

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

CROWDSTRIKE, INC.
Petitioner

v.

OPEN TEXT INC.
Patent Owner

Case IPR2023-00124
Patent No. 9,578,045

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

Mail Stop PATENT BOARD
Patent Trial and Appeal Board
U.S. Patent & Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450

TABLE OF CONTENTS

I.	QUALIFICATIONS	2
II.	UNDERSTANDING OF RELEVANT LEGAL PRINCIPLES	7
III.	INTRODUCTION	9
IV.	INADEQUACY OF THE PRIOR ART AND THE NEED FOR IMPROVED NETWORK FORENSICS TECHNIQUES.	11
V.	THE '045 PATENT'S NETWORK FORENSICS TECHNIQUES.	14
A.	Overview of the '045 Patent.....	14
B.	Claim Construction.....	18
1.	“Originating Object”	22
2.	“End Recipient Device”	26
C.	Level Of Ordinary Skill In The Art.....	29
VI.	THE CHALLENGED CLAIMS ARE PATENTABLE.	31
A.	Overview of Morris	34
B.	Overview Of Capalik.....	36
C.	Overview of Van Oorschot	39
D.	Petitioner Fails to Establish That Morris, Alone or in Combination, Renders Obvious Claims 1–4, 8-10, 12 and 16.....	42
1.	Morris Does Not Disclose Or Render Obvious The “Originating Object” Limitations	44
2.	Morris Does Not Disclose the “Sensor Agent” in Claims 8 and 16	49
3.	Morris Does Not Render Obvious a Sensor Agent Transmitting the Event Line to the End Recipient Device	59

4.	Morris in View of van Oorschot Does Not Render Obvious “Obtaining an Event Priority” in Claim 3	63
E.	Petitioner Fails To Establish That Capalik, Alone or in Combination, Renders Obvious Claims 1–4, 7–10, 12, 15-18	65
1.	Capalik Does Not Disclose Or Render Obvious A Global Perspective Related to One or More Other Events related to the Event Across a Network	66
2.	Capalik in View of van Oorschot Does Not Render Obvious Claim 3’s “Event Priority” Because van Oorschot’s “Severity Level” Does Not Include an Indication of any Originating Object.....	74
VII.	PETITIONER’S GROUNDS FAIL TO ANTICIPATE OR RENDER OBVIOUS ALL DEPENDENT CLAIMS.....	75
VIII.	SECONDARY OBJECTIVE INDICIA OF NONOBVIOUSNESS.	75
IX.	CONCLUSION.....	76

EXHIBIT LIST

Exhibit	Description
2001	Declaration of Professor Nenad Medvidovic, Ph.D.
2002	<i>Curriculum Vitae</i> of Professor Nenad Medvidovic, Ph.D.
2003	<i>Webroot, Inc. and Open Text Inc., v. CrowdStrike, Inc. and CrowdStrike Holdings, Inc.</i> , Case No. 6:22-cv-00241-ADA-DTG, First Amended Complaint for Patent Infringement, Dkt. 60 (W.D. Tex., Sept. 8, 2022) (excerpted)
2004	<i>Sonrai Memory Ltd. v. Kingston Tech. Co. and Kingston Tech. Corp.</i> , Case No. 6:21-cv-1284-ADA, Order Denying Kingston Tech. Co. and Kingston Tech. Corp.’s Opposed Motion to Stay Pending Resolution of <i>Inter Partes</i> Review, Dkt. 94 (W.D. Tex., Oct. 18, 2022)
2005	<i>mCom IP, LLC v. Cisco Systems, Inc.</i> , Case No. 6:22-cv-00261-ADA, Order Denying Defendant’s Motion to Stay Pending <i>Inter Partes</i> Review, Dkt. 42 (W.D. Tex., Oct. 20, 2022)
2006	<i>Intellectual Ventures I LLC et al. v. Hewlett Packard Enterprise Co.</i> , No. 6:21-cv-00596-ADA, Order Denying Hewlett Packard Enterprise Company’s Opposed Motion to Stay Pending Resolution of <i>Inter Partes</i> Review, Dkt. 104 (W.D. Tex., Dec. 22, 2022)
2007	<i>Webroot, Inc., Open Text Inc., v. AO Kaspersky Lab et al.</i> , Case No. 6:22-cv-00243-ADA-DTG, Order Granting Third Amended Scheduling Order, Dkt. 142 (W.D. Tex., Dec. 27, 2022)
2008	<i>Standing Order Governing Proceedings (OGP) 4.2 – Patent Cases</i> (W.D. Tex., Sept. 16, 2022), available at chrome-extension://efaid-nbmnnnibpcajpcglclefindmkaj/https://www.txwd.uscourts.gov/wp-content/uploads/Standing%20Orders/Waco/Albright/Standing%20Order%20Governing%20Proceedings%20-%20Patent%20Cases%20091622.pdf (excerpted)

