the same reasons as claims 1 and 6.

’137 Patent Claim	Ineligibility Contentions
system kernel of the computing device.	
29. A computer-readable medium having computer-executable instructions for performing the steps recited in claim 25.	Claim 29 recites a computer-readable medium having computer-executable instructions for performing the steps recited in claim 25. Because a claim that commands “do it on a computer” does not transform an otherwise patent-ineligible claim into patent eligible subject matter, claim 29 does not add any patent eligible matter to patent ineligible claim 25.

3. ’441 Patent

Although couched in technical language, the Asserted Claims of the ’441 Patent are simply directed to gathering, analyzing, and presenting security information for a running application. Claim 1 of the ’441 Patent recites “[a] method of providing an attestation service for an application at runtime,” comprising (1) receiving an execution context indicating attributes of the application and a security context providing security information about the application, (2) generating a report indicating security risks associated with the application, and (3) sending an attestation result associated with the application. See ’441 Patent at 26:27–45. Claim 1 is representative of all Asserted Claims of the ’441 Patent, primarily because claims 2–12 depend from claim 1 and do not add patent-eligible subject matter.

Starting with *Alice* step one, the ’441 Patent recites collecting information, analyzing such information by mathematical algorithms, and presenting the results of such analysis. Similar methods have been held abstract on several occasions. In *Elec. Power Grp.*, the Federal Circuit noted that it has “treated collecting information, including when limited to particular content . . . as within the realm of abstract ideas.” 830 F.3d at 1353–54. Additionally, the Federal Circuit treats “analyzing information by steps people go through in their minds, or by mathematical algorithms, without more, as essentially mental processes within the abstract-idea category.” *Id.* at 1354. Further, “merely presenting the results of abstract processes of collecting and analyzing information, without more . . . is abstract as an ancillary part of such collection and analysis.” *Id.*;

see also Cellular Commc'ns Equip. LLC v. AT&T Inc., No. 2:15-CV-00576-RWS-RSP, 2017 WL 2984074, at *3 (E.D. Tex. June 27, 2017) ("Here, claim 1 requires nothing more than conventional computer equipment . . . to collect information . . . and present that information in the form of the power headroom report to the network.").

In *Uniloc USA, Inc. v. ADP, LLC*, a claimed "license management method that indicate[d] a user's authorization to access an application," was abstract because the patent recited "the use of a computer as a tool to process information." 772 F. App'x 890, 901 (Fed. Cir. 2019). Similarly, in *Bozeman Fin. LLC v. Fed. Rsrv. Bank of Atlanta*, the Federal Circuit held that a check fraud detection "method of receiving data from two financial records, storing that data, comparing that data, and displaying the results" was invalid under § 101. 955 F.3d 971, 978 (Fed. Cir. 2020).

Turning to *Alice* step two, the '441 Patent recites concepts that are well-understood, routine, and conventional. Said another way, the '441 Patent fails to recite additional elements that transform the nature of the claims into patent-eligible subject matter. As shown by the prior art cited in Defendant's Invalidity Contentions, the claims of the '441 Patent recite concepts that were well known in the art.

4. '517 Patent

a. Chart identifying each exception to eligibility to which each challenged claim is directed

Asserted Claims	Representative Claim	Eligibility Exception
1-5, 12-17, 24	1	Abstract Idea

Asserted Claims	Identification of Exception to Eligibility
1. A method for assessing runtime risk for an application program that executes on a device, comprising: storing, in a rules database, a plurality of rules, wherein	The '517 patent's asserted claims recite an abstract framework for managing a security policy, without reciting any technological improvement. Claim 1—one of only two independent claims—recites a method for (i) storing "rules" that correspond to an "action sequence"; (ii) storing "assessment policies"; (iii) identifying "a runtime risk"; and (iv) identifying "a behavior score."

each rule identifies an action sequence; storing, in a policy database, a plurality of assessment policies, wherein each assessment policy includes at least one rule of the plurality of rules; identifying, using at least one assessment policy, a runtime risk for an application program that executes on a device, wherein the identified runtime risk indicates a risk or threat of the identified action sequence of the application; and identifying, by a runtime monitor including a processing device, a behavior score for the application program that executes on the device based on the identified runtime risk, wherein the action sequence is a sequence of at least two performed actions, and each performed action is at least one of: a user action, an application action, and a system action.	Claim 1 is directed to the abstract idea of assigning a score based on risk. The claim assigns the "runtime risk," and ultimately the "behavior score," based on known (but unspecified) "rules" and "policies," which could be assessed mentally by a human. Humans routinely assign scores based on rules to assess risk, such as credit ratings/scores, insurance rates, and medical prognoses. Claim 1 is similar to those invalidated in *Dropbox, Inc. v. Synchronoss Technologies, Inc.*, which were directed to abstract ideas including "associating a security level with a data resource." 815 F. App'x 529, 532 (Fed. Cir. 2020). Like the □517 patent's "runtime risk" and "behavior score," the claim in *Dropbox* recited an "access checker" that invoked "a sensitivity level" and "a trust level" in deciding whether to grant access to computer resources (*id*. at 532), which the Federal Circuit held "offers nothing but a functional abstraction" (*id*. at 533). *See also Cisco Sys., Inc. v. Uniloc 2017 LLC*, 813 F. App'x 495, 497 (Fed. Cir. 2020) (invalidating claims "directed to the abstract idea of ranking stations based on antenna performance characteristics") (quotation omitted); *SmartGene, Inc. v. Advanced Biological Labs., SA*, 555 F. App'x 950, 952 (Fed. Cir. 2014) (invalidating claims that invoked "plurality of expert rules" to generate "a ranked listing"). Confirming that claim 1 is abstract, it does not limit how the claimed method must "identify[]" the "runtime risk for an application program" or the "behavior score for the application program." Dkt. No. 1-5 at claim 1; *Elec. Power Grp*., 830 F.3d at 1356. Nor does it recite any improvement to computer functionality or solve a computer-specific problem. The only steps relate to "storing" and "identifying" information. The claim does not require doing anything with this information, let alone in a way that provides a "specific asserted improvement in computer capabilities," as opposed to "computers . . . invoked merely as a tool." *Smart Sys. Innovations, LLC v. Chicago Transit Auth*., 873 F.3d 1364, 1372 (Fed. Cir. 2017) (quotations omitted).
2. The method of claim 1, wherein the user action is any form of explicit input from a user of a computing system.	The only other independent claim—claim 13—simply restates the same steps in the context of a "system." *Compare* Dkt. No. 1-5 at claim 1 *with id.* at claim 13. The dependent claims also add nothing material. Claims 2–5 and 14–17 limit the "actions" in the "action sequence" assessed for runtime risk (Dkt. No. 1-5), but such "limitation of the claims to a particular field of information—here, [actions evaluated for risk]—does not move the claims out of the realm of abstract ideas." *SAP*, 898 F.3d at 1169. Claims 12 and 24 add a step of "assessing" whether actions "describe a forensic chain of events," but nothing prevents this from being performed by a human. Thus,
3. The method of claim 1, wherein the application action is any activity performed by the application initiated programmatically by a task in execution of a computing system.	

4. The method of claim 1, wherein the system action is any operation performed by a computing system on behalf of, or as a consequence of, a user action or application action that changes the state of the computing system.	claim 1 is representative of all asserted claims, which are abstract.
5. The method of claim 1, wherein a first action of the at least two performed actions is performed at a first time, and the second action of the at least two performed actions is performed at a second time.	
12. The method of claim 1, further comprising: assessing, by the processing device, that the at least two performed actions of the action sequence describe a forensic chain of events based on a plurality of factors, wherein the plurality of factors includes at least one of: ordered sequences, unordered sequences, proximity analysis, outliers, probabilistic weights and scores, and probable action inferences based on an application's competency.	
13. A system for assessing runtime risk for an application program that executes on a device, comprising: a rules database storing a plurality of rules, wherein each rule identifies an action sequence; a policy database storing a plurality of assessment policies, wherein each assessment policy includes at	

least one rule of the plurality of rules; and a runtime monitor including a processing device identifying, using at least one assessment policy, a runtime risk for an application program that executes on a device, wherein the identified runtime risk indicates a risk or threat of the identified action sequence of the application, and identifying a behavior score for the application program that executes on the device based on the identified runtime risk, wherein the action sequence is a sequence of at least two performed actions, and each performed action is at least one of: a user action, an application action, and a system action.	
14. The system of claim 13, wherein the user action is any form of explicit input from a user of a computing system.	
15. The system of claim 13, wherein the application action is any activity performed by the application initiated programmatically by a task in execution of a computing system.	
16. The system of claim 13, wherein the system action is any operation performed by a computing system on behalf of, or as a consequence of, a user action or application action that changes the state of the computing system.	

17. The system of claim 13, wherein a first action of the at least two performed actions is performed at a first time, and the second action of the at least two performed actions is performed at a second time.	
24. The system of claim 13, wherein the processing device assesses that the at least two performed actions of the action sequence describe a forensic chain of events based on a plurality of factors, and the plurality of factors includes at least one of: ordered sequences, unordered sequences, proximity analysis, outliers, probabilistic weights and scores, and probable action inferences based on an application's competency.	

b. Chart identifying a description of the industry at the relevant time and how each element of each challenged claim, individually and in combination with the other elements, was well understood, routine, and conventional.

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
1. A method for assessing runtime risk for an application program that executes on a device, comprising: storing, in a rules database, a plurality of rules, wherein each rule identifies an action sequence; storing, in a policy database, a plurality of assessment policies, wherein each	None of the ’517 patent’s asserted claims recite an inventive concept. The “rules,” “policies,” “runtime risk,” “behavior score,” and “actions” are all in the abstract realm. *SAP*, 898 F.3d at 1168. While the claims recite a “device,” “database[s],” and “processing device,” these generic elements are insufficient. *Id*. at 1169–70 (“Some of the claims require various databases and processors,” but these “require no improved computer resources”). Moreover, the ’517 patent admits that no specialized technology is needed to practice the claims, and “other configurations or systems for performing the functions disclosed may be suitable.” Dkt. No. 1-5 at 2:43–54; *see also id*. at 2:58–62. Thus, all asserted claims of the ’517 patent fail both steps of the *Alice* test.

