UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

CROWDSTRIKE, INC.

Petitioner

v.

OPEN TEXT INC.

Patent Owner

Case IPR2023-00556

Patent No. 8,726,389

DECLARATION OF PROFESSOR NENAD MEDVIDOVIC, Ph.D.
IN SUPPORT OF PATENT OWNER'S PRELIMINARY RESPONSE

TABLE OF CONTENTS

I. QUALIFICATIONS 2

II. UNDERSTANDING OF RELEVANT LEGAL PRINCIPLES 7

III. STATE OF THE PRIOR ART 9

IV. The '389 Patent 11

A. Summary Of The '389 Patent 11

B. Claim Construction 17

C. Level Of Ordinary Skill In The Art 19

V. OVERVIEW OF PRIOR ART 20

A. Overview Of Kester 20

B. Overview Of Kennedy 32

VI. KESTER IN VIEW OF Kennedy and Honig DOES NOT DISCLOSE OR RENDER OBVIOUS THE CHALLENGED CLAIMS 36

A. Kester Is Non-Analogous To The '389 Patent 36

B. Independent Claims 1 and 15 42

1. Kester and Kennedy Do Not Disclose Or Render Obvious Limitations [1.5] and [15.4]. 44

2. POSITA Would Not Be Motivated To Modify Combine Kester With Kennedy's Automated Classification Heuristics Because Kester's Human Administrator Component Is Incompatible With Kennedy's Alleged Automation 47

3. A POSITA Would Not Expect To Succeed Modifying Kester With Kennedy's Automated Classification Heuristics Because Kester Embodies the Problem Addressed By The '389 Patent. 53

C. Dependent Claims 55

VII. CLAIMS AND/OR LIMITATIONS NOT ADDRESSED IN THIS DECLARATION 55

VIII. SECONDARY OBJECTIVE INDICIA OF NONOBVIOUSNESS 56

IX. CONCLUSION.....57

EXHIBIT LIST

Exhibit	Description
2001	Declaration of Professor Nenad Medvidovic, Ph.D
2002	*Curriculum Vitae* of Professor Nenad Medvidovic, Ph.D.
2003	*Webroot, Inc., Open Text Inc., v. AO Kaspersky Lab et al.*, Case No. 6:22-cv-00243-ADA-DTG, Order Granting Third Amended Scheduling Order, Dkt. 142 (W.D. Tex., Dec. 27, 2022)
2004	*Sonrai Memory Ltd. v. Kingston Tech. Co. and Kingston Tech. Corp.,* Case No. 6:21-cv-1284-ADA, Order Denying Kingston Tech. Co. and Kingston Tech. Corp.'s Opposed Motion to Stay Pending Resolution of *Inter Partes* Review, Dkt. 94 (W.D. Tex., Oct. 18, 2022)
2005	*mCom IP, LLC v. Cisco Systems, Inc.,* Case No. 6:22-cv-00261-ADA, Order Denying Defendant's Motion to Stay Pending *Inter Partes* Review, Dkt. 42 (W.D. Tex., Oct. 20, 2022)
2006	*Intellectual Ventures I LLC et al. v. Hewlett Packard Enterprise Co.*, No. 6:21-cv-00596-ADA, Order Denying Hewlett Packard Enterprise Company's Opposed Motion to Stay Pending Resolution of *Inter Partes* Review, Dkt. 104 (W.D. Tex., Dec. 22, 2022)
2007	*Webroot, Inc. and Open Text Inc., v. AO Kaspersky Lab. et al.,* Case No. 6:22-cv-243-ADA-DTG, Dec. 9, 2022 Transcript of Motions Hearing, Dkt. 133 (W.D. Tex. Dec. 15, 2022) (excerpted)
2008	Dani Kass, "Catching Up On Patent Litigation With Judge Albright" (Law360 Mar. 14, 2023); https://www.law360.com/pulse/articles/1582438/print?section=pulse/courts
2009	U.S. District Courts—Federal Court Management Statistics–Profiles—During the 12-Month Periods Ending March 31, 2018 Through 2023, "Table N/A—U.S. District Courts–Combined Civil and Criminal Federal Court Management Statistics (March 31,

Exhibit	Description
	2023)", *available* https://www.uscourts.gov/sites/default/files/data_tables/fcms_na_distprofile0331.2023.pdf
2010	*Webroot, Inc. and Open Text Inc., v. AO Kaspersky Lab. et al.,* Case No. 6:22-cv-243-ADA-DTG, Claim Construction Order, Dkt. 236 (W.D. Tex. Mar. 16, 2023)
2011	*Webroot, Inc. and Open Text Inc., v. CrowdStrike, Inc. and CrowdStrike Holdings, Inc.,* Case No. 6:22-cv-00241-ADA-DTG, First Amended Complaint for Patent Infringement, Dkt. 60 (W.D. Tex., Sept. 8, 2022) (excerpted)
2012	*Webroot, Inc. and Open Text Inc., v. AO Kaspersky Lab. et al.,* Case No. 6:22-cv-243-ADA-DTG, Scheduling Order, Dkt. 39 (W.D. Tex. Aug. 17, 2022)
2013	*Webroot, Inc. and Open Text Inc., v. ForcePoint LLC*, Case No. 6:22-cv-00342-ADA-DTG, Defendant ForcePoint LLC's Preliminary Invalidity Contentions (W.D. Tex., Sept. 13, 2022) (excerpted)
2014	*Webroot, Inc. and Open Text Inc., v. Sophos Ltd.,* Case No. 6:22-cv-00240-ADA-DTG, Defendant Sophos Ltd.'s Preliminary Invalidity Contentions (W.D. Tex., Sept. 13, 2022) (excerpted)
2015	*Webroot, Inc. and Open Text Inc., v. Trend Micro, Inc.,* Case No. 6:22-cv-00239-ADA-DTG, Defendant's Preliminary Invalidity Contentions (W.D. Tex., Sept. 13, 2022) (excerpted)
2016	*Webroot, Inc. and Open Text Inc., v. AO Kaspersky Lab*, Case No. 6:22-cv-00243-ADA-DTG, Defendant's Preliminary Invalidity Contentions Appendix A (W.D. Tex., Sept. 14, 2022) (excerpted)
2017	*Webroot, Inc. and Open Text Inc., v. CrowdStrike, Inc. and CrowdStrike Holdings, Inc.*, Case No. 6:22-cv-00241-ADA-DTG, Defendant's Preliminary Invalidity Contentions (W.D. Tex., Sept. 13, 2022) (excerpted)

Exhibit	Description
2018	*Ex Parte* James M. Brennan, et al., Appeal 2021-003606, Application 15/285,875, Decision On Appeal Statement of the Case (PTAB Jul. 26, 2022)
2019	File History of U.S. Patent No. 8,418,250 ("'250 File History")

I, Professor Nenad Medvidovic, Ph.D., declare as follows:

1. I have been engaged by King & Spalding LLP on behalf of OpenText Inc. ("Patent Owner"), to provide this Declaration concerning the technical subject matter relevant to the petition for Inter Partes Review ("Petition") of U.S. Patent 8,726,389 ("the '389 patent").

2. I am over 18 years of age. I have personal knowledge of the facts stated in this Declaration and could testify competently to them if asked to do so.

3. I have reviewed and am familiar with the specification and the claims of the '389 patent. In general, I will cite to the specification of a United States patent using the following formats: (Exhibit No. Col:Line Number(s)) or (Exhibit No. Paragraph Number(s)). For example, the citation (EX-1001 1:1–10) points to the '389 patent specification at column 1, lines 1–10. Also, for convenience, I use italics to denote limitations from the challenged claims.

4. All the opinions in this Declaration are based on the documents I reviewed and my knowledge and professional judgment. In forming the opinions expressed in this Declaration, I reviewed the Petition for *Inter Partes* Review of U.S. Patent No. 8,726,389 and supporting exhibits.

5. I have been asked to provide my technical opinions regarding how a person of ordinary skill in the art would have understood the claims of the '389 patent at the time of the alleged invention. I have also been asked to provide my

technical opinions on how concepts in the '389 patent specification relate to claim limitations of the '389 patent. In reaching the opinions provided herein, I have considered the '389 patent, its prosecution history, and the references cited above and have drawn as appropriate on my own education, training, research, knowledge, and personal and professional experience.

I. QUALIFICATIONS

6. In formulating my opinions, I have relied on my knowledge, training, and experience in the relevant filed, which I will summarize briefly. A more detailed summary of my background, experience, and publications is set forth in my curriculum vitae (CV), which is submitted as Exhibit 2002.

7. I received a Bachelor of Science degree, Summa Cum Laude, from Arizona State University's Computer Science and Engineering department.

8. I received a Master of Science degree from the University of California at Irvine's Information and Computer Science department.

9. I received a Doctor of Philosophy degree from the University of California at Irvine's Information and Computer Science department. My dissertation was entitled "Architecture-Based Specification-Time Software Evolution."

10. For more than 25 years, my research has focused on the building blocks that comprise software systems: software components (computational and data

elements) and connectors (interaction-facilitating elements), and mechanisms for their implementation, primarily middleware facilities and software frameworks. In particular, my Ph.D. dissertation—whose first research results were published in 1995 while the dissertation was completed in 1999—yielded a new design style for the development and evolution of highly distributed software systems built out of independent components that exchange different types of information encoded in messages in order to accomplish specific goals. This research focused on the design principles behind such systems, the appropriate support for modeling the systems (both at the level of individual agents as well as their integrations into subsystems and entire systems), and the infrastructure for their implementation, deployment, and optimization. This work was applied across a wide range of domains, from traditional distributed systems to mobile devices and autonomous robots.