Exhibit	Description
2009	<i>Webroot, Inc. and Open Text Inc., v. AO Kaspersky Lab et al.</i> , Case No. 6:22-cv-00243-ADA-DTG, Defendants’ Opening Claim Construction Brief, Dkt. 86 (W.D. Tex., Oct. 28, 2022) (excerpted)
2010	<i>Webroot, Inc. and Open Text Inc., v. AO Kaspersky Lab et al.</i> , Case No. 6:22-cv-00243-ADA-DTG, Defendants’ Reply Claim Construction Brief, Dkt. 110 (W.D. Tex., Dec. 6, 2022) (excerpted)
2011	<i>Webroot, Inc. and Open Text Inc., v. AO Kaspersky Lab et al.</i> , Case No. 6:22-cv-00243-ADA-DTG, Scheduling Order, Dkt. 50 (W.D. Tex., Sept. 28, 2022) (excerpted)
2012	<i>Open Text Inc. et al., v. AO Kaspersky Lab</i> , Case No. 6:22-cv-00243-ADA-DTG, Defendant AO Kaspersky Lab’s Preliminary Invalidity Contentions Exhibit 4A (W.D. Tex., Sept. 14, 2022) (excerpted)
2013	<i>Webroot, Inc. and Open Text Inc., v. Sophos Ltd.</i> , Case No. 6:22-cv-00240-ADA-DTG, Defendant Sophos Ltd.’s Preliminary Invalidity Contentions (W.D. Tex., Sept. 13, 2022) (excerpted)
2014	<i>Webroot, Inc. and Open Text Inc., v. Trend Micro Inc.</i> , Case No. 6:22-cv-00239-ADA-DTG, Defendant’s Preliminary Invalidity Contentions (W.D. Tex., Sept. 13, 2022) (excerpted)
2015	<i>Webroot, Inc. and Open Text Inc., v. CrowdStrike, Inc. and CrowdStrike Holdings, Inc.</i> , Case No. 6:22-cv-00241-ADA, Complaint for Patent Infringement, Dkt. 1 (W.D. Tex., Mar. 4, 2022)
2016	<i>Open Text Inc. et al., v. AO Kaspersky Lab</i> , Case No. 6:22-cv-00243-ADA-DTG, First Amended Complaint for Patent Infringement, Dkt. 80 (W.D. Tex., Oct. 27, 2022)
2017	<i>Webroot, Inc. and Open Text Inc., v. Trend Micro Inc.</i> , Case No. 6:22-cv-00239-ADA-DTG, First Amended Complaint for Patent Infringement, Dkt. 53 (W.D. Tex., Sept. 7, 2022)