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
assessment policy includes at least one rule of the plurality of rules; identifying, using at least one assessment policy, a runtime risk for an application program that executes on a device, wherein the identified runtime risk indicates a risk or threat of the identified action sequence of the application; and identifying, by a runtime monitor including a processing device, a behavior score for the application program that executes on the device based on the identified runtime risk, wherein the action sequence is a sequence of at least two performed actions, and each performed action is at least one of: a user action, an application action, and a system action.	Defendant further incorporates by reference the above-identified Applicant Admitted Prior Art, wherein the ’517 patent itself admits that the elements of the asserted claims were well-understood, routine, and conventional. Further, while prior art references are unnecessary to conclude that the asserted claims fail *Alice* step two, Defendant incorporates by reference each of its invalidity charts for the ’517 patent (Exhibit C), which identify prior art teachings that further confirm that the elements of the asserted claims, individually or when combined, were well-understood, routine, and conventional in the industry.
2. The method of claim 1, wherein the user action is any form of explicit input from a user of a computing system.	*See* claim 1, above.
3. The method of claim 1, wherein the application action is any activity performed by the application initiated programmatically by a task in execution of a computing system.	
4. The method of claim 1, wherein the system action is any operation performed by a computing system on behalf of, or as a consequence of, a user action or application	

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
action that changes the state of the computing system.	
5. The method of claim 1, wherein a first action of the at least two performed actions is performed at a first time, and the second action of the at least two performed actions is performed at a second time.	
12. The method of claim 1, further comprising: assessing, by the processing device, that the at least two performed actions of the action sequence describe a forensic chain of events based on a plurality of factors, wherein the plurality of factors includes at least one of: ordered sequences, unordered sequences, proximity analysis, outliers, probabilistic weights and scores, and probable action inferences based on an application's competency.	
13. A system for assessing runtime risk for an application program that executes on a device, comprising: a rules database storing a plurality of rules, wherein each rule identifies an action sequence; a policy database storing a plurality of assessment policies, wherein each assessment policy includes at least one rule of the plurality of rules; and	

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
a runtime monitor including a processing device identifying, using at least one assessment policy, a runtime risk for an application program that executes on a device, wherein the identified runtime risk indicates a risk or threat of the identified action sequence of the application, and identifying a behavior score for the application program that executes on the device based on the identified runtime risk, wherein the action sequence is a sequence of at least two performed actions, and each performed action is at least one of: a user action, an application action, and a system action.	
14. The system of claim 13, wherein the user action is any form of explicit input from a user of a computing system.	
15. The system of claim 13, wherein the application action is any activity performed by the application initiated programmatically by a task in execution of a computing system.	
16. The system of claim 13, wherein the system action is any operation performed by a computing system on behalf of, or as a consequence of, a user action or application action that changes the state of the computing system.	

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
17. The system of claim 13, wherein a first action of the at least two performed actions is performed at a first time, and the second action of the at least two performed actions is performed at a second time.	
24. The system of claim 13, wherein the processing device assesses that the at least two performed actions of the action sequence describe a forensic chain of events based on a plurality of factors, and the plurality of factors includes at least one of: ordered sequences, unordered sequences, proximity analysis, outliers, probabilistic weights and scores, and probable action inferences based on an application's competency.	

Defendant incorporates its Motion for Partial Judgment on the Pleadings in Case No. 2:22-cv-00314 and 2:22-MD-03042-JRG, including all related briefing and oral argument, showing the factual and legal basis that support that the challenged ’517 patent claims are patent-ineligible under 35 U.S.C. § 101. Defendant further incorporates by reference the above-identified Applicant Admitted Prior Art, wherein the ’517 patent itself admits that certain technologies related to the elements of the asserted claims were well-understood, routine, and conventional. Defendant also incorporates each of its invalidity charts for the ’517 patent (Exhibit C), which may additionally

identify, among others, how the elements of the asserted claims, individually or when combined, were well-understood, routine, and conventional in the industry at the relevant time frame.[34]

5. '038 and '918 Patents

a. Chart identifying each exception to eligibility to which each challenged claim is directed

Asserted Claims	Representative Claim	Eligibility Exception
'038: 1-33	1	Abstract Idea
'918: 1-24	1	

Asserted Claims	Identification of Exception to Eligibility
'038 Patent Claim 1. A method for controlling the operation of an endpoint, comprising: providing a user interface, at a computing system remote from the end point, configured to allow configuration of a plurality of policies; maintaining the plurality of policies in a data store on the computing system; identifying, from the plurality of policies, a plurality of operating conditions on the endpoint to monitor; configuring one or more software agents on the endpoint to monitor the plurality of operating conditions;	The '038 and '918 patents are related and share a common specification. The specification admits that the claims recite "methods and systems for controlling access to computing resources based on known computing security vulnerabilities" (Dkt. No. 1-6 at 1:22–25; *see also id*. at 3:36–38), which is an abstract idea. *See Ericsson Inc. v. TCL Commc'n Tech. Holdings Ltd.*, 955 F.3d 1317, 1326 (Fed. Cir. 2020) (invalidating claims "directed to the abstract idea of controlling access to, or limiting permission to, resources" using a "security access manager"). As the Federal Circuit has explained, determining or conditioning access on compliance with a policy "is pervasive in human activity, whether in libraries (loaning materials only to card-holding members), office buildings (allowing certain employees entrance to only certain floors), or banks (offering or denying loans to applicants based on suitability and intended use)." *Id*. at 1327. Here, the claims purport to control "remote electronic access to [ ] networks and computing resources, typically referred to as endpoint access control." Dkt. No. 1-6 at 1:53–55. Claim 1 of the '038 patent below recites a generic method to monitor endpoints for compliance: Claim 1 of the '918 patent merely replaces the final "initiating" step (*id*.) with "authorizing access by the endpoint to a

[34] Defendants refer to its exemplary identification of motivations to combine and its invalidity charts solely for the purpose of complying with the Court's standing order. Defendant's reference to its exemplary identification of motivation to combine or its invalidity charts is not intended to be, and shall not be construed as, an admission that an analysis under 35 U.S.C. 102 (anticipation) or 103 (obviousness) is relevant or necessary to determine the patent eligibility of the asserted claims.

Asserted Claims	Identification of Exception to Eligibility
receiving, across a network, at the computing system, status information about the plurality of operating conditions on the endpoint gathered by the one or more software agents; determining, by the computing system, a compliance state of the endpoint based on the status information and a plurality of compliance policies in the data store; and initiating, by the computing system, based on the compliance state, an action identified in at least one rule in the data store, wherein the action is carried out by a processor on the endpoint.	computing resource . . . in response to the compliance state; and continuing to monitor the compliance state by the endpoint and restricting access to the computing resource if the compliance state changes" (Dkt. No. 1-9 at claim 1). These claims are directed to the abstract idea of determining compliance with a policy and (for the '918 patent) restricting access to resources. "[W]ith the exception of generic computer implemented steps," this could be "performed by a human." *Symantec*, 838 F.3d at 1318. Claims 1 of the '038 and '918 patents are similar to those invalidated in *Ericsson*, which recited "receiving a request . . . to access the software services" and "determining if the request should be granted" based on "access and permission policies." *Id*. at 1326–27. "Although written in technical jargon," the claims were "directed to the abstract idea of controlling access to, or limiting permission to, resources." *Id*. at 1326; *see also In re Gale*, 856 F. App'x 887, 889 (Fed. Cir. 2021) (invalidating claims "directed to the abstract idea of (1) collecting information (here, receiving messages and reading their metadata), (2) analyzing the information (here, calculating a usage pattern and determining its compliance with a predetermined usage pattern), and (3) reporting the results"). Moreover, nothing limits how the "software agents . . . monitor the plurality of operating conditions," how they determine "status information about the plurality of operating conditions," or how the system "determin[es] . . . a compliance state of the endpoint." Thus, as in *Ericsson*, the claims do not recite "a specific technique for improving computer performance" under *Enfish* or *DDR* because the claims "are silent as to how access is controlled." 955 F.3d at 1328.
'918 Patent Claim 1. 1. A method for controlling the operation of an endpoint, comprising: providing a user interface, at a computing system that is remote from the endpoint, configured to allow configuration of a plurality of policies; maintaining the plurality of policies in a data store on the computing system; identifying, from the plurality of policies, a plurality of operating conditions on the endpoint to evaluate; configuring one or more software services provided by an operating system on the endpoint to monitor the plurality of operating conditions;	

Asserted Claims	Identification of Exception to Eligibility
receiving, across a network, at the computing system, status information about the plurality of operating conditions on the endpoint, gathered by the one or more software services on the endpoint, and user information that identifies a user of the endpoint; determining, by the computing system, a compliance state of the endpoint based on the user information and status information, and a plurality of compliance policies in the data store; authorizing access by the endpoint to a computing resource on the network, authorization being determined by the remote computing system in response to the compliance state; and continuing to monitor the compliance state by the endpoint and restricting access to the computing resource if the compliance state changes.	
’038 Patent Claims: 2. The method of claim 1, wherein the action comprises controlling access of the endpoint to computing resources. 3. The method of claim 1, wherein the user interface comprises a web page. 4. The method of claim 1, further comprising requesting, at the computing	The other independent claims recite the same process in the form of “[a] non-transitory computer readable medium” (’038 claim 12; ’918 claim 9) or a “system” (’038 claim 23; ’918 claim 17) but are otherwise identical. And like claim 1 of the ’918 patent, claims 2, 13, and 24 of the ’038 patent require “controlling access of the endpoint to computing resources” (Dkt. No. 1-6), which is an abstract idea (*Ericsson*, 955 F.3d at 1326). Most other claims recite variations on determining compliance with a policy.[3] They merely limit analyzed data to “a particular field of information,” which is immaterial. *SAP*, 898 F.3d at 1169. The other claims recite more conventional components: a “web page” (’038 claims 3, 14, 25; ’918 claims 2, 10, 18); “mobile device” (’038 claims 5, 16, 27; ’918 claims

Asserted Claims	Identification of Exception to Eligibility
system, the status information on a periodic basis. 5. The method of claim 1, wherein the endpoint comprises a mobile device. 6. The method of claim 1, wherein the one or more software agents comprise at least one service provided by the operating system of the endpoint. 7. The method of claim 1, wherein the one or more software agents comprise at least one application running on the endpoint. 8. The method of claim 1, wherein the conditions comprise at least one hardware condition. 9. The method of claim 1, wherein the conditions comprise at least one software condition. 10. The method of claim 1, wherein the computing system comprises a plurality of servers. 11. The method of claim 1, wherein the plurality of policies includes at least one policy that includes the at least one rule that identifies the action. 12. A non-transitory computer readable medium containing computer instructions for controlling the operation of an endpoint, comprising: providing a user interface, at a computing system remote from the end point, configured to allow	4, 12, 20); “servers” (’038 claims 10, 21, 32; ’918 claims 8, 16, 24); and “software agents” provided by the operating system (’038 claims 6, 17, 28) or endpoint applications (’038 claims 7, 18, 29).