11. I am employed by the University of Southern California ("USC") as a faculty member in the Computer Science Department and have been since January 1999. I currently hold the title of Professor with tenure. Since July 2022, I have served as my department's Chairman. Between January 2009 and January 2013, I served as the Director of the Center for Systems and Software Engineering at USC. Between July 2011 and July 2015, I served as my Department's Associate Chair for Ph.D. Affairs. Between July 2021 and June 2022, I served as my Department's Vice-Chair for Tenure Affairs.

12. I teach graduate and undergraduate courses in Software Architecture, Software Engineering, and Embedded Systems, and I advise Ph.D. students. I have graduated 19 Ph.D. students and advise 7 students currently pursuing a Ph.D.

13. I served as Program Co-Chair for the flagship conference in my field—International Conference on Software Engineering ("ICSE")—held in May 2011. I have served as Chair or Co-Chair for various other conferences in the Software Engineering field, including: the Fifth Working Institute of Electrical and Electronics Engineers ("IEEE")/IFIP Conference on Software Architecture, the Third IEEE International Conference on Self-Adaptive and Self-Organizing Systems, the Fifteenth International Association for Computing Machinery's ("ACM") Special Interest Group on Software Engineering ("SIGSOFT") Symposium on Component Based Software Engineering, the IEEE/CSSE/ISE Workshop on Software Architecture Challenges for the 21st Century, and the Doctoral Symposia at the Sixteenth ACM SIGSOFT International Symposium on the Foundations of Software Engineering and at the 44th International Conference on Software Engineering.

14. I serve or have served as an editor of several peer-reviewed journals, including: "IEEE Transactions on Software Engineering," "ACM Transactions on Software Engineering and Methodology," "Journal of Software Engineering for Robotics," "Elsevier Information and Software Technology Journal," "Journal of

Systems and Software," "Journal of Software Engineering Research and Development," and "Springer Computing Journal." Additionally, I have served as a guest editor of several special issues for different journals. Since January 2018, I have served as Editor-in-Chief of "IEEE Transactions on Software Engineering."

15. Between September 2013 and September 2015, I served as Chair of the ICSE Steering Committee. Both prior to that (2008-2013) and subsequent to that (2015-present) I have been a member of the ICSE Steering committee. I am currently a member of the Steering Committee of the joint European Conference on Software Engineering and the Symposium on the Foundations of Software Engineering ("ESEC/FSE"). From July 2015 to September 2017, I chaired the Steering Committee of FSE, while it was a standalone symposium. I previously served as a member of the Steering Committees of the Working IEEE/IFIP Conference on Software Architecture.

16. Between July 2015 and June 2018, I served as Chair of the ACM SIGSOFT, the largest professional organization in my field of work. During the past year, I have chaired the Committee in charge of ACM's Software System Award, one of the most important technical awards given in my field.

17. I co-authored "Software Architecture: Foundations, Theory, and Practice," a widely used textbook in the field of Software Systems' Architecture.

18. I have served as editor of various books in the Software Engineering field including: "Proceedings of the 3rd International Conference on Self-Adaptive and Self-Organizing Systems," "Proceedings of the Warm-Up Workshop for the 32nd International Conference on Software Engineering," "Proceedings of the 5th Working IEEE/IFIP Conference on Software Architecture," and "40 Editions of ICSE – The Ruby Anniversary Celebration."

19. I have authored or co-authored nearly 300 papers in the Software Engineering field. My most cited paper has been cited well over 3,000 times. A paper I co-authored in the 1998 International Conference on Software Engineering, my field's flagship conference, was given ten years later, in 2008, that conference's Most Influential Paper Award. Another one of my publications received a Most Influential Paper Award in 2020. Several of my other publications have received "best paper" or "most cited" awards."

20. Several of my publications have focused on different aspects of computer security. A paper describing a security-vulnerability detection tool for Android devices, built in my lab, won the Best Tool Paper award at the 2017 Automated Software Engineering Conference. I have co-authored three patents in the area of cybersecurity.

21. I have been named a Distinguished Scientist of the ACM. I have been elected a Fellow of the IEEE, IEEE's highest grade that is granted to less than 0.1% of its membership annually.

II. UNDERSTANDING OF RELEVANT LEGAL PRINCIPLES

22. I am not a lawyer, and I will not provide any legal opinions. Although I am not a lawyer, I have been advised certain legal standards are to be applied by technical experts in forming opinions regarding the meaning and validity of patent claims.

23. I understand that a patent claim is invalid if it is anticipated or obvious in view of the prior art, and that a claim can be unpatentable even if all of the requirements of the claim cannot be found in a single prior-art reference. I further understand that invalidity of a claim requires that the claim be anticipated or obvious from the perspective of a person of ordinary skill in the art at the time the invention was made.

24. I have been informed that a patent claim is invalid if it would have been obvious to a person of ordinary skill in the art. In analyzing the obviousness of a claim, I understand the following factors may be taken into account: (1) the scope and content of the prior art; (2) the differences between the prior art and the claims; (3) the level of ordinary skill in the art; and (4) any so called "secondary considerations" of nonobviousness if they are present.

25. I have been informed that a claim can be obvious in light of a single prior art reference or multiple prior art references. I understand that to prove that prior art or a combination of prior art renders a patent obvious, it is necessary to:

1) Identify the particular references that, singly or in combination, make the patent obvious;
2) Specifically identify which elements of the patent claim appear in each of the asserted references; and
3) Explain why a person of ordinary skill in the art would have combined the references, and how they would have done so, to create the inventions claimed in the patent. I further understand that exemplary rationales that may support a conclusion of obviousness include:
    - Combining prior art elements according to known methods to yield predictable results;
    - Simple substitution of known element for another to obtain predictable results;
    - Use of known technique(s) to improve similar devices (methods or products) in the same way;
    - Applying a known technique to a known device (method or product) ready for improvement to yield predictable results;
    - "Obvious to try" – choosing from a finite number of identified, predictable solutions with a reasonable expectation of success;
    - Known work in one field of endeavor may prompt variations of the work for use in either the same field or a different field based on design

incentives or other market forces if the variations are predicable to a person of ordinary skill in the art; and

- Some teaching, suggestion, or motivation in the prior art that would have led a person of ordinary skill in the art to modify the prior art reference or to combine the prior art reference teachings to arrive at the claimed invention.

26. I have been informed that, in considering obviousness, hindsight reasoning derived from the patent-at-issue may not be used.

27. I have been informed that a prior art reference may be relied upon not only for what it expressly teaches, but also for what is inherent therein—i.e., what necessarily flows from the express teachings. I further understand that the mere fact that a certain thing may result from a given set of circumstances is not sufficient.

III. STATE OF THE PRIOR ART

28. "Malware" generally refers to an executable computer file or an object that is or contains malicious code, such as viruses, trojans, worms, and the like. (EX-1001 1:19-22.) Different versions of malware can function in different manners, and what works as a detection for one version of malware (*e.g.*, a virus) may be less effective against another version (*e.g.*, a worm). (*See generally* EX-1001 1:19-3:3.)

29. At the time of invention of the '389 patent, many malware systems focused on looking for previously known malware, *e.g.*, by examining objects (*i.e.*, any executable computer file) for known signatures indicative of malware. (*E.g.*,

EX-1001 1:23-34.) "Generally, the method of dealing with malware is that when new types of malware are released, for example via the Internet, these are eventually detected." (EX-1001 1:27-29.) Providers of anti-virus software push an update to their software intended to recognize and handle known malware. (EX-1001 1:29-33.) However, these types of systems suffered from inherent delay. (EX-1001 1:37-40.) A system would be vulnerable after release of a new piece of malware until such malware was detected and a solution generated and released to users. (*Id.*)

30. Other anti-malware systems sought to protect from previously unknown malware. For example, local area networks were known to implement a centralized computer responsible for authorizing the running of files on remote computers within the network. (EX-1001 1:65-2:2.) When a remote computer encountered a file that was unknown to it, it would forward a request to the central computer. (EX-1001 2:2-5.) When the file was unknown even to the central computer, the central computer would obtain the whole file from the remote computer in order to analyze it to determine whether it should be regarded as safe or malware. (EX-1001 2:9-13.) But such an analysis had to be "carried out manually or semi-manually" by subjecting the file to a detailed analysis. (EX-1001 2:14-17.)

31. These analysis methods included, for example, emulation or interpretation, "which can still take days given the human involvement that is typically required." (EX-1001 2:14-17.) Other prior art systems similarly relied on

a human system administrator to "make a better informed though still 'manual' decision as to whether or not the file is safe to run." (*E.g.*, EX-1001 2:24-33.) "There is therefore still a considerable period of time before a new file is classified as safe or as malware." (EX-1001 2:18-19.)

32. Thus, prior art systems "rel[ied] on deep analysis of a new object in order to determine whether or not the object is malicious, which introduces delay and therefore risk to users during the period that the file is analysed and new anti-malware signatures distributed[.]" (EX-1001 2:63-67.)

33. Accordingly, there was a long-felt need for an improved technique to efficiently determine whether objects were safe or malware.

IV. THE '389 PATENT

A. Summary Of The '389 Patent

34. The '389 Patent is entitled "Methods and Apparatus for Dealing with Malware" and describes and claims a method and system for classifying a computer "object" as malware. (*E.g.*, EX-1001 Claims 1, 15.) The '389 patent describes a computer "object" as including a computer file, partial file, sub-program, macro, web page, any piece of code to be operated by or on the computer, or any other event. (EX-1001 8:1-4.)

35. Unlike the methods described in the prior art, the '389 Patent describes a rapid determination of whether an object (*e.g.*, a file) should be classified as safe

or malware, without requiring a preliminary detailed analysis of the object. (EX-1001 3:14-24.)