Exhibit	Description
2018	<i>Webroot, Inc. and Open Text Inc., v. Sophos, Ltd.</i> , Case No. 6:22-cv-00240-ADA-DTG, Second Amended Complaint for Patent Infringement, Dkt. 76 (W.D. Tex., Oct. 24, 2022)
2019	File History for Continuation Patent, U.S. Patent No. 10,257,224
2020	U.S. Publication No. 20080016570 to Capalik (“Capalik 570”)
2021	<i>Webroot, Inc. and Open Text Inc., v. AO Kaspersky Lab. et al.</i> , Case No. 6:22-cv-243-ADA-DTG, (W.D. Tex. Dec. 9, 2022) Dkt. 133, Transcript of Motions Hearing (excerpted)
2022	U.S. Patent No. 10, 257,224 to Joseph Jaroch (“’224 Patent”)
2023	Joseph Jaroch LinkedIn
2024	<i>Webroot, Inc. and Open Text Inc., v. AO Kaspersky Lab. et al.</i> , Case No. 6:22-cv-243-ADA-DTG, Claim Construction Order, Dkt. 236 (W.D. Tex. Mar. 16, 2023)
2025	<i>CrowdStrike, Inc. v. Open Text Inc.</i> , IPR2023-00126, Deposition Transcript of Wenke Lee, Ph.D., dated July 17, 2023
2026	Second Declaration of Professor Nenad Medvidovic, Ph.D.
2027	Email chain from Brian Eutermoser to Hunter Horton regarding CrowdStrike, Inc. v. Open Text Inc. IPR2023-00124 - r. Lee deposition

I, Professor Nenad Medvidovic, 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 9,578,045 (“the ’045 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 ’045 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 ’045 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. 9,578,045 and supporting exhibits, as well as those Exhibits attached to the Patent Owner Preliminary Response, Petitioner’s Reply, Patent Owner’s Sur-Reply, and Patent Owner’s Response.

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 '045 patent at the time of the alleged invention. I have also been asked to provide my technical opinions on how concepts in the '045 patent specification relate to claim limitations of the '045 patent. In reaching the opinions provided herein, I have considered the '045 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 field, 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 nearly 30 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 10 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. I am currently serving as General Co-Chair of the 21st International Conference on Software Architecture (ICSA 2024).

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. And in May 2023, I was inducted into the University of California Irvine Donald Bren School of Information and Computer Science Alumni Hall of Fame. 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.

III. INTRODUCTION

27. The '045 patent introduced a novel network forensic technique capable of collecting and using contextual information relating to the activity of computer objects at endpoints. The '045 patent enabled the detection and identification of computer objects by intercepting network events, creating audit trails for each individual

event by correlating the event to objects such as their originating processes, devices, and/or users, and establishing a global perspective of the computer objects.

28. I understand that Petitioner is a direct competitor of Webroot and began providing endpoint security software and systems that, without authorization, implemented Patent Owner's patented technology, including the '045 patent. I further understand that, in March 2022, Patent Owner filed suit against Petitioner in the Western District of Texas, asserting infringement of the '045 patent and other patents in Patent Owner's endpoint security portfolio.

29. Petitioner's challenge to the '045 patent should be rejected. All of Petitioner's grounds distort the prior art, and even then, still fail to disclose or render obvious the claimed invention.

30. Ground 1 fails because (1) Morris does not render obvious the "originating object" limitation that is part of each challenged claim, and Petitioner wrongly cites GenesisActor information for this limitation—which the '045 patent expressly distinguishes from the claimed "originating object"; (2) with respect to independent claims 8 and 16 and their dependent claims, Morris does not disclose or render obvious a "sensor agent stored on the data storage that is executable by the processor" because there is no sensor agent in Morris that performs all steps required of the sensor agent in claims 8 and 16; (3) with respect to dependent claim 10, Morris does not render obvious the claimed end recipient device or transmitting an event line

thereto; (4) with respect to de-pendent claim 3, van Oorschot's "severity level" is not based on generated con-text because it does not include an originating object, so it cannot render obvious the claimed event priority.

31. In its institution decision, the Board correctly found that Petitioner failed to establish a reasonable likelihood that it would prevail on its Capalik-based Ground 2. (Paper 11 at 44-47.) Capalik's system observes computer objects running on virtual machines and virtual operating systems on a single device. The '045 patent teaches obtaining information about computer objects operating across a network, i.e., not on the same device. A person of ordinary skill in the art ("POSITA") would not understand Capalik's description of information collected from virtual operating systems on a single device as meeting the "across a network" limitation in the challenged claims. Furthermore, like Ground 1, Ground 2 fails because van Oorschot's "severity level" is not based on generated context.