Asserted Claims	Identification of Exception to Eligibility
configuration of a plurality of policies; maintaining the plurality of policies in a data store on the computing system; identifying, from the plurality of policies, a plurality of operating conditions on the endpoint to monitor; configuring one or more software agents on the endpoint to monitor the plurality of operating conditions; receiving, across a network, at the computing system, status information about the plurality of operating conditions on the endpoint gathered by the one or more software agents; determining, by the computing system, a compliance state of the endpoint based on the status information and a plurality of compliance policies in the data store; and initiating, by the computing system, based on the compliance state, an action identified in at least one rule in the data store, wherein the action is carried out by a processor on the endpoint. 13. The computer readable medium of claim 12, wherein the action comprises controlling access of the endpoint to computing resources. 14. The computer readable medium of claim 12, wherein	

Asserted Claims	Identification of Exception to Eligibility
the user interface comprises a web page. 15. The computer readable medium of claim 12, further comprising requesting, at the computing system, the status information on a periodic basis. 16. The computer readable medium of claim 12, wherein the endpoint comprises a mobile device. 17. The computer readable medium of claim 12, wherein the one or more software agents comprise at least one service provided by the operating system of the endpoint. 18. The computer readable medium of claim 12, wherein the one or more software agents comprise at least one application running on the endpoint. 19. The computer readable medium of claim 12, wherein the conditions comprise at least one hardware condition. 20. The computer readable medium of claim 12, wherein the conditions comprise at least one software condition. 21. The computer readable medium of claim 12, wherein the computing system comprises a plurality of servers. 22. The computer readable medium of claim 12, wherein the plurality of policies includes at least one policy that includes the at least one rule that identifies the action.	

Asserted Claims	Identification of Exception to Eligibility
23. A system for controlling the operation of an endpoint, comprising: a user interface, provided by a computing system remote from the end point, configured to allow configuration of a plurality of policies; a data store, at the computing system, that contains the plurality of policies; one or more software agents on the endpoint configured to monitor a plurality of operating conditions identified in the plurality of policies; and one or more hardware processors at the computing system configured to: receive, across a network, status information about the plurality of operating conditions on the endpoint gathered by the one or more software agents, determine a compliance state of the endpoint based on the status information and a plurality of compliance policies in the data store, and initiate, based on the compliance state, an action identified in at least one rule in the data store, wherein the action is carried out by the hardware processor on the endpoint. 24. The system of claim 23, wherein the action comprises controlling access of the endpoint to computing resources.	

Asserted Claims	Identification of Exception to Eligibility
25. The system of claim 23, wherein the user interface comprises a web page. 26. The system of claim 23, wherein the one or more processors are further configured to request the status information from the endpoint on a periodic basis. 27. The system of claim 23, wherein the endpoint comprises a mobile device. 28. The system of claim 23, wherein the one or more software agents comprise at least one service provided by the operating system of the endpoint. 29. The system of claim 23, wherein the one or more software agents comprise at least one application running on the endpoint. 30. The system of claim 23, wherein the conditions comprise at least one hardware condition. 31. The system of claim 23, wherein the conditions comprise at least one software condition. 32. The system of claim 23, wherein the computing system comprises a plurality of servers. 33. The system of claim 23, wherein the plurality of policies includes at least one policy that includes the at least one rule that identifies the action.	
‘918 Patent Claims:	

Asserted Claims	Identification of Exception to Eligibility
2. The method of claim 1, wherein the user interface comprises a web page. 3. The method of claim 1, further comprising requesting, at the computing system, the status information on a periodic basis. 4. The method of claim 1, wherein the endpoint comprises a mobile device. 5. The method of claim 1, further comprising configuring one or more applications running on the endpoint on the endpoint to monitor at least a subset of the plurality of operating conditions. 6. The method of claim 1, wherein the conditions comprise at least one hardware condition. 7. The method of claim 1, wherein the conditions comprise at least one software condition. 8. The method of claim 1, wherein the computing system comprises a plurality of servers. 9. A non-transitory computer readable medium containing computer instructions for controlling the operation of an endpoint, comprising: providing a user interface, at a computing system that is remote from the endpoint, configured to allow configuration of a plurality of policies;	

Asserted Claims	Identification of Exception to Eligibility
maintaining the plurality of policies in a data store on the computing system; identifying, from the plurality of policies, a plurality of operating conditions on the endpoint to evaluate; configuring one or more software services provided by an operating system on the endpoint to monitor the plurality of operating conditions; receiving, across a network, at the computing system, status information about the plurality of operating conditions on the endpoint, gathered by the one or more software services on the endpoint, and user information that identifies a user of the endpoint; determining, by the computing system, a compliance state of the endpoint based on the user information and status information, and a plurality of compliance policies in the data store; authorizing access by the endpoint to a computing resource on the network, authorization being determined by the remote computing system in response to the compliance state; and continuing to monitor the compliance state by the endpoint and restricting access to the computing resource if the compliance state changes.	

Asserted Claims	Identification of Exception to Eligibility
10. The non-transitory computer readable medium of claim 9, wherein the user interface comprises a web page. 11. The non-transitory computer readable medium of claim 9, further comprising requesting, at the computing system, the status information on a periodic basis. 12. The non-transitory computer readable medium of claim 9, wherein the endpoint comprises a mobile device. 13. The non-transitory computer readable medium of claim 9, further comprising configuring one or more applications running on the endpoint on the endpoint to monitor at least a subset of the plurality of operating conditions. 14. The non-transitory computer readable medium of claim 9, wherein the conditions comprise at least one hardware condition. 15. The non-transitory computer readable medium of claim 9, wherein the conditions comprise at least one software condition. 16. The non-transitory computer readable medium of claim 9, wherein the computing system comprises a plurality of servers. 17. A system for controlling the operation of an endpoint, comprising: a user interface, provided by a computing system remote	

Asserted Claims	Identification of Exception to Eligibility
from the end point, configured to allow configuration of a plurality of policies; a data store, at the computing system, that contains the plurality of policies; one or more software services provided by an operating system on the endpoint configured to evaluate a plurality of operating conditions identified in the plurality of policies; and one or more hardware processors at the computing system configured to: receive, across a network, at the computing system, status information about the plurality of operating conditions on the endpoint, gathered by the one or more software services on the endpoint, and user information that identifies a user of the endpoint, determine, by the computing system, a compliance state of the endpoint based on the user information and status information, and a plurality of compliance policies in the data store, and authorize access by the endpoint to a computing resource on the network, authorization being determined by the remote computing system in response to the compliance state. 18. The system of claim 17, wherein the user interface comprises a web page.	

Asserted Claims	Identification of Exception to Eligibility
19. The system of claim 17, further comprising requesting, at the computing system, the status information on a periodic basis. 20. The system of claim 17, wherein the endpoint comprises a mobile device. 21. The system of claim 17, further comprising configuring one or more applications running on the endpoint on the endpoint to monitor at least a subset of the plurality of operating conditions. 22. The system of claim 17, wherein the conditions comprise at least one hardware condition. 23. The system of claim 17, wherein the conditions comprise at least one software condition. 24. The system of claim 17, wherein the computing system comprises a plurality of servers.	

b. Chart identifying a description of the industry at the relevant time and how each element of each challenged claim, individually and in combination with the other elements, was well understood, routine, and conventional.

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
’038 Patent Claim 1. A method for controlling the operation of an endpoint, comprising: providing a user interface, at a computing system remote from the end point, configured to allow	The claims also fail *Alice* step two. Apart from the recited categories of information, which are in the abstract realm (e.g., “policies,” “rules,” “operating conditions,” “status information,” “compliance state[s]”), the computer and network components are all conventional. The independent claims recite an “endpoint”—a catch-all term for “remote electronic access” from another device. Dkt. No. 1-6 at 1:53–57; *see id*. at 7:52–64 (“endpoint system” can be “any