36. The '389 patent teaches a method that may be implemented in a distributed network, such as a local area network, having a "central" or "base" computer connected to multiple "remote" computers. (EX-1001 7:54-59, Fig. 1.)

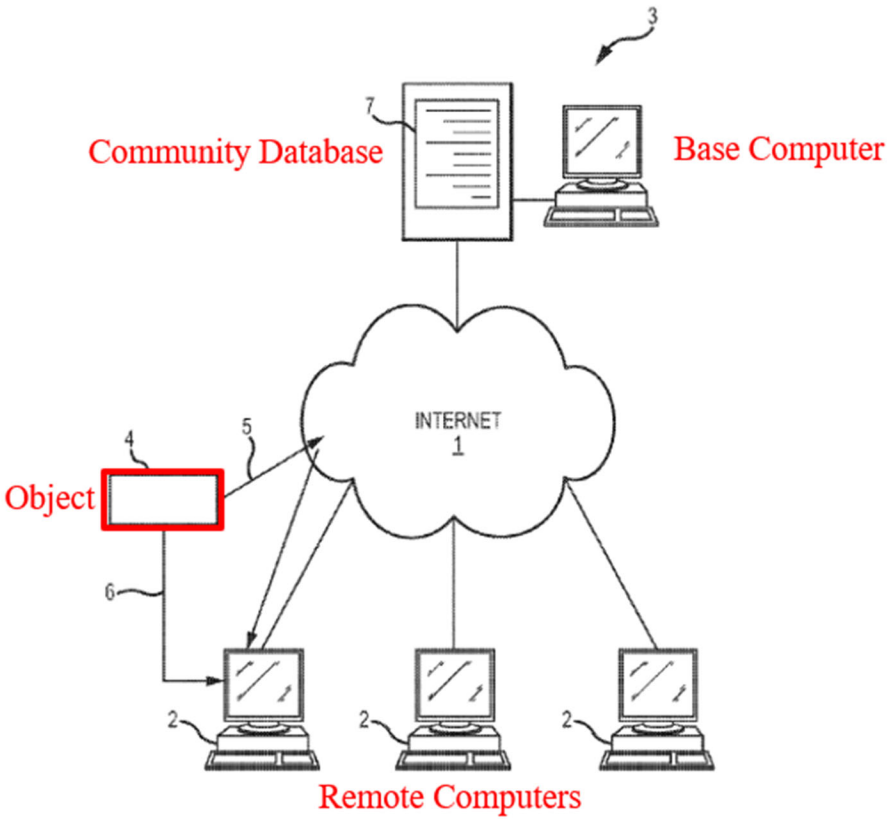

FIG.1

(EX-1001, Fig. 1, 7:59-8:20 (annotated).)

37. At a high level, the '389 patent claims a method (and system for implementing this method), in which the base computer receives information about a computer object from multiple remote computers and correlates the received data

about the object. (EX-1001 3:4-14, cls. 1, 15.) After correlating the data, the base computer then compares the correlated data to other objects or entities and, on the basis of that comparison, classifies the computer object as malware. *Id.*

38. More specifically, the '389 patent discloses a "community database" that "is populated, over time, with information relating to each object run on all of the connected remote computers." (EX-1001 8:12-14.) When an object is encountered at one of the remote computers, data related to the object is sent to the base computer, where it is stored in the community database for further analysis. (EX-1001 8:21-22, 8:47-50.) Such data may include executable instructions within the object; the size of the object; the current name of the object; the physical and folder location of the object; the original name of the object; the creation and modification dates of the object; vendor, product and version and any other information stored within the object; the object header or header held by the remote computer; and events initiated by or involving the object. (EX-1001 4:58-67.) The data stored in the community database provides "an extensive corollary of an object's creation, configuration, execution, behaviour, identities and relationships to other objects." (EX-1001 10:37-42.)

39. This object-specific data facilitates a rapid and more efficient determination of whether a particular object is malware. For example, one of the ways that this data may be provided to the base computer is in the form of a "key,"

or "signature" which allows for reduced transmission and storage requirements and faster analysis at the base computer. (EX-1001 3:48-56.) A key containing data that represents executable instructions of an object "allows a comparison to be made at the base computer of only the executable instructions of the object." (EX-1001 3:56-61.) "This means for example that differently named objects that basically have the same executable instructions, which is often an indicator that the objects are malware, can nevertheless be regarded as the 'same' object for this purpose." (EX-1001 3:61-65.)

40. As another example, a new version of a program may be released which has minor changes compared to a previous version that is already known by the base computer. (EX-1001 3:65-4:3.) In substance—at least with respect to their executable instructions—the two programs can be regarding as being the same, despite minor differences. (EX-1001 3:65-4:4.) "[T]his [is] useful in distinguishing between malware and for example revised versions of previous software…because the base computer can inform the remote computers that an apparently new object is for this purpose the same as a previously known object." (EX-1001 4:4-12.) That is, the object-specific data allows for a more rapid and efficient determination of whether that object is malware.

41. When the base computer receives data about a particular computer object from two or more computers, the base computer can correlate this object-

specific data. (EX-1001 cls. 1, 15, 10:43-46.) "Compared to the prior art that relies solely on signature matching, this aspect allows a comparison to be made between the objects and/or their effects on the different remote computers to determine whether or not a particular object should be classed as good or as malware." (EX-1001 3:14-18.)

42. The '389 patent teaches that this comparison preferably "identifies relationships between the object and other objects." (EX-1001 3:37-38.) "[A]n object is considered to have a relationship based on the event performed by it upon another object, or upon it by another object." (EX-1001 11:11-13.) In plain terms, a relationship may exist where (1) one object created or was created by the other object; (2) when one object has created a registry key to run or configure the other object; or (3) when one object was configured by or had registry keys created by the other object. (EX-1001 11:7-11.) Thus, the "other objects" may include the object-in-question, parent objects (*i.e.*, the object-in-question's creator), child objects (*i.e.*, objects the object-in-question has created), or other process-related objects. (EX-1001 3:43-47.)

43. This relationship can then be used to make an efficient determination as to whether a particular object should be classed as malware. (*See* EX-1001 cls. 1, 15, 3:38-43.) The '389 patent explains:

> In an example, this can be used to immediately and automatically to mark a child object as bad (or good) if the [] parent or other related

> object is bad (or good). Thus, if at least one other object to which said object is related is classed as malware, then the method may comprise classifying said object as malware.

(EX-1001 3:38-43; *see also id.* 11:5-60, 17:24-18:7.)

44. The '389 patent provides a real-world example of this method, involving the notepad.exe program, which is commonly installed on computers running the Windows operating system. (EX-1001 11:30-32.) In this case, the object is the notepad.exe program, which has a "limited range of functionality" (*e.g.*, writing and editing text). (EX-1001 11:32-35.) Thus, it would be unexpected to see an object like notepad.exe perform certain actions such as transmitting data to another computer or program. *Id.* Prior art systems might have simply flagged such behavior as suspicious. (*See* EX-1006 [0006] (describing the prior art's propensity for false alarms due to the dynamic behavior of users and programs).)

45. The '389 patent, however, is able to run a comparison with other objects and identify relationships which may provide context for the expanded capability. (*See* EX-1001 11:35-42.) For example, the object notepad.exe may be modified by injecting new code into it, *e.g.*, via a dynamic link library injection, which would modify the capabilities of notepad.exe. (EX-1001 11:35-38.) The '389 patent discloses using the data that defines the relationships between objects to deduce whether or not the new behavior of a program can be attributed to its association

with another object. (EX-1001 11:35-42.) The object can then be marked as malware or not, as appropriate. (EX-1001 11:42-44.)

46. The '389 patent's centralization of object-specific community data allows the base computer to correlate and compare the data and make a "rapid identification and determination" of "whether or not a particular object should be classed as good or as malware." (EX-1001 3:14-24, 11:55-59, cls. 1, 15.) This classification promotes future efficiency, because when the network encounters objects that have been classified in the past, the prior classification negates the need to perform time-consuming "deep analysis" of the object itself. (EX-1001 3:14-24, cl. 14.)

B. Claim Construction

47. Petitioner states that it relies on the plain and ordinary meaning of the claims. (Petition 6.) I understand that Petitioner and its co-defendants raised multiple terms for construction in a parallel proceeding in the Western District of Texas, and that the District Court issued a claim construction order on March 16, 2023, finding that no terms of the '389 patent were indefinite and that all terms should be construed to have their plain and ordinary meaning. The Court also provided several clarifying remarks, reproduced below. Consistent with the Court's order, I apply the plain and ordinary meaning of all claim terms.

Claims	Claim Term	Court's Construction
1, 15	"similar objects" / "similar computer objects"	Not indefinite. Plain-and-ordinary meaning. (EX-2010 3.)
14	"a mask for said computer object that defines acceptable behaviour for the computer object"	Plain-and-ordinary meaning. Note not for the jury: the plain-and-ordinary meaning of this term is not the inverse of a mask of unacceptable behavior (*Id.* 4.)
1-14, 26, 30	"events initiated [by] or involving the computer object"	Plain-and-ordinary meaning. Note: the plain-and-ordinary meaning of "events" includes but is not limited to "actions or behaviours of an object acting upon another object or some other entity" (*Id.* 5.)
1	"first remote computer" / "second remote computer"	Plain-and-ordinary meaning. (*Id.* 5.)
1	"correlating…at least a portion of the data about the computer object…comparing…the correlated data about the computer object"	Plain-and-ordinary meaning. The claim language recites that at least a portion of the data, but not all of the data, must be correlated before comparing. (*Id.* 6-7.)
1, 6, 8, 9, 12, 14, 30	"remote computer"	Plain-and-ordinary meaning. (*Id.* 8.)