32. Accordingly, it is my opinion that Petitioner has not met its burden and the challenged claims are patentable.

IV. INADEQUACY OF THE PRIOR ART AND THE NEED FOR IMPROVED NETWORK FORENSICS TECHNIQUES.

33. Network forensics generally relates to intercepting and analyzing network events to discover the source of security attacks. (EX-1001 1:22-24.) At the time of invention of the '045 patent, prior art network forensic techniques commonly

used network forensics tools, e.g., network sniffers and packet capture tools, to detect and capture information transmitted during communication sessions. (*See* EX-1001 1:24-33.)

34. Although such network forensics tools passively collect network traffic, the tools reside at a network edge, e.g., a location where an organization's network interfaces with the Internet. (*See* EX-1001 1:34-43.) That is, a network forensics tool, such as a network sniffer, may reside at a network edge and may receive network events concerning the network from one or most hosts or systems, but network edge devices cannot establish any sort of context from within the host or system generating the event. (*See* EX-1001 1:40-43.) As a result, network forensics tools could not obtain useful information within a host or establish any sort of context from within a host that was generating and/or receiving network events. (EX-1001 1:43-49; 2:5-12; *see also* EX-2019 385-86.)

35. Without this information, prior art network forensics tools were insufficient to identify and address the threats faced by modern computer networks. (*See* EX-1001 1:46-49.) Although a network edge device may be able to monitor, capture, and analyze communications traffic, the network edge device could not provide details regarding files modified, registry entry changes, or new processes created within the network core. (*See* EX-1001 2:5-12.) The network edge device thus had no way of knowing to what extent a virus may have impacted computing devices on

the network, what additional files may have become infected, or what operating processes and/or objects are linked to the virus. (*See* EX-1001 2:12-16.)

36. Then-existing prior art techniques that were able to collect context about events from within a computing device employed low-level system filters to intercept events. System events “tend to be very raw and generally not individually useful—on an average system, tens of millions of events take place every minute and the noise ratio can prevent forensic solutions from being able to provide sufficient value to the end consumer of their data due to an inability to quickly find important events.” (EX-1001 2:39-65.) Transmitting all the collected event information after applying only a low-level filter resulted in inefficient transmissions that included more information than an end consumer needed to identify important events. (*Id.*; *see also* 8:4-19.) “[M]any applications perform thousands of the same events (network access to the same destination IP address, for example) and transmitting these events in-to an external log source would create a significant amount of unnecessary noise and potential performance degradation.” (EX-1001 3:9-14.)

37. Moreover, prior art techniques that implemented data normalization after transmitting event information suffered from delays and inefficiencies resulting from unnecessary redundancy. (EX-1001 3:19-37.) Data normalization refers to the process whereby “items such as kernel filenames, usermode file-names, process objects, registry objects, and other system structures” are normalized into a standard

format that is more easily consumed by automated parsers. (EX-1001 3:19-24.) By consolidating and normalizing data at a later stage, i.e., after transmission, results in more reprocessing of data “and potentially missing events just because they were referred to slightly differently (\Device\HarddiskVolume1\Windows\file.exe instead of C:\Windows\file.exe for example).” (EX-1001 3:23-28.)

38. Thus, new network forensic techniques were needed to provide improved forensic end-to-end visibility without sacrificing precision or speed, or at the expense of significant computing resources.

V. THE '045 PATENT'S NETWORK FORENSICS TECHNIQUES.

A. Overview of the '045 Patent

39. The '045 patent is entitled “Method and Apparatus for Providing Forensic Visibility into Systems and Networks” and describes and claims a system and method that improved on traditional network forensics tools used to discover or identify security issues on computer networks.

40. Unlike traditional network forensics tools, the '045 patent efficiently creates forensic visibility *into* the computing devices on the communication network to identify malware or other security issues in the operation of those devices using enhanced local processing of event information. (EX-1001 2:35-38.) In particular, the '045 patent improves network security by gathering an “event,” generating “contextual state information,” obtaining a “global perspective” for the event, generating

an “event line” comprising information relating to the “event,” and transmitting the generated “event line” that includes information for the event. (*See* EX-1001 cl. 1.) The described and claimed systems and methods intercept network events, create audit trails for each individual event by correlating the event to objects such as their originating processes, devices, and/or users, and establish a global perspective of the objects. (EX-1001 3:45-49.)