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
configuration of a plurality of policies; maintaining the plurality of policies in a data store on the computing system; identifying, from the plurality of policies, a plurality of operating conditions on the endpoint to monitor; configuring one or more software agents on the endpoint to monitor the plurality of operating conditions; receiving, across a network, at the computing system, status information about the plurality of operating conditions on the endpoint gathered by the one or more software agents; determining, by the computing system, a compliance state of the endpoint based on the status information and a plurality of compliance policies in the data store; and initiating, by the computing system, based on the compliance state, an action identified in at least one rule in the data store, wherein the action is carried out by a processor on the endpoint.	processing system" made up of "conventional components"). The claims also recite a "computing system," "data store," "software agents [or services]," "network," and "processor" (e.g., Dkt. No. 1-6 at claim 1; Dkt. No. 1-9 at claim 1), but these invoke only "already available computers" (*SAP*, 898 F.3d at 1169–70) that were conventional and are used in the claims in their conventional manner. *See* Dkt. No. 1-6 at 1:30–43 (network connections "are ubiquitous in the industry and well known to the reader"), 7:24–29 (using, "in a conventional manner, a processor" and "conventional storage components"), 9:6–9 (using "the accepted definition in the art" for software agents), 59:6–7 ("conventional data communications network"). The generic elements in the dependent claims—i.e., "web page" ('038 claims 3, 14, 25; '918 claims 2, 10, 18), "mobile device" ('038 claims 5, 16, 27; '918 claims 4, 12, 20), "servers" ('038 claims 10, 21, 32; '918 claims 8, 16, 24), and software agents within the operating system ('038 claims 6, 17, 28) or endpoint applications ('038 claims 7, 18, 29)—are also conventional and used in a conventional manner. Defendant further incorporates by reference the above-identified Applicant Admitted Prior Art, wherein the '038 and '918 patents admit that the elements of the asserted claims were well-understood, routine, and conventional. Further, while prior art references are unnecessary to conclude that the asserted claims fail *Alice* step two, Defendant incorporates by reference each of its invalidity charts for the '038 and '918 patent (Exhibits E and I), which identify prior art teachings that further confirm that the elements of the asserted claims, individually or when combined, were well-understood, routine, and conventional in the industry.
'918 Patent Claim 1. 1. A method for controlling the operation of an endpoint, comprising: providing a user interface, at a computing system that is remote from the endpoint, configured to allow	

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
configuration of a plurality of policies; maintaining the plurality of policies in a data store on the computing system; identifying, from the plurality of policies, a plurality of operating conditions on the endpoint to evaluate; configuring one or more software services provided by an operating system on the endpoint to monitor the plurality of operating conditions; receiving, across a network, at the computing system, status information about the plurality of operating conditions on the endpoint, gathered by the one or more software services on the endpoint, and user information that identifies a user of the endpoint; determining, by the computing system, a compliance state of the endpoint based on the user information and status information, and a plurality of compliance policies in the data store; authorizing access by the endpoint to a computing resource on the network, authorization being determined by the remote computing system in response to the compliance state; and continuing to monitor the compliance state by the endpoint and restricting access to the computing	

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
resource if the compliance state changes.	
’038 Patent Claims: 2. The method of claim 1, wherein the action comprises controlling access of the endpoint to computing resources. 3. The method of claim 1, wherein the user interface comprises a web page. 4. The method of claim 1, further comprising requesting, at the computing system, the status information on a periodic basis. 5. The method of claim 1, wherein the endpoint comprises a mobile device. 6. The method of claim 1, wherein the one or more software agents comprise at least one service provided by the operating system of the endpoint. 7. The method of claim 1, wherein the one or more software agents comprise at least one application running on the endpoint. 8. The method of claim 1, wherein the conditions comprise at least one hardware condition. 9. The method of claim 1, wherein the conditions comprise at least one software condition. 10. The method of claim 1, wherein the computing system comprises a plurality of servers.	*See* ’038 Patent Claim 1 and ’918 Patent Claim 1, above.

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
11. The method of claim 1, wherein the plurality of policies includes at least one policy that includes the at least one rule that identifies the action. 12. A non-transitory computer readable medium containing computer instructions for controlling the operation of an endpoint, comprising: providing a user interface, at a computing system remote from the end point, configured to allow configuration of a plurality of policies; maintaining the plurality of policies in a data store on the computing system; identifying, from the plurality of policies, a plurality of operating conditions on the endpoint to monitor; configuring one or more software agents on the endpoint to monitor the plurality of operating conditions; receiving, across a network, at the computing system, status information about the plurality of operating conditions on the endpoint gathered by the one or more software agents; determining, by the computing system, a compliance state of the endpoint based on the status information and a plurality of compliance policies in the data store; and	

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
initiating, by the computing system, based on the compliance state, an action identified in at least one rule in the data store, wherein the action is carried out by a processor on the endpoint. 13. The computer readable medium of claim 12, wherein the action comprises controlling access of the endpoint to computing resources. 14. The computer readable medium of claim 12, wherein the user interface comprises a web page. 15. The computer readable medium of claim 12, further comprising requesting, at the computing system, the status information on a periodic basis. 16. The computer readable medium of claim 12, wherein the endpoint comprises a mobile device. 17. The computer readable medium of claim 12, wherein the one or more software agents comprise at least one service provided by the operating system of the endpoint. 18. The computer readable medium of claim 12, wherein the one or more software agents comprise at least one application running on the endpoint. 19. The computer readable medium of claim 12, wherein the conditions comprise at least one hardware condition.	

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
20. The computer readable medium of claim 12, wherein the conditions comprise at least one software condition. 21. The computer readable medium of claim 12, wherein the computing system comprises a plurality of servers. 22. The computer readable medium of claim 12, wherein the plurality of policies includes at least one policy that includes the at least one rule that identifies the action. 23. A system for controlling the operation of an endpoint, comprising: a user interface, provided by a computing system remote from the end point, configured to allow configuration of a plurality of policies; a data store, at the computing system, that contains the plurality of policies; one or more software agents on the endpoint configured to monitor a plurality of operating conditions identified in the plurality of policies; and one or more hardware processors at the computing system configured to: receive, across a network, status information about the plurality of operating conditions on the endpoint gathered by the one or more software agents,	

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
determine a compliance state of the endpoint based on the status information and a plurality of compliance policies in the data store, and initiate, based on the compliance state, an action identified in at least one rule in the data store, wherein the action is carried out by the hardware processor on the endpoint. 24. The system of claim 23, wherein the action comprises controlling access of the endpoint to computing resources. 25. The system of claim 23, wherein the user interface comprises a web page. 26. The system of claim 23, wherein the one or more processors are further configured to request the status information from the endpoint on a periodic basis. 27. The system of claim 23, wherein the endpoint comprises a mobile device. 28. The system of claim 23, wherein the one or more software agents comprise at least one service provided by the operating system of the endpoint. 29. The system of claim 23, wherein the one or more software agents comprise at least one application running on the endpoint. 30. The system of claim 23, wherein the conditions	

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
comprise at least one hardware condition. 31. The system of claim 23, wherein the conditions comprise at least one software condition. 32. The system of claim 23, wherein the computing system comprises a plurality of servers. 33. The system of claim 23, wherein the plurality of policies includes at least one policy that includes the at least one rule that identifies the action.	
'918 Patent Claims: 2. The method of claim 1, wherein the user interface comprises a web page. 3. The method of claim 1, further comprising requesting, at the computing system, the status information on a periodic basis. 4. The method of claim 1, wherein the endpoint comprises a mobile device. 5. The method of claim 1, further comprising configuring one or more applications running on the endpoint on the endpoint to monitor at least a subset of the plurality of operating conditions. 6. The method of claim 1, wherein the conditions comprise at least one hardware condition. 7. The method of claim 1, wherein the conditions	

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
comprise at least one software condition. 8. The method of claim 1, wherein the computing system comprises a plurality of servers. 9. A non-transitory computer readable medium containing computer instructions for controlling the operation of an endpoint, comprising: providing a user interface, at a computing system that is remote from the endpoint, configured to allow configuration of a plurality of policies; maintaining the plurality of policies in a data store on the computing system; identifying, from the plurality of policies, a plurality of operating conditions on the endpoint to evaluate; configuring one or more software services provided by an operating system on the endpoint to monitor the plurality of operating conditions; receiving, across a network, at the computing system, status information about the plurality of operating conditions on the endpoint, gathered by the one or more software services on the endpoint, and user information that identifies a user of the endpoint; determining, by the computing system, a compliance state of the endpoint based on the user	

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
information and status information, and a plurality of compliance policies in the data store; authorizing access by the endpoint to a computing resource on the network, authorization being determined by the remote computing system in response to the compliance state; and continuing to monitor the compliance state by the endpoint and restricting access to the computing resource if the compliance state changes. 10. The non-transitory computer readable medium of claim 9, wherein the user interface comprises a web page. 11. The non-transitory computer readable medium of claim 9, further comprising requesting, at the computing system, the status information on a periodic basis. 12. The non-transitory computer readable medium of claim 9, wherein the endpoint comprises a mobile device. 13. The non-transitory computer readable medium of claim 9, further comprising configuring one or more applications running on the endpoint on the endpoint to monitor at least a subset of the plurality of operating conditions. 14. The non-transitory computer readable medium	

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
of claim 9, wherein the conditions comprise at least one hardware condition. 15. The non-transitory computer readable medium of claim 9, wherein the conditions comprise at least one software condition. 16. The non-transitory computer readable medium of claim 9, wherein the computing system comprises a plurality of servers. 17. A system for controlling the operation of an endpoint, comprising: a user interface, provided by a computing system remote from the end point, configured to allow configuration of a plurality of policies; a data store, at the computing system, that contains the plurality of policies; one or more software services provided by an operating system on the endpoint configured to evaluate a plurality of operating conditions identified in the plurality of policies; and one or more hardware processors at the computing system configured to: receive, across a network, at the computing system, status information about the plurality of operating conditions on the endpoint, gathered by the one or more software services on the endpoint, and user	

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
information that identifies a user of the endpoint, determine, by the computing system, a compliance state of the endpoint based on the user information and status information, and a plurality of compliance policies in the data store, and authorize access by the endpoint to a computing resource on the network, authorization being determined by the remote computing system in response to the compliance state. 18. The system of claim 17, wherein the user interface comprises a web page. 19. The system of claim 17, further comprising requesting, at the computing system, the status information on a periodic basis. 20. The system of claim 17, wherein the endpoint comprises a mobile device. 21. The system of claim 17, further comprising configuring one or more applications running on the endpoint on the endpoint to monitor at least a subset of the plurality of operating conditions. 22. The system of claim 17, wherein the conditions comprise at least one hardware condition. 23. The system of claim 17, wherein the conditions comprise at least one software condition.	

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
24. The system of claim 17, wherein the computing system comprises a plurality of servers.	