C. Level Of Ordinary Skill In The Art

48. I understand that certain issues relating to the validity of the '389 patent must be judged from the perspective of a person of ordinary skill in the relevant art, as I discuss below. I have been asked to define the level of a "person having ordinary skill in the art" or "POSITA" that someone would have had at the time the alleged invention as claimed was made. In deciding the level of ordinary skill, I have considered the following factors, which I have been informed are relevant in determining the level of ordinary skill in the art:

- The levels of education and experience of persons working in the field;
- The types of problems encountered in the art;
- The prior art solutions to these problems;
- The rapidity with which innovations are made; and
- The sophistication of the technology.

49. A POSITA at the time of the invention of the '389 patent (which claims priority to an earlier application (GB 0513375.6) filed in Great Britain on June 30, 2005) would have had a Bachelor's degree in an accredited program of Electrical Engineering, Computer Engineering, or Computer Science or in a similar discipline, and have 2-3 years of practical work or research experience in the general fields of electrical engineering, computer science, networking, communications, and device

and network security. More advanced degrees and/or training in a related discipline can compensate for shorter work experience.

50. I understand that Petitioner has defined the level of ordinary skill in the art as:

> As of June 30, 2005, a person having ordinary skill in the art (PHOSITA) would be a person having a bachelor's degree in computer science, computer engineering, or an equivalent, as well as two years of industry experience and would have had a working knowledge of host monitoring systems, software security analysis, and dynamic malware analysis. Additional graduate education could subsitutte for professional experience, or significant experience in the field could substitute for formal education.

(Petition, 6-7.)

51. Despite minor variations in these two positions, these variations would not alter my opinions expressed in this Declaration.

52. Based on either standard, I was at least a person of ordinary skill in the art at the time of the invention of the '389 patent.

V. OVERVIEW OF PRIOR ART

A. Overview Of Kester

53. Kester teaches that its "invention is related to computing devices and, more particularly to monitoring and controlling application files operating thereon." (EX-1004 [0003].) Kester explains that the increase in Internet connectivity poses

risks to employers, because employees are able to download and install illegal or unlicensed applications themselves, run processes that monopolize network bandwidth, and/or expose the employer to risk to sensitive data. (EX-1004 [0005].) For example, instant messaging (IM) can increase productivity by facilitating file transfers and communication, but can also pose a risk to sensitive data. (*Id.*) As another example, "streaming media is a growing concern because of its drain on network bandwidth." (*Id.*)

54. While software was available to manage how employees access the Internet in the workplace, Kester sought to provide a solution capable of managing "the broader intersection of employees with their computing environments." (EX-1004 [0006].) Accordingly, Kester discloses a system and method for controlling both the launch of applications and network access requests. (*Id.* [0040]-[0041].)

55. Consistent with Kester's goal of controlling the "broader" intersection of employees with their computing environments, Kester teaches a method that is capable of controlling a wide variety of application categories. (EX-1004 Fig. 4A (listing 44 exemplary categories).) Figure 4A lists at least 44 exemplary categories of applications:

FIG. 4A

Parent group	Category	Security VC	Liability VC	Productivity VC
System				
	Operating Systems			
	Device Drivers			
	File Management			
	Installers			
	Miscellaneous Utilities			
Access/Privacy				
	Anti-virus Software			
	Authentication			
	Encryption			
	Firewalls			
	Hacking	X	X	
	Remote Access			
	Spyware	X		
	System Audit			
Productivity				
	Contact Managers			
	CRM			
	Data Warehouse, Analytics, Reporting			
	Database			
	Document Viewers			

	Document Viewers			
	ERP/SCM			
	Graphics			
	Infrastructure			
	Presentation			
	Project Managers			
	Proprietary			
	Reference			
	Search/Retrieval/Knowledge			
	Management			
	Software Development			
	Spreadsheets			
	Suite/integrated			
	Web/Desktop Publishing			
	Word Processing			
Communication				
	Collaboration			
	Dedicated Browsers			X
	Email			
	Instant Messaging			
	P2P Filesharing		X	
	Telephone/Conference/Fax			
	Web Browser			
Audio/Video				
	Media Players			X
	Imaging			X
Entertainment				
	Adult		X	
	Gambling		X	
	Games			X
Malware				
	Malicious Applets and Scripts			

(EX-1004 Fig. 4A, [0081]-[0082].) Consistent with Kester's acknowledgement of risks associated with communication-related applications, Kester discloses communication-related categories, such as "Instant Messaging," "Email," "P2P Filesharing" and others. (*Id.* Fig. 4A, [0005].) Consistent with Kester's acknowledgment of bandwidth problems with streaming applications, Kester discloses audio/video-related categories, such as "media players" and "imaging." (*Id.*)

56. Both Petitioner and Dr. Lee characterize Kester as a method for classifying a computer object as malware based on the computer object's behavior. (Petition 7-8; EX-1003 ¶56.) But it is important to understand that Kester is not directed to the ability to determine whether a process was safe or malware. Rather, as I have explained above, Kester's goal was to provide employers with control across a large variety of applications. (EX-1004 [0006], Fig. 4A.)

57. Malware is merely one of the seven exemplary parent groups of categories, and contains a single exemplary category of "malicious applets and scripts." (EX-1004 Fig. 4A.) This is the sole mention of malware in Kester, which contains no further description of what should be categorized as malicious and/or what policy should be implemented for this category. (EX-1004 Fig. 4A, [0082].) The majority of categories depicted in Kester's embodiment are dedicated to monitoring productivity, communication, or privacy. (*Id*.)

58. Kester's method is focused on controlling when a user or workstation is permitted to launch an application and when and how the application can access a network. (EX-1004 Abstract, [0040]-[0042].) Kester discloses a "workstation management module" ("WMM") that can detect the launch of an application on a workstation and/or a network access request and determine an access privilege. (EX-1004 [0040]-[0041].) Recognizing that risk may be particular to the individual user or particular workstation, Kester teaches that the access privilege may be unique to

the workstation and/or user. (EX-1004 [0042], [0057] ("module 208 applies the policy(s)/rule(s) associated with the hash that matches the application and/or the user requesting the application or network access"), [0059] ("module 208 can stop execution or network access for the application depending on a policy associated with the user at this workstation").)

59. For example, a user (and/or their workstation) in the finance department may have access to an application that can access and adjust payroll, whereas another user in a different department may be subject to a different policy regarding that or a similar application. *See, e.g.,* EX-1004 [0057] ("module 208 applies the policy(s)/rule(s) associated with the hash that matches the application and/or the user requesting the application or network access"), [0059] ("module 208 can stop execution or network access for the application depending on a policy associated with the user at this workstation").

60. Figure 1 of Kester illustrates an example of multiple workstations (1 through N) linked together on a local area network 100, with each workstation equipped with its own WMM:

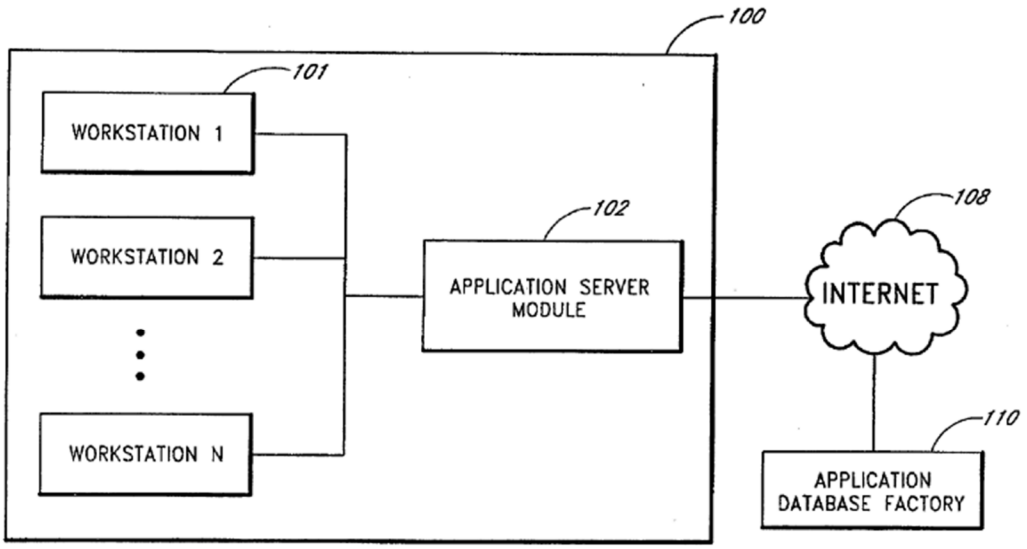

FIG. 1

(EX-1004 Fig. 1.) Kester teaches that each WMM may include a hash/policy table that allows determination of an appropriate policy for each application and/or request. (EX-1004 [0039].)

61. The hash/policy table can include a list of application names, publishers, suites, hashes, ports, protocols, I.P. addresses, categories, and rules or policies associated therewith. (EX-1004 [0054].) For example, in one embodiment, the table comprises a list of hash values associated with corresponding policies, as well as categories that are associated with the hash values and/or policies. (*Id.*) In another embodiment, the table includes names/publishers/suites or other properties which identify the application instead of hash values. (*Id.*) And in another embodiment, the table includes the port/I.P. address/protocol or other properties which identify the applications in the hash/policy table. (*Id.*) Kester's ultimate goal is that, when

a user launches an application or when the application or user requests access to the network, the policy which corresponds to that application or user can be easily identified to determine what access privileges apply to the request. (EX-1004 [0055].)