41. The ’045 patent uses a “local aggregator and interpreter” that “receives events from the low level system filters, collecting them in batches from the queues in which they are inserted in real-time, and transmits these events to a local service process where the events are held for further analysis.” (EX-1001 2:66-3:18.) During this stage of processing, “intelligent caching may be invoked to drop events which are duplicates of recent events, and some data normalization may be performed to ensure consistency.” (EX-1001 3:4-7.) This caching stage “ensure[s] redundant events are not continually reported—many applications perform thousands of the same events... and transmitting these events into an external log source would create a significant amount of unnecessary noise and potential performance degradation.” (EX-1001 3:7-14.)

42. The ’045 patent also uses local resources, i.e., a local aggregator and interpreter, for improved data normalization before transmitting information, which “at this stage allows the data to be handled much more quickly at the higher levels

to avoid reprocessing data and potentially missing events.” (EX-1001 2:66-3:28.) This local subsystem “provides the ability to process events asynchronously,” which permits events to be “intelligently queued in multiple layers to minimize the number of individual threads but still ensure each event is given sufficient time to be independently processed. Because of this structure, there is no performance difference if an event takes one millisecond to be processed or one hour for the full series of routines to complete.” (EX-1001 3:28-40.)

43. In addition to analyzing the behavior of objects to identify those that are potentially malicious, malware detection is further improved by understanding the context of the event and computer objects of interest. (EX-1001 2:23-27, 3:41-49 (“[T]he context analyzer is utilized to ensure that a user, such as a network administrator, is provided the deepest view of the most important data but still has access to the full range of data if they wish to see it. . . . This allows an administrator to forensically track back any event which occurs to what triggered it.”).) In many cases, a potentially malicious object is identified by the system as a result of other events that provide information as to whether the code is malicious. (*See* EX-1001 22:43-50 (“Accordingly, an event having a specified Actor, Victim, Target, originating process, or the like together with a global perspective indicating that at least one of the Actor, Victim, Target, origination process has been determined to be malware may result in a rule set that assigns a priority level to the event indicating that [] the

event should be immediately brought to the attention of an administrator.”) The ’045 patent explains that “[t]he analysis starts by correlating events to objects (their originating processes, devices, and users), and creating an audit trail for each individual event.” (EX-1001 3:41-58.) Using the originating object responsible for the current event under analysis enables “an administrator to forensically track back any event which occurs to what triggered it.” (EX-1001 3:47-49.) This approach of correlating an object with the particular object that created it “eliminates the guesswork and allows for end-to-end visibility of every event from the bottom of the stack to the top.” (EX-1001 3:53-55; 12:57–60 (“the context analyzer 348 may correlate an actor, victim, and/or event type to one or more originating processes, devices, and users. Such a correlation creates an audit trail for each individual event.”).)

44. The ’045 patent further explains that, in addition to context information, the patented systems and techniques can also use information from the network to obtain a global perspective of the network operation. (EX-1001 3:41–4:7.) The combination of contextual information and global perspective enables detection of new zero-day threats, including objects created from objects (or similar objects) that have been identified previously as malicious. (EX-1001 3:55-4:4.) As malicious code and threats evolve more quickly, this combined approach improves detection time and enhances overall network security. (EX-1001 3:62–4:1, 9:39-49)

45. The patents further disclose technical improvements to forensics systems by “assembling” or “generating” an “event line” based on the contextual information and global perspective. (EX-1001 9:50–58.) The generation of the event line makes it easier for end users to “identify events, and/or instances of malware, that require more immediate attention”—thereby improving the accuracy and efficiency of identifying additional malicious code, as well as enabling administrators to more readily analyze malware, assess vulnerabilities, and correct any damage inflicted by an originating object. (EX-1001 9:45–49.)

46. Thus, the '045 patent describes and claims systems and methods that provide more efficient and accurate identification of malware and improved the identification of other malware (and corresponding events) that might otherwise go undetected in prior systems, thereby improving system performance and reducing the required resources. (*See* EX-1001 2:16–32.)