Defendant incorporates its Motion for Partial Judgment on the Pleadings in Case No. 2:22-cv-00314 and 2:22-MD-03042-JRG, including all related briefing and oral argument, showing the factual and legal basis that support that the challenged ’038 and ’918 patent claims are patent-ineligible under 35 U.S.C. § 101. Defendant further incorporates by reference the above-identified Applicant Admitted Prior Art, wherein the ’038 and ’918 patents admit that certain technologies related to the elements of the asserted claims were well-understood, routine, and conventional. Defendant also incorporates each of its invalidity charts for the ’038 and ’918 patents (Exhibits E and I), which may additionally identify, among others, how the elements of the asserted claims, individually or when combined, were well-understood, routine, and conventional in the industry at the relevant time frame.[35]

6. ’948 Patent

Asserted claims 1-5 of the ’948 patent constitute ineligible subject matter under 35 U.S.C. §101. Under the test articulated by the U.S. Supreme Court in *Alice Corp. v. CLS Bank Int’l*, 573 U.S. 208, 217-18 (2014), claims 1-5 are (i) directed to abstract ideas, and (ii) fail to recite any inventive concepts that otherwise would lead to eligibility. Claim 1 of the ’948 patent is representative:

> A method of providing real-time operational integrity of an application on a native computing environment, the method comprising:

[35] Defendants refer to its exemplary identification of motivations to combine and its invalidity charts solely for the purpose of complying with the Court’s standing order. Defendant’s reference to its exemplary identification of motivation to combine or its invalidity charts is not intended to be, and shall not be construed as, an admission that an analysis under 35 U.S.C. 102 (anticipation) or 103 (obviousness) is relevant or necessary to determine the patent eligibility of the asserted claims.

monitoring, by a plurality of sensory inputs, one or more of network dialogs of the application, system operations initiated by the application, a runtime configuration of the application, resource utilization by the application, and integrity of the application;

generating real-time behavior based events for determining the real-time operational integrity of the application executing on the native computing environment which includes a network analyzer, an integrity processor, an event correlation matrix, a risk correlation matrix, and a trust supervisor;

correlating, by the event and risk correlation matrix, threat classifications based on the temporal sequence of the generated real-time behavior based events; and

displaying, in a plurality of runtime dashboards of an administrative console of the computing environment, real-time status indications for operational integrity of the application.

Claim 1 is directed to the abstract idea of monitoring for trustworthiness. For example, someone who is not previously known to be trustworthy can be monitored (for example, by video or audio) to see if that person engages in conduct that would call his or her trustworthiness into question. This may include watching and listening to things the person says or does to gauge whether they can be trusted or not. Monitoring "network dialogs," "system operations," "runtime configuration," "resource utilization," and "integrity" is analogous to simply listening and watching what the person (e.g., an employee) says and does. If the person is not behaving in a certain way, or is taking actions inconsistent with being trustworthy, the system so indicates by way of a real-time status indication.

There is nothing beyond well-understood computer hardware and software underlying the claimed system. As the specification admits, the methods claimed are implemented using standard computing technology and software—well known to those of skill in the art. *See*, *e.g.*, cols. 34-36 (describing well-known computer architectures running standard Microsoft operating systems.).

Nor is there a recitation of any improvement that makes a computer system operate more efficiently or in an advantageous manner over what existed in prior art. Accordingly, claim 1 – 5 of the '948 patent are invalid because they are directed to ineligible subject matter under 35 U.S.C. §101.

7. ’616 Patent

Asserted claims 1-7 of the ’616 patent constitute ineligible subject matter under 35 U.S.C. §101 and are therefore invalid. Under the test articulated by the U.S. Supreme Court in *Alice Corp. v. CLS Bank Int’l*, 573 U.S. 208, 217-18 (2014), claims 1-7 are (i) directed to abstract ideas, and (ii) fail to recite any inventive concepts that otherwise would lead to eligibility. Claim 1 of the ’616 patent is representative:

> A method of providing an attestation service for providing runtime operational integrity of a systems using a computing platform comprising a network trust agent, an endpoint trust agent, and a trust orchestration server, the method comprising:
>
> sending, by the endpoint trust agent on a monitored device, a dynamic context including endpoint events and actions of the monitored device and applications executing on the monitored device at runtime;
>
> receiving, at the trust orchestration server, the dynamic context including the endpoint events of the monitored device and the applications executing on the monitored device at runtime;
>
> analyzing, by the trust orchestration server, the received endpoint events;
>
> receiving, by the trust orchestration server, third party network endpoint assessments;
>
> generating, by the trust orchestration server, temporal events based at least in part on analyzing the third party network endpoint assessments;
>
> correlating, by the trust orchestration server, the received endpoint events and the generated temporal events; and
>
> generating, by the trust orchestration server, an integrity profile for the system.

Claim 1 is directed to the abstract idea of monitoring for trustworthiness. For example, someone who is not previously known to be trustworthy can be monitored (for example, by video or audio) to see if that person engages in conduct that would call his or her trustworthiness into question. This may include watching and listening to things the person says or does to gauge whether they can be trusted or not. Monitoring “temporal events” and “endpoint events” is analogous to simply listening and watching what the person (e.g., an employee) says and does. If the person is not behaving in a certain way, or is taking actions inconsistent with being trustworthy, the system so indicates by way of a real-time status indication.

There is nothing beyond well-understood computer hardware and software underlying the claimed system. As the specification admits, the methods claimed are implemented using standard

computing technology and software—well known to those of skill in the art. *See*, *e.g.*, '616 Patent at cols. 34-36 (describing well-known computer architectures running standard Microsoft operating systems.).

Nor is there a recitation of any improvement that makes a computer system operate more efficiently or in an advantageous manner over what existed in prior art. Accordingly, claims 1 – 7 of the '616 patent are invalid because they are directed to ineligible subject matter under 35 U.S.C. §101.

8. '356 Patent

a. Chart identifying each exception to eligibility to which each challenged claim is directed

Asserted Claims	Representative Claim	Eligibility Exception
1-4, 6, 8-10, 12-20	1	Abstract Idea

Asserted Claims	Identification of Exception to Eligibility
1. A computer program product for automatically determining if a packet is a new, exploit candidate, the computer program product comprising: a computer-readable tangible storage device; first program instructions to determine if the packet is a known exploit; second program instructions to determine if the packet is addressed to a broadcast IP address of a network; third program instructions to determine if the packet is network administration traffic; fourth program instructions, responsive to the packet being a known exploit OR the	All of the Asserted Claims of the '356 Patent, which issued in 2012, are directed to non-statutory subject matter, under 35 U.S.C. § 101 and the case law following *Alice Corp. v. CLS Bank Int'l*, 134 S. Ct. 2347 (2014), which developed after the '356 Patent issued. The Asserted Claims of the '356 Patent merely relate to sorting packets of data using simple Boolean logic—in particular, using a known algorithm that the patent admits was applied by humans and is analogous to a person sorting through their mailbox to discard junk mail. As background, the patent describes its invention as "automatically determining if a packet [of data in a network] is a new, exploit candidate." '356 Patent at 3:11–13. The patent explains that new exploit candidates were previously identified using a "honeypot," i.e., a network-connected device that is "not serving any useful function" besides receiving unsolicited packets. *Id.* at 2:38–50. The patent admits "[i]t is known for a human analyst to analyze all of the packets received by the honeypot" to determine if they are new exploit candidates: (1) the human "analyst will determine which packets are harmless broadcast traffic, network administration, or web crawler requests" (which are not exploits); and (2) the human "analyst will also look for harmful known viruses, worms, and

Asserted Claims	Identification of Exception to Eligibility
packet being addressed to a broadcast IP address of a network OR the packet being network administration traffic, to determine that the packet is not a new, exploit candidate; and fifth program instructions, responsive to the packet not being a known exploit AND the packet not being addressed to a broadcast IP address of a network AND the packet not being network administration traffic AND the packet not being another type of traffic known to be benign, to determine and report that the packet is a new, exploit candidate; and wherein the first, second, third, fourth and fifth program instructions are stored on the computer-readable tangible storage device.	exploitation code contained in the packets" (which are not new exploits). *Id.* at 2:50–57. Claim 1 recites "[a] computer program" automating these same steps. Claim 1 determines three conditions: (i) "if the packet is a known exploit"; (ii) "if the packet is addressed to a broadcast IP address of a network"; and (iii) "if the packet is network administration traffic." Id. Using simple Boolean logic, if any condition is met, the packet is not a new exploit candidate (because it is either a known exploit or trusted data). Conversely, if all three conditions are not met, the packet is a new exploit candidate. Claim 1 is directed to the abstract idea of sorting data—i.e., classifying a packet as either a new exploit candidate or not a new exploit candidate—based on simple if/then logic. Thus, "with the exception of generic computer-implemented steps, there is nothing in the claims themselves that foreclose them from being performed by a human." *Symantec*, 838 F.3d at 1318. The Federal Circuit has compared similar data-sorting claims to the "long-prevalent practice for people receiving paper mail to look at an envelope and discard certain letters . . . based on characteristics of the mail," as "[t]he list of relevant characteristics could be kept in a person's head." *Id.* at 1314. Claim 1 is similar to those invalidated in *Intellectual Ventures I LLC v. Erie Indemnity Co.*, which claimed "a method for 'identifying and characterizing' files based on one of three selection criteria." 711 F. App'x 1012, 1015 (Fed. Cir. 2017). After selecting a file, the claims "characterize[ed] the file as an unauthorized file" if certain conditions were met. *Id*. at 1014. These claims were "directed to the identification of unwanted files in a particular field (i.e., a computer network) and otherwise concern data collection related to such identification, such that they are directed to an abstract idea." *Id*. at 1015. Here, as in *Erie*, there is no evidence that the steps "of selecting errant files apply rules of selection in a manner different from those which humans used, albeit with less efficiency, before the invention." *Id*. at 1016. A computer may be faster than a human ('356 Patent at 2:66–3:2), but that merely reflects the "efficiency inherent with applying the abstract idea on a computer" (*Customedia*, 951 F.3d at 1365)—not a specific improvement to computers (*Enfish*) or a solution to any computer-specific problem (*DDR*). Claim 1 is also abstract because each of its steps merely recites "a desired function or outcome, without providing any limiting detail that confines the claim to a particular solution to an identified problem." *Affinity Labs of Texas, LLC v. Amazon.com, Inc.*, 838 F.3d 1266, 1269 (Fed. Cir. 2016). Each step generically recites "program instructions" that somehow