62. For example, when the WMM detects the launch of an application or a request to access a network, a "hash" or "application digest" is created, based on unique properties related to the application or user requesting access. (EX-1004 [0140]-[0142], Fig. 14 (step 1404), [0051].) The WMM then searches the hash/policy table for a corresponding hash that matches the network request. (EX-1004 [0053], [0142], Fig. 14 (step 1406).) If a match is found, then the policy associated with the hash is applied to the application in response to the network access request. (EX-1004 [0143], Fig. 14 (step 1410).)

63. However, prior to the WMM consulting the hash/policy table, Kester requires that a human reviewer make certain "predetermined associations" between the application/request and its expected network activity, any applicable categories, and a policy. (EX-1004 [0040] ("predetermined association between the launched application and one or more categories"), [0043] ("predetermined association between the application and an expected network behavior or activity"), [0072] ("The network administrator can…select or create access privileges/policies/rules for users, workstation, and/or groups of users/workstations.").)

64. Kester discloses an "application analyst's classification module" with which the human reviewer may interface to determine these associations. (EX-1004 [0178].) Figure 20 of Kester illustrates "a process for analyzing the network access data associated with an application at the application database factor." (EX-1004 [0178].) In particular, Kester describes two different determinations. (*See generally* EX-1004 [0178]-[0187].)

65. For one of these determinations, after the reviewer is provided with a list of applications from the "uncategorized application database" and/or "network access database," the reviewer determines the appropriate category or categories of the application. (EX-1004 [0178], [0134].) Figure 13 further elaborates on this specific determination of categorizing an application. (EX-1004 [0134], Fig. 13.) The application analyst classification module displays the application and any related data on the graphical user interface (GUI) for the human to review. (EX-1004 [0134].) "[T]he human reviewer uses the application, related information, and any Internet information to research the application." (EX-1004 [0135].) "The human reviewer can further review documents, specifications, manuals, and the like to best determine the category or categories to associate with the application." (*Id.*) The reviewer then categorizes (or classifies) each application using this evidence, "any hints from the related information," and/or other research. (*Id.*)

66. As I explained above, this categorization determination could include any one or more of the 44 exemplary categories disclosed by Kester related to "productivity," "audio/video," "communication," etc. (EX-1004 Fig. 4A.) While not included in Figure 4A, Kester further discloses two additional categories: "expected or predetermined network access" and "unexpected network access." (EX-1004 [0178].) Thus, Kester may use the term "expected" or "predetermined" when referring to the general category of expected network access. (*See id.*)

67. Returning to Figure 20, for the other determination, Kester teaches that the human reviewer again reviews the application and any related information on the application analyst's classification module's GUI "to determine an expected or allowed network activity." (EX-1004 [0179]-[0182], Fig. 20 (steps 2004, 2006).) The network activity of the same application running on different applications can be weighted in a predetermined manner. (EX-1004 [0183].) Kester explains that the "expected network activity for a given application can include one or more network attributes," such as a specific protocol, a specific I.P. address, and a specific access port. (EX-1004 [0184].) Once the human reviewer has determined this expected network activity, the WMM stores the expected network activity in the hash/policy table. (EX-1004 [0183].) Kester teaches that for the same application, network attributes may be different for each attempted access. (EX-1004 [0185].) That is, Kester also uses "expected" or "predetermined" to refer to the expected network

activity/attributes that are unique to each entry of the hash/policy table. (*See* EX-1004 [0185].)

68. Thus, the network administrator determines a policy specific to each combination of network attributes and "[e]ach combination of the one or more attributes can be associated with one or more categories." (EX-1004 [0185]-[0186].) And as I discussed above, this policy or category determination may be specific to the user or workstation running the application and/or making the network request. (EX-1004 [0042], [0186].)

69. Kester provides an illustrative example to help put these two different uses of "expected" within context:

> For example, a first entry in the application inventory database 103 corresponds to the application when the application is operating in a predetermined manner. These expected features relate to normal network activity by the application. A second entry corresponds to the same application hash but different parsed properties or features associated with the application when the application is not operating in a predetermined manner.

(EX-1004 [0070].)

70. Thus, a POSITA would understand that Kester first uses "expected" or "predetermined" to refer to the expected network activity (or "expected features") that are unique to each entry in the hash/policy table. (*See id.* [0070], [0179]-[0184].) That is, if an application is operating in an expected or predetermined way,

then there will be an entry in the hash/policy table which matches the network attributes associated with that operation. (*See id.*) But Kester also uses "expected" or "predetermined" to refer to the "expected network access" category—*i.e.*, in the example of paragraph [0070], whether the features for a particular application operation are "operating in a predetermined manner" and "relate to normal network activity by the application." (*See* EX-1004 [0070], [0178], [0134]-[0135].)

71. As another example, you could have a hypothetical application request, which is an "expected network activity"—*i.e.*, the network attributes associated with that request match the "expected" or "predetermined" network attributes that are unique to an entry in the hash/policy table. (*See* EX-1004 [0070].) This same request could also be categorized as "unexpected network access"—*i.e.*, that network attribute combination, while known and recognized, was determined by the human reviewer to be "unexpected network access" for that application. (*See id.* [0070]. [0178].) Accordingly, you would have an application or request that is both "expected network activity" and "unexpected network access."

72. While one might question the need for categories on top of the initial determination as to whether an application matches an entry in the hash/policy table, a POSITA would understand that the variety of categories was critical to Kester's stated goal—providing employers with a system and method for controlling applications files across the "broader intersection" of employees with their computing

environment. (EX-1004 [0006].) By allowing a human reviewer to perform an in-depth review of each application and request, within the context of the particular user(s), workstation(s), and employer's policies, the reviewer could make an informed decision of any and all applicable categories across all types of applications/requests. (EX-1004 [0006], Fig. 4A, [0081]-[0082].) That is, Kester could provide the "broader" "monitoring and controlling" invention it claimed. (EX-1004 [0003], [0006].)

B. Overview Of Kennedy

73. Kennedy discloses a "malicious software detection module" (or "MSDM") that operates on a computer connected to a network. (EX-1005 1:42-52) The MSDM runs on a computer system and "monitors a storage device of the computer system for the arrival of software from a suspicious portal." (EX-1005 1:45-48.) Kennedy's MSDM is implemented through five modules, as illustrated below:

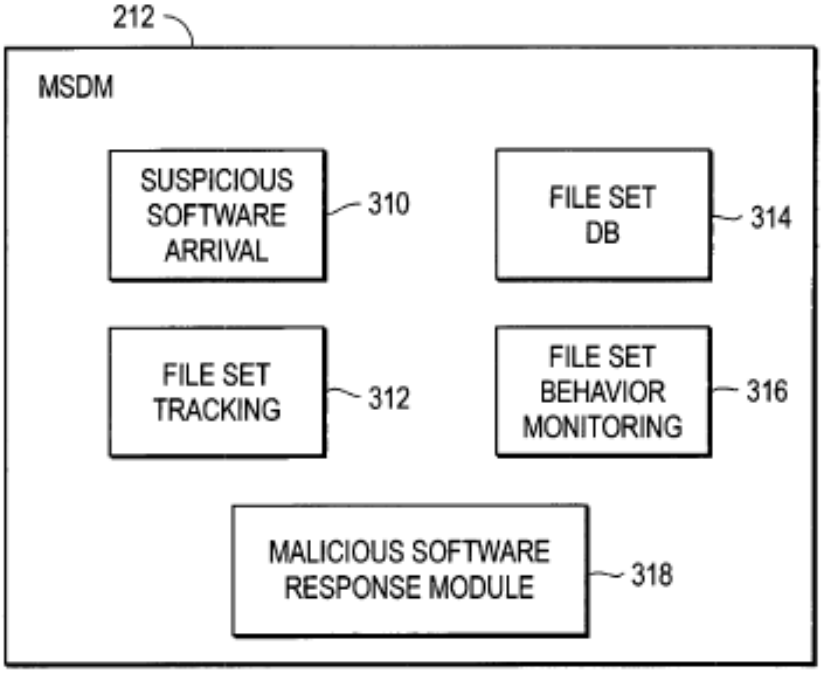

FIG. 3

(EX-1005 Fig. 3.)

74. The "suspicious software module 310 monitors the storage device 108 and detects the arrival of new software," which "can arrive on the storage device 108 via one or more entry points, or 'portals,'" such as open drive shares, specific network ports on the computer system, or removable media readers. (EX-1005 3:47 – 57.)

75. The "file set tracking module 312 maintains a set of files for each suspicious software" and "tracks the behaviors of the files in the set and records each time a file in the set 'touches'" (*e.g.*, create, modify, and/or read) another file on the computer system. (EX-1005 4:19 – 47.) The file set tracking module operates in

concert with "file set database 314, [which] persistently stores the file sets identified by the file set tracking module 312." (EX-1005 4:48-63.)

76. The "file set behavior monitoring module 316 monitors the behavior of the files within the set associated with each suspicious software" both individually and collectively. (EX-1005 4:64 – 5:3.) To determine whether software represented by the file set is malicious, "[t]he file set behavior monitoring module 316 employs one or more heuristics" that describe "one or more conditions that, if satisfied, indicate that the software is malicious." (EX-1005 5:4 – 11.) If the heuristics determine that an object is malware, the malicious software response module 318 performs one or more remedial actions, such as alerting the end user of the client computer system, quarantining the software, preventing replication, or performing secondary tests on the software. (EX-1005 5:12 – 42.)