B. Claim Construction

47. I understand that Petitioner does not offer a construction for any claim term and purports to apply the ordinary and customary meaning of the claim terms as understood by a POSITA. (Petition 5.)

48. I understand that March 16, 2023, the district court concluded that no terms of the '045 patent were indefinite.¹ (EX-2024 8-11.) In its claim construction order, the Court construed certain limitations of the '045 patent as follows:

Claims	Claim Term	Court's Construction
1	"wherein the global perspective for one or more related events to the at least one event across a network"	Not indefinite "wherein the global perspective [is] for one or more related events to the at least one event across [other devices on] a network"
1 ²	"global perspective"	Not indefinite. Plain-and-ordinary meaning wherein the plain-and-

¹ The Court provided additional clarifying remarks regarding certain terms' plain and ordinary meaning. For brevity and because those clarifications are not critical to the issues addressed in this response, Patent Owner does not reproduce those clarifying remarks here and instead includes the district court's claim construction order as an exhibit.

² The Court's claim construction order only construes "global perspective" for claim 1. (EX-2024 8.) Because "global perspective" also appears in independent claims 8 and 16 and the petition provides no basis for applying a different meaning of "global perspective" for those claims, Patent Owner applies the Court's

Claims	Claim Term	Court's Construction
		ordinary meaning is “information about events related to the monitored even [sic] ³ across a network”
1 ⁴	“event line”	Plain-and-ordinary meaning.
1, 8, 16	“gathering an event defining an action of a first object acting on a target, wherein the first object is executed on a device; generating contextual state information for the event by correlating the event to an originating object of the first	No order of steps required

construction for “global perspective” in claim 1 for all claims that recite a “global perspective” limitation.

³ Patent Owner understands “even” in the Court’s claim construction order as a typographical error that should instead recite: “event.” (*See* EX-2024 8.)

⁴ The Court’s claim construction order only construes “event line” for claim 1. (EX-2024 8.) Because “event line” is also recited by independent claims 8 and 16 and the petition provides no basis for applying a different meaning of “event line” for those claims, Patent Owner applies the Court’s construction for “event line” in claim 1 for all claims that recite an “event line” limitation.

Claims	Claim Term	Court's Construction
	object; obtaining a global perspective for the event based on the contextual state information, wherein the global perspective comprises information associated with one or more of the first object and the originating object, and wherein the global perspective relates to one or more other events related to the event across a network”	

49. I reviewed the transcript from Dr. Lee’s deposition (EX-2025) and understand from his testimony that he did not consider the claim construction order but had no new or altered opinions based on the Court’s claim construction order. (EX-2025 70:8-14.)

50. Patent Owner and I apply the Court’s construction for all construed terms and the plain and ordinary meaning of the claim terms that the Court did not construe. Although not defined by the Court, two claim terms—“originating object” and “end recipient device”—are directly relevant to Petitioner’s obviousness arguments, and therefore require further clarification for purposes of this proceeding. I therefore address each of those terms below.

1. “Originating Object”

51. I understand that Petitioner argues that GenesisActor information about an object is a type of originating object information, and that Morris’s use of GenesisActor information discloses the ’045 patent’s claimed “originating object.” (Petition 14-15.)

52. Petitioner’s implicit construction of “originating object” improperly broadens the scope of that term as it would be read by a POSITA in view of the specification.

53. The ’045 patent distinguishes between an originating object (or originating process⁵) and GenesisActor information. The originating object refers to “a previous event that initiated or otherwise created the current process and/or object.” (EX-1001 17:40-42.) However, the GenesisActor “is determined based on a chain of previous events.” (*Id.* at 17:42-43.) The specification also explains that “GenesisActor information may contain the object that *is not the direct Actor of an event* but which is the ultimate parent of the event being performed.” (EX-1001 8:45-49

⁵ The ’045 patent uses “originating object” and “originating process” interchangeably. (See EX-1001 17:39-42 (“Moreover, as illustrated in FIG 5B, the originating process or object may determined [sic] based on a previous event that initiated or otherwise created the current process and/or object.”).)