Asserted Claims	Identification of Exception to Eligibility
	"determine" if conditions are met, without specifying how any of the claimed determinations are made.
2. The computer program product of claim 1 further comprising: sixth program instructions to determine if the packet is web crawler traffic; and wherein the fourth program instructions are responsive to the packet being a known exploit OR the packet being addressed to a broadcast IP address of a network OR the packet being network administration traffic OR the packet being web crawler traffic, to determine that the packet is not a new, exploit candidate; and the fifth program instructions are responsive to the packet not being a known exploit AND the packet not being addressed to a broadcast IP address of a network AND the packet not being network administration traffic AND the packet not being web crawler traffic, to determine that the packet is a new, exploit candidate; and the sixth program instructions are stored on the computer-readable tangible storage device.	All other independent claims (9–11, 13, and 17) are equally abstract. Claim 17 merely reframes the same steps into a generic "computer system" with "one or more processors," "memories," and "storage devices" (*id.*), but such "mere recitation of a generic computer cannot transform a patent-ineligible abstract idea into a patent-eligible invention" (*Alice*, 573 U.S. at 223). The remaining independent claims recite more "determin[ations]" and corresponding if/then statements, which are abstract mental steps for the same reasons discussed above. Claim 9 ("determine if the packet has a protocol listed in a list of protocols previously determined to be harmless network broadcast traffic"); claim 10 ("determine if the packet is network administration traffic by comparing an IP protocol and IP address of the packet to a list of combinations of IP protocols and IP addresses previously determined to be network administration traffic"); claim 11 ("determine if the packet is web crawler traffic by comparing an IP address of the packet to a list of IP addresses of known web crawlers"); claim 13 ("determine if the packet has a protocol listed in a list of protocols previously determined to be harmless broadcast traffic"). The dependent claims tack on more data collection and analysis steps, which do not change the focus of the claims: determining if the packet is "web crawler traffic" (claims 2, 14, 18); searching or determining a "signature" (claims 3, 6, 7, 20); comparing "identities" (claim 4); using a honeypot (claims 5, 19); comparing IP addresses (claims 8, 16); and processing a "sequence of packets" (claims 12, 15). Thus, all claims are represented by claim 1 and fail Alice step one.
3. The computer program product of claim 1 wherein the first program instructions determine if the packet is a known exploit by searching the packet for a known signature of a known exploit.	

Asserted Claims	Identification of Exception to Eligibility
4. The computer program product of claim 1 wherein the first program instructions determine if the packet is a known exploit by comparing an identity of the packet to one or more identities, sent by an intrusion detection system, of respective packet(s) which the intrusion detection system determined to contain a known exploit.	
6. The computer program product of claim 1 further comprising: sixth program instructions, responsive to the fifth program instructions determining that the packet is a new exploit candidate, to determine a signature of the packet, and report the new exploit candidate and the signature to an administrator; and wherein the sixth program instructions are stored on the computer-readable tangible storage device.	
8. The computer program product of claim 1 wherein the second program instructions determine if the packet is addressed to a broadcast IP address of the network by comparing a destination IP address of the packet to a gateway IP address of the network and a netmask of the network which identifies a broadcast IP address of the network.	
9. A computer program product for automatically	

Asserted Claims	Identification of Exception to Eligibility
determining if a packet is a new, exploit candidate, the computer program product comprising: a computer-readable tangible storage device; first program instructions to determine if the packet is a known exploit; second program instructions to determine if the packet is addressed to a broadcast IP address of a network; third program instructions to determine if the packet is network administration traffic; fourth program instructions, responsive to the packet being a known exploit OR the packet being addressed to a broadcast IP address of a network OR the packet being network administration traffic, to determine that the packet is not a new, exploit candidate; and fifth program instructions, responsive to the packet not being a known exploit AND the packet not being addressed to a broadcast IP address of a network AND the packet not being network administration traffic AND the packet not being another type of traffic known to be benign, to determine and report that the packet is a new, exploit candidate; and wherein the first, second, third, fourth and fifth program instructions are stored on the computer-	

Asserted Claims	Identification of Exception to Eligibility
readable tangible storage device; the second program instructions also determine if the packet has a protocol listed in a list of protocols previously determined to be harmless network broadcast traffic; the fourth program instructions are responsive to the packet being a known exploit OR the packet being addressed to a broadcast IP address of a network OR the packet being network administration traffic OR the packet having a protocol listed in a list of protocols previously determined to be harmless network broadcast traffic, to determine that the packet is not a new, exploit candidate; and the fifth program instructions are responsive to the packet not being a known exploit AND the packet not being addressed to a broadcast IP address of a network AND the packet not being network administration traffic AND the packet not having a protocol listed in a list of protocols previously determined to be harmless network broadcast traffic, to determine and report that the packet is a new, exploit candidate.	
10. A computer program product for automatically determining if a packet is a new, exploit candidate, the computer program product comprising:	

Asserted Claims	Identification of Exception to Eligibility
a computer-readable tangible storage device; first program instructions to determine if the packet is a known exploit; second program instructions to determine if the packet is addressed to a broadcast IP address of a network; third program instructions to determine if the packet is network administration traffic; fourth program instructions, responsive to the packet being a known exploit OR the packet being addressed to a broadcast IP address of a network OR the packet being network administration traffic, to determine that the packet is not a new, exploit candidate; and fifth program instructions, responsive to the packet not being a known exploit AND the packet not being addressed to a broadcast IP address of a network AND the packet not being network administration traffic AND the packet not being another type of traffic known to be benign, to determine and report that the packet is a new, exploit candidate; and wherein the first, second, third, fourth and fifth program instructions are stored on the computer-readable tangible storage device; and the third program instructions determine if the packet is network administration traffic	

Asserted Claims	Identification of Exception to Eligibility
by comparing an IP protocol and IP address of the packet to a list of combinations of IP protocols and IP addresses previously determined to be network administration traffic.	
12. The computer program product of claim 1 further comprising sixth program instructions, responsive to the packet not being a known exploit AND the packet not being network broadcast traffic AND the packet not being addressed to a broadcast IP address of a network AND the packet not being another type of traffic known to be benign, to identify a sequence of packets including the first said packet, the sequence of packets being a new, exploit candidate; and wherein the sixth program instructions are stored on the computer-readable tangible storage device.	
13. A computer program product for automatically determining if a packet is a new, exploit candidate, the computer program product comprising: a computer-readable tangible storage device; first program instructions to determine if the packet is a known exploit; second program instructions to determine if the packet is addressed to a broadcast IP address of a network;	

Asserted Claims	Identification of Exception to Eligibility
third program instructions to determine if the packet has a protocol listed in a list of protocols previously determined to be harmless broadcast traffic; fourth program instructions to determine if the packet is network administration traffic; fifth program instructions, responsive to the packet being a known exploit OR the packet being addressed to a broadcast IP address of a network OR the packet being network administration traffic OR the packet having a protocol listed in a list of protocols previously determined to be harmless broadcast traffic, to determine that the packet is not a new, exploit candidate; and sixth program instructions, responsive to the packet not being a known exploit AND the packet not being addressed to a broadcast IP address of a network AND the packet not being network administration traffic AND the packet not having a protocol listed in a list of protocols previously determined to be harmless broadcast traffic, to determine and report that the packet is a new, exploit candidate; and wherein the first, second, third, fourth, fifth and sixth program instructions are stored on the computer-readable tangible storage device.	

Asserted Claims	Identification of Exception to Eligibility
14. The computer program product of claim 13 further comprising: seventh program instructions to determine if the packet is web crawler traffic; and wherein the fifth program instructions are responsive to the packet being a known exploit OR the packet being addressed to a broadcast IP address of a network OR the packet being network administration traffic OR the packet being web crawler traffic OR the packet having a protocol listed in a list of protocols previously determined to be harmless broadcast traffic, to determine that the packet is not a new, exploit candidate; and the sixth program instructions are responsive to the packet not being a known exploit AND the packet not being addressed to a broadcast IP address of a network AND the packet not being network administration traffic AND the packet not being web crawler traffic AND the packet not being other traffic known to be benign AND the packet not having a protocol listed in a list of protocols previously determined to be harmless broadcast traffic, to determine that the packet is a new, exploit candidate; and the seventh program instructions are stored on the computer-readable tangible storage device.	

Asserted Claims	Identification of Exception to Eligibility
15. The computer program product of claim 13 further comprising: seventh program instructions, responsive to the sixth program instructions determining that the packet is a new, exploit candidate, to determine a signature of the packet or a sequence of packets including the first the packet, and report the new, exploit candidate and the signature to an administrator; and wherein the seventh program instructions are stored on the computer-readable tangible storage device.	
16. The computer program product of claim 13 wherein the second program instructions determine if the packet is addressed to a broadcast IP address of the network by comparing a destination IP address of the packet to a gateway IP address of the network and a netmask of the network which identifies a broadcast IP address of the network.	
17. A computer system for automatically determining if a packet is a new, exploit candidate, the computer system comprising: one or more processors, one or more computer-readable memories, one or more computer-readable tangible storage devices, and program instructions stored on at least one of the one or more storage devices for execution	

Asserted Claims	Identification of Exception to Eligibility
by at least one of the one or more processors via at least one of the one or more memories, the program instructions comprising: first program instructions to determine if the packet is a known exploit; second program instructions to determine if the packet is addressed to a broadcast IP address of a network; third program instructions to determine if the packet is network administration traffic; fourth program instructions, responsive to the packet being a known exploit OR the packet being addressed to a broadcast IP address of a network OR the packet being network administration traffic, to determine that the packet is not a new, exploit candidate; and fifth program instructions, responsive to the packet not being a known exploit AND the packet not being addressed to a broadcast IP address of a network AND the packet not being network administration traffic AND the packet not being another type of traffic known to be benign, to determine and report that the packet is a new, exploit candidate.	
18. The computer system of claim 17 further comprising: sixth program instructions, stored on at least one of the one or more storage devices	

Asserted Claims	Identification of Exception to Eligibility
for execution by at least one of the one or more processors via at least one of the one or more memories, to determine if the packet is web crawler traffic; and wherein the fourth program instructions are responsive to the packet being a known exploit OR the packet being addressed to a broadcast IP address of a network OR the packet being network administration traffic OR the packet being web crawler traffic, to determine that the packet is not a new, exploit candidate; and the fifth program instructions are responsive to the packet not being a known exploit AND the packet not being addressed to a broadcast IP address of a network AND the packet not being network administration traffic AND the packet not being web crawler traffic, to determine that the packet is a new, exploit candidate.	
19. The computer system of claim 17 wherein the packet was received by a honeypot computing device at an unused IP address, and the first, second, third, fourth and fifth program instructions are executed at the honeypot computing device.	
20. The computer system of claim 17 further comprising: sixth program instructions, stored on at least one of the one or more storage devices	

Asserted Claims	Identification of Exception to Eligibility
for execution by at least one of the one or more processors via at least one of the one or more memories, responsive to the fifth program instructions determining that the packet is a new exploit candidate, to determine a signature of the packet, and report the new exploit candidate and the signature to an administrator.	

b. Chart identifying a description of the industry at the relevant time and how each element of each challenged claim, individually and in combination with the other elements, was well understood, routine, and conventional.