77. The MSDM's malicious software detection steps are illustrated in Figure 4:

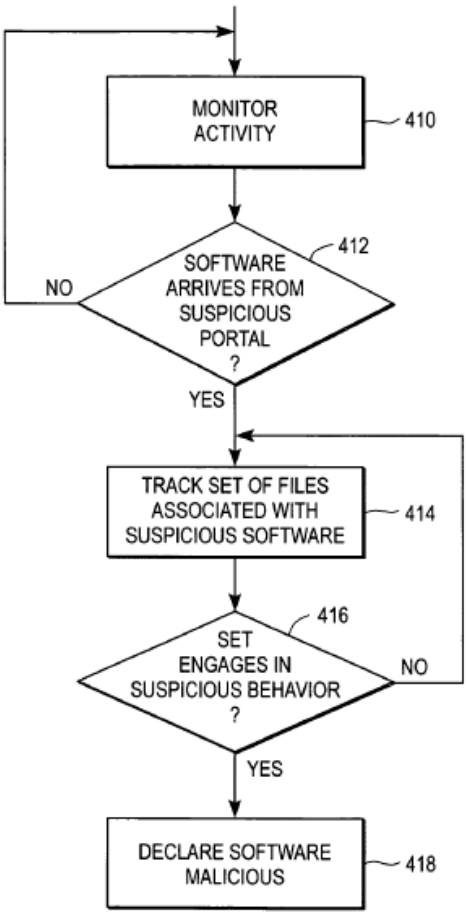

FIG. 4

(EX-1005 Fig. 4.)

78. "The MSDM 212 monitors 410 activity on the local storage device 108." (EX-1005 5:51-52.) The computer system's MSDM will initially classify an object received by that computer system as suspicious if "the software arrived through a portal designated as suspicious." (EX-1005 5:52–56.) Next, the MSDM tracks files on the local storage device that are associated with the software it has designated as suspicious and, if the files in the set either individually or collectively engage in malicious behavior, the MSDM classifies "the software [as] malicious and

notifies the end-user, quarantines the software, blocks the software from performing certain activities, and/or performs other actions." (EX-1005 5:57–6:4.)

VI. KESTER IN VIEW OF KENNEDY AND HONIG DOES NOT DISCLOSE OR RENDER OBVIOUS THE CHALLENGED CLAIMS

79. Petitioner and its expert allege that Kester in view of Kennedy renders obvious every limitation of independent claims 1 and 15, and this allegations underlies their challenge of all remaining dependent claims 2-14 and 16-30.

Ground 1	Section 103(a)	Obvious over Kester in view of Kennedy	Claims 1–13, 15–27, 29, 30
Ground 2	Section 103(a)	Obvious over Kester in view of Kennedy and Kennedy	Claims 14 and 28

80. As I discussed above, I apply the plain and ordinary meaning of all claim terms, consistent with the district court's claim construction order. (*See* Section IV(B), *supra*.)

81. I disagree with the arguments in the Petition and the opinions set forth by Dr. Lee for at least the reasons that I discuss in further detail below.

A. Kester Is Non-Analogous To The '389 Patent

82. I am informed that to qualify as prior art for an obviousness analysis, a reference must qualify as "analogous art," i.e., it must satisfy one of the following conditions: (1) the reference must be from the same field of endeavor; or (2) the

reference must be reasonably pertinent to the particular problem with which the inventor is involved.

83. I understand that in determining whether two references are in the same field of endeavor, the Patent Office's classification of the prior art is some evidence of analogy or non-analogy. I further understand that comparing the structure and functions of the inventions to analyze their similarities and differences can carry even greater weight.

84. I understand that a reference is "reasonably pertinent" to a problem if it, as a result of its subject matter, logically would have commended itself to an inventor's attention in considering his problem. In determining whether a reference is reasonably pertinent, I understand that one should consider the problem faced by the inventor, as reflected – either explicitly or implicitly – in the specification.

85. It is my opinion that Kester does not satisfy either of the conditions listed above.

86. I understand that the '389 patent was classified by the USPTO as "information security" and is titled "methods and apparatus for dealing with malware." (EX-1001 p.1.) As I discussed above, the inventors of the '389 patent sought to resolve the need for detailed, manual analysis and human decision and the inherent delay these introduced. (Section IV(A), *supra*.) The '389 patent explains that in the prior art:

> [I]f the file is not known at the central computer, then the whole file is sent to the central computer where it can be analysed to determine whether it should be regarded as safe or malware. Such analysis is typically carried out manually or "semi-manually" by subjecting the file to detailed analysis, for example by emulation or interpretation, which can still take days given the human involvement that is typically required. There is therefore still a considerable period of time before a new file is classified as safe or as malware.

(EX-1001 2:9-19.) It further acknowledged the security risk, where prior art systems "rely on deep analysis of a new object in order to determine whether or not the object is malicious, which introduces delay and therefore risk to users during the period that the file is analysed." (EX-1001 2:63-66.)

87. Accordingly, the '389 patent sought to improve upon the delays inherent in these prior art methods that relied on manual or semi-manual detailed analysis of an object and a human administrator to decide whether or not the file is safe or malware. Consistent with this goal, the '389 patent claims a method and system which "allows a rapid determination of the nature of the object to be made, without requiring detailed analysis of the object itself as such to determine whether it [is] malware." (EX-1001 3:14-24.)

88. Thus, it is my opinion that the '389 patent's field of endeavor is the rapid determination of whether an object is malware or safe. (EX-1001 3:14-24, 1:9-18, 5:4-8, 8:28-37, cls. 1, 15.)

89. In contrast, Kester was classified as "data processing" and defines its field of invention as "related to computing devices and, more particularly to monitoring and controlling application files operating thereon." (EX-1004 [0002]-[0003], Title, cls. 1, 23, 37, 39; *see also id.* [0008]-[0011] (describing aspects of the invention, all relating to collecting data for or updating a system which "controls" a program's operation or network access).) For example, Kester describes various aspects of its invention, all of which relate to collecting data for or updating a system which "controls" a program's operating or network access. (*See id.* [0008]-[0011].)

90. Dr. Lee and the petition conclude that Kester is in the field of classifying a computer object as malware based on a one-line disclosure in Kester of a single exemplary category, "Malicious Applets and Scripts." (Petition 9-10; EX-1003 ¶63.) But this category is just one exemplary category of more than forty exemplary categories disclosed by Kester that relate to access/privacy, productivity, communication, audio/video, and other types of applications. (EX-1004 Fig. 4A [0082].)

91. While Kester discloses these categories and the fact that a human reviews data to determine whether these categories are applicable to an application, Kester makes no teachings regarding what, in the data, might indicate whether a particular category is appropriate. That is, Kester provides no guidance to a POSITA as to when or why the single "malicious applets and scripts" category should be associated with an application or request. Kester does not even recommend a

particular policy that should be applied if an application were to be categorized as malware. Thus, no POSITA would understand Kester to be in the field of malware classification, just as no POSITA would understand Kester to be in the field of "instant messaging" classification, "media player" classification, "gambling" classification, etc. (*See* EX-1004 Fig. 4A.) A POSITA would not have looked to Kester to inform him on the determination of whether an object is safe or malware. Thus, Kester's field of endeavor (controlling application files) is not the same field of endeavor as the '389 patent (rapid determination of whether an object is safe or malware).

92. I understand that during prosecution of the U.S. Patent Application No. 11/477,807 (the "'807 Application") to which the '389 patent claims priority and of which the '389 patent is a continuation, the applicant made these same observations. (EX-2019 739.) For example, after the examiner opined in an office action that "Dozortsev and Kester are from the same field of endeavor, monitoring for malware," the applicant explained that "Kester has no teachings relating to classifying an object as malware at all." (EX-2019 739.) Rather, "Kester is concerned with a system to allow an employee to control what their employees can do on their computers." (*Id.* 739.) The applicant further identified at least three reasons that "Kester is impractical for identifying malware." (*Id.*) The examiner did not repeat his rejection based on Kester and subsequently allowed U.S. Patent No. 8,418,250 (the

"'250 patent") to issue from the '807 Application. Since the '389 patent is a continuation of the '250 patent, this further supports my opinion that Kester is not in the same field of endeavor as the '389 patent.

93. Kester would also not have logically commended itself to an inventor's attention in considering the problems of the '389 patent, because Kester uses some of the same techniques the '389 patent sought to improve upon. As I discussed above, Kester teaches that human decision, in addition to a detailed analysis of each object, is necessary to its method for controlling application files. (Section V.A., *supra*.) For example, Kester teaches that a human reviewer conducts a detailed analysis of collected data—even Internet research or review of specifications and manuals—to determine expected network activity. (*Id.*) A POSITA would not have looked to Kester (and Kester would not have logically commended itself) to improve upon manual or semi-manual methods that relied on human administrators and detailed analysis, because Kester suffers from the same drawbacks. (*Compare* EX-1004 [0179]-[0186] *with* EX-1001 2:4-27.)

94. I understand that during prosecution of the '807 Application, when the examiner suggested that Dozortsev and Kester were both directed to malware, (EX-1002 612), the applicant corrected the examiner, explaining that Kester was not directed to the detection of malware and was concerned with different problems:

> Kester is concerned with a system to allow an employee to control what their employees can do on their computers. This is in essence a matter

> of individual policy for each company and may vary between companies or even between users in a company. . . . Dozortsev is concerned with detecting malware. Kester is concerned with allowing a company to implement policies to control what programs and network access their employees can make on their computers. These are fundamentally different problems. One of ordinary skill in the relevant art attempting to cure the deficiencies in Dozortsev would have no motivation to look to Kester for relevant teachings as the two documents are in different fields and are concerned with different problems.

(EX-2019 739-740.) The examiner subsequently allowed the related '250 patent to issue, further supporting my opinion that Kester is nor pertinent to the problems considered by the inventors of the '389 patent.

95. Because Kester is neither in the same field of endeavor not pertinent to the problems considered by the inventors of the '389 patent, it is my opinion that Kester is non-analogous to the '389 patent.