(emphasis added).) When read as a whole, the specification shows that the originating object is the object that directly created the first object. In contrast, the GenesisActor may be many steps removed from the first object through a chain of previous events. (EX-1001 17:42-43.)

54. This distinction is significant because the GenesisActor may provide less helpful contextual state information than the originating object, which is more closely related to the first object. Just as information about a person's parents is likely to be more helpful than information about a distant relative, contextual state information based on the originating object is likely to be more helpful than contextual state information based on the GenesisActor.

55. Nothing in the specification suggests that the GenesisActor may substitute for the originating object to provide contextual state information. On the contrary, the specification repeatedly presents the originating object and the GenesisActor as separate pieces of information. For example, the specification states that “[c]ontextual state information may include, but is not limited to, *originating processes*, devices, and users. *Additionally*, contextual state information may include *GenesisActor information* and ancillary data.” (EX-1001 8:42-45 (emphasis added).) As another example, the specification states:

Moreover, as illustrated in FIG. 5B, the *originating process or object* may be determined based on a previous event that initiated or otherwise created the current process and/or objects. *Additionally*, as illustrated

in FIG. 5B, the **GenesisActor** is determined based on a chain of previous events.
(EX-1001 17:39-43 (emphasis added).) Figure 5B shows that the “Originating Process” and “Genesis Actor” are two different and separate fields in the data structure of the context 520B.⁶ In other words, the ’045 patent includes a separate field for “Originating Process” information even if it has “Genesis Actor” information. Indeed, by consistently contrasting the originating object with the GenesisActor, the ’045 patent informs a POSITA that GenesisActor information should not be substituted for the originating object. While the GenesisActor may provide additional information, it is distinct from and does not replace the *originating object*.

56. The distinction between the GenesisActor and the originating object is consistent with the broader context of the ’045 patent, which uses audit trails to provide “end-to-end visibility” into events.

[T]he context analyzer 348 may correlate an actor, victim and/or event type to one or more originating processes, devices, and users. Such a

⁶ As shown in Figure 5B, it is possible for the originating object and the GenesisActor to be the same object (Object 1) despite each representing a distinct value (i.e., immediate parent vs. ultimate parent). However, this is only possible in one unique circumstance, i.e., if the Actor (Object 2) is the second link in the event chain. Regardless, the ’045 patent clearly distinguishes each value—including in the unique circumstance that they are the same object. (EX-1001 Fig. 5B.)

correlation creates an **audit trail for each individual event**. As will be described later, this audit trail allows an administrator, at an end recipient 220 for example, to forensically track back any event which occurs to what triggered it.

(See EX-1001 12:55–63.) This aspect of the '045 patent is identified at the very beginning of the specification: “methods and systems for providing forensic visibility into client systems having **full** audit trails are described.” (EX-1001 1:15-18 (emphasis added); *see also* EX-1001 3:53-55 (“The present solution eliminates the guesswork and allows for end-to-end visibility of every event from the bottom of the stack to the top.”).)

57. A full audit trail that allows forensic tracking back to “any event which occurs to what triggered it” requires each individual event’s immediate parent. Skipping to the ultimate parent, as Petitioner proposes through its implicit “originating object” construction, results in an incomplete event audit trail due to gaps between the immediate parent and the ultimate parent. In contrast with Petitioner’s construction, the '045 patent uses the claimed “originating object” to create a full audit trail “to forensically track back any event which occurs to what triggered it.” (See EX-1001 12:55–63.) A full audit trail cannot be achieved with GenesisActor information because there would be gaps between the originating object (i.e., immediate parent) and the ultimate parent.

58. Accordingly, “originating object” should be construed as the “previous event that initiated or otherwise created the current process and/or objects.”