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
1. A computer program product for automatically determining if a packet is a new, exploit candidate, the computer program product comprising: a computer-readable tangible storage device; first program instructions to determine if the packet is a known exploit; second program instructions to determine if the packet is addressed to a broadcast IP address of a network; third program instructions to determine if the packet is network administration traffic; fourth program instructions, responsive to the packet being a known exploit OR the packet being addressed to a broadcast IP address of a network OR the packet being network administration	Claims are not "directed to" an improvement in computer technology if the technical benefits flow solely from performing an abstract idea in conjunction with well-understood structure, as the patent readily admits here. *BSG Tech LLC v. Buyseasons, Inc.*, 899 F.3d 1281, 1288 (Fed. Cir. 2018); *Elec. Power Grp., LLC v. Alstom S.A.*, 830 F.3d 1350, 1353-1354 (Fed. Cir. 2016) ("we have treated collecting information, including when limited to particular content …, as within the realm of abstract ideas … [a]nd we have recognized that merely presenting the results of abstract processes of collecting and analyzing information … is abstract as an ancillary part of such collection and analysis."); *Content Extraction & Transmission LLC v. Wells Fargo Bank, National Ass'n*, 776 F.3d 1343, 1345, 1358-1359 (Fed.Cir.2014) (finding claims directed to the abstract idea of collecting data from hard-copy documents, recognizing certain information within the collected data, and storing that information in memory were ineligible, despite the claims requiring conventional scanning and processing technology). The Asserted Claims of the '356 Patent fail *Alice* step two. Other than mental steps of "determin[ing]" conditions and classifying packets, the independent claims recite a generic "computer program product" with "a computer-readable tangible storage device" and generic "program instructions" (claims 1, 9– 11, 13) or a "computer system" with generic "processors," "memories," and "storage devices" (claim 17). These elements, which "merely require generic computer

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
traffic, to determine that the packet is not a new, exploit candidate; and fifth program instructions, responsive to the packet not being a known exploit AND the packet not being addressed to a broadcast IP address of a network AND the packet not being network administration traffic AND the packet not being another type of traffic known to be benign, to determine and report that the packet is a new, exploit candidate; and wherein the first, second, third, fourth and fifth program instructions are stored on the computer-readable tangible storage device.	implementation, fail to transform th[e] abstract idea." *Alice*, 573 U.S. at 221; *see id*. at 226 (elements for "a computer system and a computer-readable medium fail for substantially the same reasons," as do "data processing system" and "data storage unit"). As for the "determin[ing]" steps (claims 1, 9–11, 13, 17), they are in the abstract realm and provide no inventive concept, "no matter how groundbreaking" (*SAP*, 898 F.3d at 1170). Regardless, the patent admits these steps were conventional—it was "known for a human analyst to analyze all of the packets . . . to determine their type and whether they represent a known or unknown computer attack." '356 Patent at 2:50–66. The Asserted Claims of the '356 Patent also do not include any "additional features" that embody an "inventive concept." *Alice*, 134 S. Ct. at 2357. Indeed, the Federal Circuit has held that merely selecting information, by content, for collection, analysis, and presentation "does nothing significant to differentiate a process from ordinary mental processes, whose implicit exclusion from § 101 undergirds the information-based category of abstract ideas." *Elec. Power Grp.*, 830 F.3d at 1355. And, as discussed above, the specification of the '356 Patent readily admits that the arguably technical features of the claims are well-known and conventional, thus firmly establishing that the claims do not recite an inventive concept. *Intellectual Ventures I LLC v. Symantec Corp.*, 838 F.3d 1307, 1317 (Fed. Cir. 2016) (a patent's specification is "particularly useful in determining what is well-known or conventional"); *Intellectual Ventures I LLC v. Capital One Bank*, 792 F.3d 1363, 1367 (Fed. Cir. 2017) ("Nor, in addressing the second step of *Alice*, does claiming the improved speed or efficiency inherent with applying the abstract idea on a computer provide a sufficient inventive concept.") (citation omitted); *Berkheimer v. HP Inc.*, 881 F.3d 1360, 1371 (Fed. Cir. 2018) ("In a situation where the specification admits the additional claim elements are well-understood, routine, and conventional, it will be difficult, if not impossible, for a patentee to show a genuine dispute."). Therefore, the Asserted Claims of the '356 Patent are not directed to patent-eligible subject matter under 35 U.S.C. § 101 and are invalid. Defendant further incorporates by reference the above-identified Applicant Admitted Prior Art, wherein the '356 patent admit that the elements of the asserted claims were well-understood, routine, and conventional. Further, while prior art references are unnecessary to conclude that the asserted claims fail *Alice* step two, Defendant incorporates by reference each of its invalidity charts for the '356 patent (Exhibit P), which

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
	identify prior art teachings that further confirm that the elements of the asserted claims, individually or when combined, were well-understood, routine, and conventional in the industry.
2. The computer program product of claim 1 further comprising: sixth program instructions to determine if the packet is web crawler traffic; and wherein the fourth program instructions are responsive to the packet being a known exploit OR the packet being addressed to a broadcast IP address of a network OR the packet being network administration traffic OR the packet being web crawler traffic, to determine that the packet is not a new, exploit candidate; and the fifth program instructions are responsive to the packet not being a known exploit AND the packet not being addressed to a broadcast IP address of a network AND the packet not being network administration traffic AND the packet not being web crawler traffic, to determine that the packet is a new, exploit candidate; and the sixth program instructions are stored on the computer-readable tangible storage device.	The same is true for the elements in the dependent claims—e.g., checking for web crawler traffic, packet signatures, using a honeypot, and processing packet sequences, all of which were admittedly conventional. ’356 Patent at 2:53–55 (known to “determine which packets are . . . web crawler requests”); *id*. at 1:40–59, 2:21–37, 6:64–67 (known to search and compare packet signatures); *id*. at 2:38–39 (“A ‘honeypot’ is currently known to collect suspicious Internet message packets.”), *id*. at 4:28–32 (“honeypot” can be a conventional computer); *id*. at 7:36–38 (“There are currently known methods that can identify which TCP packets are part of the same sequence and to reassemble these packets into the sequence.”). Thus, the ’356 claims are all invalid. *See also* claim 1.
3. The computer program product of claim 1 wherein the first program instructions determine if the packet is a known exploit by searching	

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
the packet for a known signature of a known exploit.	
4. The computer program product of claim 1 wherein the first program instructions determine if the packet is a known exploit by comparing an identity of the packet to one or more identities, sent by an intrusion detection system, of respective packet(s) which the intrusion detection system determined to contain a known exploit.	
6. The computer program product of claim 1 further comprising: sixth program instructions, responsive to the fifth program instructions determining that the packet is a new exploit candidate, to determine a signature of the packet, and report the new exploit candidate and the signature to an administrator; and wherein the sixth program instructions are stored on the computer-readable tangible storage device.	
8. The computer program product of claim 1 wherein the second program instructions determine if the packet is addressed to a broadcast IP address of the network by comparing a destination IP address of the packet to a gateway IP address of the network and a netmask of the network	

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
which identifies a broadcast IP address of the network.	
9. A computer program product for automatically determining if a packet is a new, exploit candidate, the computer program product comprising: a computer-readable tangible storage device; first program instructions to determine if the packet is a known exploit; second program instructions to determine if the packet is addressed to a broadcast IP address of a network; third program instructions to determine if the packet is network administration traffic; fourth program instructions, responsive to the packet being a known exploit OR the packet being addressed to a broadcast IP address of a network OR the packet being network administration traffic, to determine that the packet is not a new, exploit candidate; and fifth program instructions, responsive to the packet not being a known exploit AND the packet not being addressed to a broadcast IP address of a network AND the packet not being network administration traffic AND the packet not being another type of traffic known to be benign, to determine and report that the packet is a	

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
new, exploit candidate; and wherein the first, second, third, fourth and fifth program instructions are stored on the computer-readable tangible storage device; the second program instructions also determine if the packet has a protocol listed in a list of protocols previously determined to be harmless network broadcast traffic; the fourth program instructions are responsive to the packet being a known exploit OR the packet being addressed to a broadcast IP address of a network OR the packet being network administration traffic OR the packet having a protocol listed in a list of protocols previously determined to be harmless network broadcast traffic, to determine that the packet is not a new, exploit candidate; and the fifth program instructions are responsive to the packet not being a known exploit AND the packet not being addressed to a broadcast IP address of a network AND the packet not being network administration traffic AND the packet not having a protocol listed in a list of protocols previously determined to be harmless network broadcast traffic, to determine and report that the packet is a new, exploit candidate.	