B. Independent Claims 1 and 15

96. Petitioner and its expert rely on the same analysis of Kester in view of Kennedy to support Limitations [1.5] and [15.4]. (Petition 24–31, 43; EX-1003 ¶¶105-119, ¶152.) Claim limitations 1.5 and 15.4 each require the base computer comparing data about the computer object received from the first and second remote computers to other objects or entities to identify relationships between the two. (EX-1001, cls. 1, 15.)

Claim Limitation	Claim Language
[1.5]	"*comparing, by the base computer, the correlated data about the computer object received from the first and second remote computers to other objects or entities to identify relationships between the correlated data and the other objects or entities*"
[15.4]	"*the base computer being constructed and arranged to compare the data about the computer object received from the first and second remote computers to other objects or entities to identify relationships between the correlated data and the other objects or entities*"

97. The arguments made by Petitioner and Dr. Lee against claim limitations 1.5 and 15.4 fail to disclose or render obvious these limitations for at least three reasons, which I explain below.

98. First, neither Kester nor Kennedy disclose each element of the "comparing" limitation. Specifically, Petitioner and its expert fail to provide any explanation as to how Kester or Kennedy disclose a base computer that performs the comparing limitation. Instead, the suggestion is made that the human network administrator's analysis may result in the claimed comparison without explaining the basis for this conclusion.

99. Second, a POSITA would not be motivated to modify Kester in view of Kennedy. Kester describes a manual workstation administration technique that

focuses specifically on managing an employee's usage of applications. Kennedy, as I explained above, utilizes a method and system of automated heuristics and file set tracking modules that check whether logical conditions are satisfied. As I further explain below, Kennedy's automated modules sacrifice the degree of control provided by Kester's manual methods and systems, making the two incompatible.

100. Third, even if a POSITA was motivated to modify Kester in view of Kennedy, they would not expect to succeed in modifying Kester's application in view of Kennedy's file set tracking. Although Dr. Lee makes the argument that a POSITA would be motivated to combine the two (a conclusion I disagree with), both Dr. Lee and Petitioner fail to describe *how* a POSITA would actually functionally modify Kester in view of Kennedy or how a POSITA would address the challenges associated with attempting to combine two distinctly different approaches and architectures.

1. Kester and Kennedy Do Not Disclose Or Render Obvious Limitations [1.5] and [15.4].

101. It is my opinion that petitioner does not identify in the cited prior art a "base computer" that performs the claimed comparison, so Petitioner has not met its burden on institution.

102. In connection with limitation 1.1, Petitioner and Dr. Lee appear to argue that Kester's "server module and/or application database factory" is a "base computer." (Petition 14 ("Kester teaches a base computer (application server module

102 and/or application database factory 110)…"); *see* EX-1003 ¶99 ("*Kester* teaches that its application server module and/or application database factor (*base computer*) correlate at least a portion of the collection data received from the remote work-stations.").) But for limitation 1.5, Petitioner does not argue that Kester's application server module and/or application database factory performs the comparison step. Instead, Petitioner acknowledges that Kester describes "a network administrator" who "interface[s] with the application server module 102" to perform the comparison. (Petition 26; EX-1003 ¶107.) However, the claims explicitly require that the "base computer"—and not a human "network administrator"—perform the comparison.

103. Although Petitioner and its expert relies in part on Kennedy for the "comparing" limitations, it does not identify any disclosure in Kennedy of a "base computer." (Petition 27–31; EX-1003 ¶¶109–118.) As a result, Petitioner and its expert cannot identify any disclosure in Kennedy of a "base computer" that performs the claimed comparison. This is confirmed by Petitioner's description of a hypothetical "combined method/system" based on Kester and Kennedy. Petitioner and its expert expressly rely on the "server module and/or database factory" in Kester (as opposed to any disclosure in Kennedy) for the claimed comparison. (Petition at 29 ("In the combined method/system, the application's file set behavior is collection data that is sent to the server module and/or database factory such that when the

comparison of this collection data to other objects indicates the application is related to known malware, Kennedy's automated heuristics can classify the application appropriately as malware."); EX-1003 ¶112 (same.)) However, as addressed above, Petitioner has acknowledged that in Kester, the comparison is not performed by the "server module and/or database factory" (or "base computer"), but rather by a network administrator.

104. Moreover, even if a POSITA considered modifying Kester by adding Kennedy's automated heuristics, the resulting system would still require a human administrator to analyze the output of Kennedy's automated heuristics to perform the comparison step, and would thus not teach the claimed comparison by the base computer. Petitioner and its expert appear to concede this point later in the petition, alleging—confusingly—that a POSITA would have "automatically generate[d] (e.g., without human input) *Kester*'s mask in view of the teachings of Honig" to make Kester's method "more efficient for a human reviewer to use." (Petition 59; EX-1003 ¶192.) That is, Petitioner admits—even if a POSITA were to implement an automated, adaptive learning system to assist or enhance human review, a human decision is still necessary in this manual process.

105. Petitioner cannot identify any "comparing" limitation that is performed "by the base computer" in Kester. Any "comparison" identified in Kester is performed mentally and manually by a network administrator. Given the absence of

this element from the prior art relied upon in all grounds of the petition, it is my opinion that the petition does not have a reasonable likelihood of success on the merits

2. POSITA Would Not Be Motivated To Modify Combine Kester With Kennedy's Automated Classification Heuristics Because Kester's Human Administrator Component Is Incompatible With Kennedy's Alleged Automation.

106. Kester describes manual workstation-/application-specific management that is incompatible with Kennedy's heuristics and file set tracking modules, which merely check whether a logical condition is satisfied.

107. Kester's alleged comparison depends on, and requires, manual human analysis and does not use automation. *See, e.g.*, EX-1004 Fig. 10 block 1006 ("Human Review of Application to Classify Application"); Fig. 13 block 1306 ("Human Reviewer Uses Hints and Web Information to Research Application"); Fig. 17 block 1706 ("Human Review of Application to Determine Allowed Behavior"); Fig. 20 block 2006 ("Human Reviewer Uses Hints and Web Information to Research Network Access by Application"). Indeed, manual human analysis is a critical component of Kester's process, as illustrated by its Figures representing flow diagrams embodying Kester's processes:

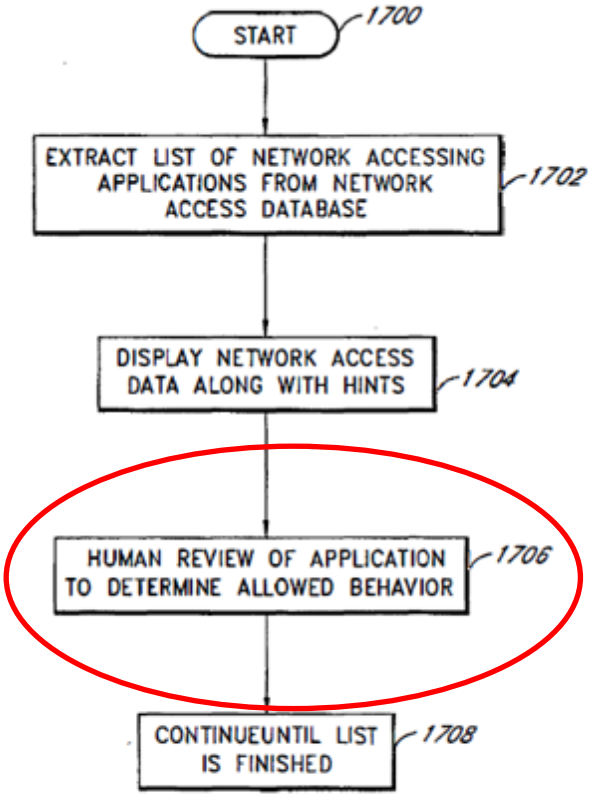

FIG. 17

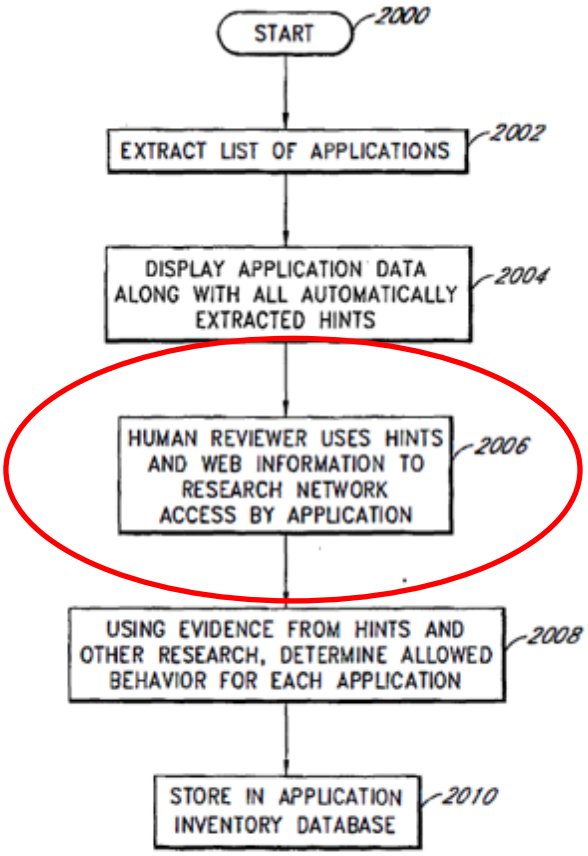

FIG. 20

(EX-1004 Figs. 17 and 20 (annotated in red).)

108. A combination system/method that automates the analysis and categorization performed by Kester's human network administrator would cut out this manual check and sacrifice the tailored approach Kester attempts to provide for enterprise software/workstation management.

109. In addressing this limitation, Petitioner and its expert argue that a POSITA would have combined Kester and Kennedy to "predictably improve the system/method's ability to detect suspicious behavior of different but related files" and "to identify the types of rogue program threats Kester was designed to mitigate more effectively." (Petition 30–31; EX-1003 75-76.). But the petition ignores Kester's requirement for a human analyst, and thus misapprehends any purported benefits from this combination while disregarding the negative implications of automating these manual processes. Specifically, petitioner and its expert ignore the benefits of a workstation-/application- specific management that incorporates information and perspective outside of the system itself, which Kester incorporates through the human network administrator. (*See* EX-1004 [0042], [0090], [0135], [0180]-[0186].)

110. In Kennedy, the MSDM is also the decision maker that determines whether to classify an object as malware. (EX-1005 5:64-6:4 ("If the files in the set either individually or collectively engage in behavior indicated that the software is malicious, in one embodiment, the MSDM 212 declares 418 that the software is malicious and notifies the end-user, quarantines the software, blocks the software from performing certain activities, and/or perform other actions.").

111. The MSDM executes locally on a computer system and "monitors a storage device of the computer system for the arrival of software from a suspicious

portal." (EX-1005 1:45-48.) As such, it applies a localized malware mitigation technique that only addresses malware running on the same system; there are no base computers or remote computers.

112. By contrast, Kester teaches that a human reviewer determines expected network activity and applicable categories within the context of a variety of information from many remote workstations, while keeping in mind that access privilege will be "unique…for the workstation 101 and/or user." (EX-1004 [0042], [90], [0180]-[0186].) Kester teaches the human reviewer conducting independent research about the applications in order to improve the accuracy of its application classification. (EX-1004 [0135] ("The process continues to a state 1306 where the human reviewer uses the application, related information, and any Internet information to research the application. The internet information can be derived from a search using a web browser search engine …. The human reviewer can further review documents, specifications, manuals, and the like to best determine the category or categories to associate with the application."); [0187] (describing "the human reviewer" "using the evidence associated with the application, any hints from related information, and/or other research" to determine the allowed behavior for an application); *see also* [0089] ("A human reviewer interacts with the application analyst's classification module 302 to perform the categorization or recategorization."); [0093] ("[T]he application analyst classification module 302 accesses the Internet 108 and

performs a search based on the application and the collection data . . . The Internet search can provide the application analyst's classification module 302 with additional information for categorizing the application.").

113. Kester would not be improved by Kennedy's heuristic models, which use a fixed set of protocols on a closed universe of information to determine whether a file's behavior indicates it is malware. (EX-1005 5:4–6:10 (describing implementation of the heuristics); *see, e.g., id*. at 5:13–20 ("In one embodiment, a heuristic is designed to detect a computer worm infection like that illustrated in FIG. 2. According to this heuristic, software is malicious if it attempts to copy itself to another computer system via a technique such as email and/or an open drive share. The heuristic is satisfied if a file in a set attempts to send a copy of itself and/or any other file in the same set to the other computer system.") Kennedy and Kester thus employ conflicting approaches to the comparison limitation, and a POSITA would understand that petitioner's proposed combination obviates the benefits of a human administrator's external research for this comparison.

114. During prosecution of the related '250 patent (from which the '389 is a continuation and claims priority), applicant argued that it would not have been obvious to automate the human determinations in Kester because "[i]t is [] essential to have the Network Administrator review each program for each user to decide what policy is appropriate for their company." (EX-2019 739-740.)

> Kester is concerned with a system to allow an employee to control what their employees can do on their computers. This is in essence a matter of individual policy for each company and may vary between companies or even between users in a company.

(EX-2019 739.) Petitioner's obviousness analysis based on Kennedy suffers from the same lack of motivation to modify Kester.

115. Kester does not suggest use of these tools like Kennedy's heuristics at all for the human's determination of expected network activity. (*See* Section V.A., supra.) Automation of these steps within Kester would remove the control that Kester provides to determine application-specific policies across a wide variety of applications and workstations . (EX-1004 Abstract, [0006]-[0007], Fig. 4A; EX-2019 739-740.) In other words, the workstation- and application-specific control provided by Kester's workstation- and application- specific policies could not be achieved with Kennedy's heuristics that merely check if a logical condition is satisfied. (See EX-1005 5:4–6:10.) Thus, Kester does not suggest, and a POSITA would not consider, automating the determination of expected network activity or the categorization of applications.

116. Given the incompatibility between Kennedy's automated heuristics and file set tracking, and Kester's required human input, a POSITA would not look to Kennedy and be motivated to combine it with Kester for the claimed comparison.

3. A POSITA Would Not Expect To Succeed Modifying Kester With Kennedy's Automated Classification Heuristics Because Kester Embodies the Problem Addressed By The '389 Patent.

117. The Petition provides no analysis of how a POSITA would combine Kester and Kennedy's teachings to practice this limitation. Specifically, the petition fails to explain whether a POSITA should incorporate Kennedy's MSDM heuristics on every remote computer, or if somehow the Kennedy's local MSDM heuristics could be integrated within Kester's centralized computer architecture.

118. Kennedy's MSDM application contains the file-set tracking and behavior monitoring modules, but the MSDM resides on a remote computer system—not a base computer—and Kennedy's malware heuristics and file-set tracking look only at the computer objects on that single remote device. (EX-1005 1:45-48 ("The MSDM executes on a computer system connected to a network. The MSDM runs on a computer system and 'monitor a storage device of the computer system for the arrival of software from a suspicious portal.'"); *see also* Section V.B., *supra*.) The Petition fails to address how, or even where, the MSDM could be incorporated into Kester's computer architecture, or explain how to reconfigure the MSDM within Kester in order to identify relationships with "other objects" not on the same base computer.

119. Notwithstanding Petitioner's failure to explain how to modify Kester in view of Kennedy, Petitioner and its expert argue that a POSITA would have a

reasonable expectation of success "using known methods with no change to their respective functions" (Petition 31; EX-1003 ¶ 118) and that since "file set tracking and behavior monitoring was a well known-way to detect the presence of malware, there would have been a reasonable expectation of success configuring Kester's application file monitoring system/method to perform Kennedy's file set tracking and behavior monitoring." (*Id.*)

120. Petitioner and Dr. Lee fail to address the challenges, and negative consequences, that would result from this proposed combination of Kester and Kennedy. Even if petitioner and Dr. Lee proposed a workable approach to practically incorporating Kennedy's file set tracking and file monitoring modules, a POSITA would understand that petitioner's proposed combination to automate the comparison step obviates Kester's claimed benefit of a human administrator's external research and analysis for this comparison. Indeed, this is a critical step in Kester's process. (*See* EX-1004 [0135].)

> The process continues to a state 1306 where the human reviewer uses the application, related information, and any Internet information to research the application. The application information can be derived from a search using a web browser search engine. The application name and any of the related application data can be used for the Internet search. The human reviewer can further review documents, specifications, manuals, and the like to best

> determine the category or categories the associate with the application. The process continues to a state 1307 where the human reviewer classifies each application using the evidence associated with the application, any hints from the related information, and or research.

(EX-1004 [0135].) Petitioner's argument thus misapprehends Kester and ignores the role of the human reviewer to suggest that this can simply be accomplish by automated heuristics and file set tracking.

121. Accordingly, Petitioner has not shown that such a combination would render obvious claim limitation 1.5 and 15.4.

C. Dependent Claims

122. Because Kester in view of Kennedy does not render obvious all limitations of the independent claims 1 and 15, I understand that Grounds 1 and 2 fail to render obvious dependent claims 2-14 and 16-30 as well.

VII. CLAIMS AND/OR LIMITATIONS NOT ADDRESSED IN THIS DECLARATION

123. It is my understanding that this declaration is being provided in support of Patent Owner's Preliminary Response. I have provided analysis of key limitations not included in the prior art. The fact that I have not analyzed particular limitations or claims in no way indicates or supports that these limitations or claims are disclosed or rendered obvious by the prior art, and I reserve the right to supplement my analysis if asked to do so.

VIII. SECONDARY OBJECTIVE INDICIA OF NONOBVIOUSNESS

124. I understand that in determining obviousness of the claimed invention, objective evidence of secondary considerations pertaining to nonobviousness must be considered, including long-felt but unsolved needs, failure of others, commercial success, etc. (*In re Huang*, 100 F.3d 135, 139 (Fed. Cir. 1996) (citing *Graham v. John Deere Co.*, 383 U.S. 1, 17-18 (1966).) I further understand that these indicia constitute independent evidence of nonobviousness that can be the most probative evidence in that regard "in avoiding the trap of hindsight when reviewing, what otherwise seems like, a combination of known elements." (*Leo Pharm. Prods., Ltd. v. Rea*, 726 F.3d 1346, 1358 (Fed. Cir. 2013).)

125. As demonstrated by my summary of the prior art, above, there was at least a long-felt and persistent need prior to the '389 patent to improve upon the time-consuming methods for determining whether computer objects are safe or malware. (*See* Section III, *supra*.) Prior attempts to develop such methods were limited in their ability to make rapid determinations and required extensive manual or semi-manual detailed analysis of objects and/or human administrator decisions. (*See* Section III, *supra*.) Where others failed, the '389 Patent succeeded. (*See* Section IV(A), *supra*.)

126. Thus, these secondary indicia only further support my opinion that the '389 Patent was non-obvious.

IX. CONCLUSION

127. For at least the foregoing reasons, it is my opinion that Petitioner's grounds do not render obvious any of the challenged claims of the '389 Patent.

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

Executed this 16th day of June, 2023 in Los Angeles.

Respectfully submitted,

Nenad Medvidovic, Ph.D.