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
10. A computer program product for automatically determining if a packet is a new, exploit candidate, the computer program product comprising: a computer-readable tangible storage device; first program instructions to determine if the packet is a known exploit; second program instructions to determine if the packet is addressed to a broadcast IP address of a network; third program instructions to determine if the packet is network administration traffic; fourth program instructions, responsive to the packet being a known exploit OR the packet being addressed to a broadcast IP address of a network OR the packet being network administration traffic, to determine that the packet is not a new, exploit candidate; and fifth program instructions, responsive to the packet not being a known exploit AND the packet not being addressed to a broadcast IP address of a network AND the packet not being network administration traffic AND the packet not being another type of traffic known to be benign, to determine and report that the packet is a new, exploit candidate; and wherein	

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
the first, second, third, fourth and fifth program instructions are stored on the computer-readable tangible storage device; and the third program instructions determine if the packet is network administration traffic by comparing an IP protocol and IP address of the packet to a list of combinations of IP protocols and IP addresses previously determined to be network administration traffic.	
12. The computer program product of claim 1 further comprising sixth program instructions, responsive to the packet not being a known exploit AND the packet not being network broadcast traffic AND the packet not being addressed to a broadcast IP address of a network AND the packet not being another type of traffic known to be benign, to identify a sequence of packets including the first said packet, the sequence of packets being a new, exploit candidate; and wherein the sixth program instructions are stored on the computer-readable tangible storage device.	
13. A computer program product for automatically determining if a packet is a new, exploit candidate, the computer program product comprising:	

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
a computer-readable tangible storage device; first program instructions to determine if the packet is a known exploit; second program instructions to determine if the packet is addressed to a broadcast IP address of a network; third program instructions to determine if the packet has a protocol listed in a list of protocols previously determined to be harmless broadcast traffic; fourth program instructions to determine if the packet is network administration traffic; fifth program instructions, responsive to the packet being a known exploit OR the packet being addressed to a broadcast IP address of a network OR the packet being network administration traffic OR the packet having a protocol listed in a list of protocols previously determined to be harmless broadcast traffic, to determine that the packet is not a new, exploit candidate; and sixth program instructions, responsive to the packet not being a known exploit AND the packet not being addressed to a broadcast IP address of a network AND the packet not being network administration traffic AND the packet not having a protocol listed in a list of protocols previously determined to be harmless	

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
broadcast traffic, to determine and report that the packet is a new, exploit candidate; and wherein the first, second, third, fourth, fifth and sixth program instructions are stored on the computer-readable tangible storage device.	
14. The computer program product of claim 13 further comprising: seventh program instructions to determine if the packet is web crawler traffic; and wherein the fifth program instructions are responsive to the packet being a known exploit OR the packet being addressed to a broadcast IP address of a network OR the packet being network administration traffic OR the packet being web crawler traffic OR the packet having a protocol listed in a list of protocols previously determined to be harmless broadcast traffic, to determine that the packet is not a new, exploit candidate; and the sixth program instructions are responsive to the packet not being a known exploit AND the packet not being addressed to a broadcast IP address of a network AND the packet not being network administration traffic AND the packet not being web crawler traffic AND the packet not being other traffic known to be benign AND the packet not having a protocol listed in a list of protocols	

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
previously determined to be harmless broadcast traffic, to determine that the packet is a new, exploit candidate; and the seventh program instructions are stored on the computer-readable tangible storage device.	
15. The computer program product of claim 13 further comprising: seventh program instructions, responsive to the sixth program instructions determining that the packet is a new, exploit candidate, to determine a signature of the packet or a sequence of packets including the first the packet, and report the new, exploit candidate and the signature to an administrator; and wherein the seventh program instructions are stored on the computer-readable tangible storage device.	
16. The computer program product of claim 13 wherein the second program instructions determine if the packet is addressed to a broadcast IP address of the network by comparing a destination IP address of the packet to a gateway IP address of the network and a netmask of the network which identifies a broadcast IP address of the network.	
17. A computer system for automatically determining if a packet is a new, exploit	

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
candidate, the computer system comprising: one or more processors, one or more computer-readable memories, one or more computer-readable tangible storage devices, and program instructions stored on at least one of the one or more storage devices for execution by at least one of the one or more processors via at least one of the one or more memories, the program instructions comprising: first program instructions to determine if the packet is a known exploit; second program instructions to determine if the packet is addressed to a broadcast IP address of a network; third program instructions to determine if the packet is network administration traffic; fourth program instructions, responsive to the packet being a known exploit OR the packet being addressed to a broadcast IP address of a network OR the packet being network administration traffic, to determine that the packet is not a new, exploit candidate; and fifth program instructions, responsive to the packet not being a known exploit AND the packet not being addressed to a broadcast IP address of a network AND the packet not being network administration traffic AND the packet not being another	

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
type of traffic known to be benign, to determine and report that the packet is a new, exploit candidate.	
18. The computer system of claim 17 further comprising: sixth program instructions, stored on at least one of the one or more storage devices for execution by at least one of the one or more processors via at least one of the one or more memories, to determine if the packet is web crawler traffic; and wherein the fourth program instructions are responsive to the packet being a known exploit OR the packet being addressed to a broadcast IP address of a network OR the packet being network administration traffic OR the packet being web crawler traffic, to determine that the packet is not a new, exploit candidate; and the fifth program instructions are responsive to the packet not being a known exploit AND the packet not being addressed to a broadcast IP address of a network AND the packet not being network administration traffic AND the packet not being web crawler traffic, to determine that the packet is a new, exploit candidate.	
19. The computer system of claim 17 wherein the packet was received by a honeypot computing device at	

Asserted Claims	Identification of Relevant Industry and How Each Element Was Well Understood, Routine, and Conventional
an unused IP address, and the first, second, third, fourth and fifth program instructions are executed at the honeypot computing device.	
20. The computer system of claim 17 further comprising: sixth program instructions, stored on at least one of the one or more storage devices for execution by at least one of the one or more processors via at least one of the one or more memories, responsive to the fifth program instructions determining that the packet is a new exploit candidate, to determine a signature of the packet, and report the new exploit candidate and the signature to an administrator.	

Defendant incorporates its Motion for Partial Judgment on the Pleadings in Case No. 2:22-cv-00314 and 2:22-MD-03042-JRG, including all related briefing and oral argument, showing the factual and legal basis that support that the challenged '356 patent claims are patent-ineligible under 35 U.S.C. § 101. Defendant further incorporates by reference the above-identified Applicant Admitted Prior Art, wherein the '356 patent admit that certain technologies related to the elements of the asserted claims were well-understood, routine, and conventional. Defendant also incorporates each of its invalidity charts for the '356 patent (Exhibit P), which may additionally identify, among others, how the elements of the asserted claims, individually or when combined, were well-understood, routine, and conventional in the industry at the relevant time frame.[36]

[36] Defendants refer to its exemplary identification of motivations to combine and its invalidity charts solely for the purpose of complying with the Court's standing order. Defendant's reference to its exemplary identification of motivation to combine or its invalidity charts is not intended to

VII. DOCUMENT PRODUCTION ACCOMPANYING INVALIDITY AND SUBJECT MATTER ELIGIBILITY CONTENTIONS

Defendant will produce or make available for inspection documents required by Patent L.R. 3-4 according to the parties' agreed-upon schedule. These disclosures are based on Defendant's current understanding of Plaintiff's infringement allegations. This disclosure is made without the benefit of discovery. Defendant reserves the right to amend or supplement these disclosures and documents as this case proceeds and its investigation continues. Defendant further reserves the right to supplement these disclosures following the receipt of more fulsome and rules-compliant infringement contentions from Plaintiff, or as appropriate in response to the Court's claim construction or determination of other issues.

1. A copy or sample of the prior art identified pursuant to Patent L.R. 3-3(a) which does not appear in the file history of the patent(s) at issue.

In addition to the prior art the defendants in the MDL Litigation have produced (DEFS-TAASERA-PA_0000001 – DEFS-TAASERA-PA_0037275), Defendant is producing a copy of, or making available for inspection on a secured source code computer, the prior art identified herein. *See* PAN_TL_PA _0000001 – PAN_TL_PA_0004182 (produced concurrently herewith).

2. Documents supporting Defendant's Subject Matter Ineligibility Contentions

Pursuant to the Court's Standing Order Regarding Subject Matter Eligibility Contentions Applicable to All Patent Infringement Cases Assigned to Chief District Judge Rodney Gilstrap, Defendant is producing the materials on which Defendant seeks to rely for their contention that certain Asserted Claims fail to satisfy the requirements of 35 U.S.C. § 101 due to their failure to claim patentable subject matter. *See* Sections III.A-C, IV-VI, VII.1-2; *see also* DEFS-TAASERA-

be, and shall not be construed as, an admission that an analysis under 35 U.S.C. 102 (anticipation) or 103 (obviousness) is relevant or necessary to determine the patent eligibility of the asserted claims.

PA_0000001 – DEFS-TAASERA-PA_0037275, PAN_TL_PA_0000001 – PAN_TL_PA_0004182 (produced concurrently herewith).

Respectfully submitted,

Dated: April 29, 2023

By: */s/ Michael R. Rueckheim*

Kelly C. Hunsaker
KHunsaker@winston.com
Michael R. Rueckheim
(Texas State Bar No. 24081129)
MRueckheim@winston.com
Eimeric Reig-Plessis
EReigPlessis@winston.com
WINSTON & STRAWN LLP
225 Shoreline Drive, Suite 520
Redwood City, CA 94065
Telephone: (650) 858-6500
Facsimile: (650) 858-6550

Melissa Richards Smith
(Texas State Bar No. 24001351)
melissa@gillamsmithlaw.com
GILLAM & SMITH, LLP
303 South Washington Avenue
Marshall, TX 75670
Telephone: 903-934-8450
Facsimile: 903-934-9257

ATTORNEYS FOR DEFENDANT,
PALO ALTO NETWORKS, INC.

CERTIFICATE OF SERVICE

I hereby certify that all counsel of record who have appeared in this case are being served with a copy of the foregoing via electronic mail on April 29, 2023.

/s/ Michael R. Rueckheim
Michael R. Rueckheim