2. “End Recipient Device”

59. Claim 10 requires the sensor agent to “transmit the event line to the end recipient device.” (EX-1001 cl. 10.)

Petitioner argues that Morris’s remote computer is the location of the claimed *sensor agent* (Petition 35 (“*Morris* discloses its ‘agent software’ running on remote computer 2 (i.e., *sensor agent*)”). Petitioner further argues that the remote computer transmits the event line to the *base computer*. (Petition 35 (“*sensor agent*[] transmits the key comprising the event and contextual data (i.e., *event line*) to base computer 3 (i.e., *end recipient device*) to perform the above-described malware analysis”).) Alternatively, Petitioner argues that the event line may be “‘copied to’ the local database of a remote computer that first ran the object.” (Petition 35 citing *id.* 24-25). Neither Morris’s base computer nor its “remote computer that ran the first object” are an “*end recipient device*” as that term is used in the ’045 patent. The end recipient device in the ’045 patent is separate from the device that includes the sensor agent and the global perspective information server. As shown in Figure 2, “[t]he computing device 204 includes and/or executes a sensor agent 208.” (EX-1001 at 7:10-11). “[T]he event line 244A-B may be sent to an end recipient 220 from the computing device 204 and/or the global perspective information server 216.” (*Id.* at 9:62-64).

Claim 10 requires that the sensor agent (in Figure 2, part of computing device 204) transmit the event line to the end recipient device.

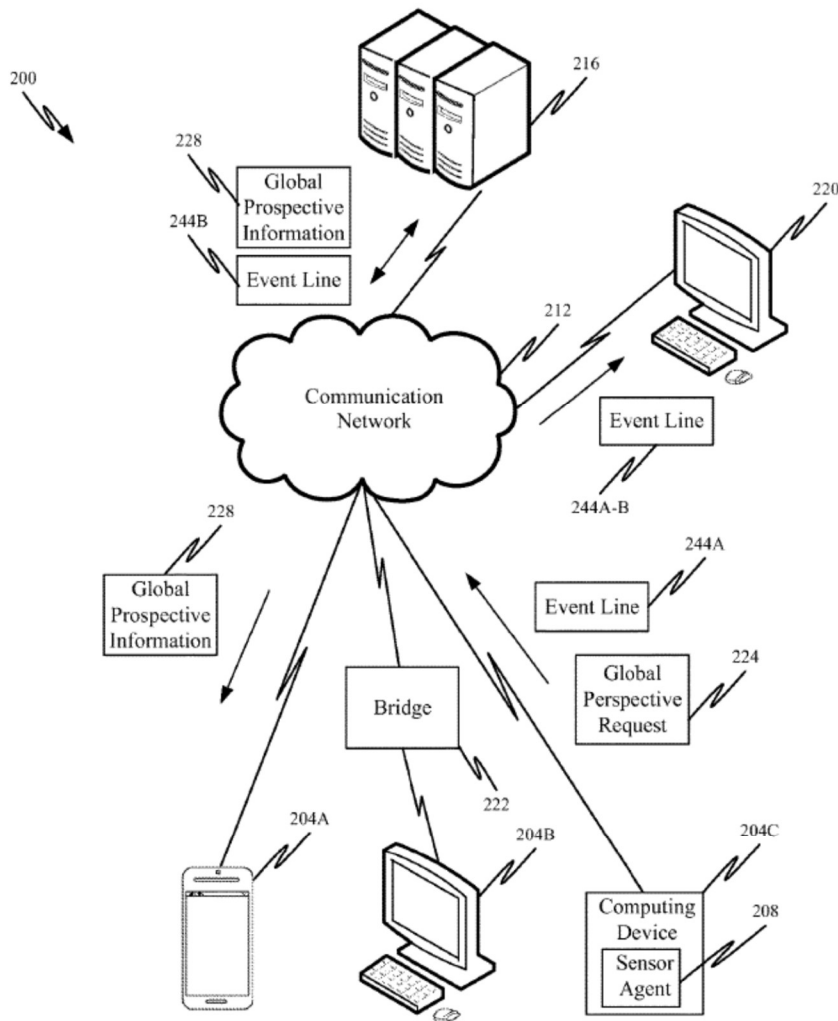


Figure 2

60. The specification repeatedly describes the end recipient device as a device that is separate from the device that includes the sensor agent and the global

perspective information server. (*See, e.g., id.* 6:46-50 (“In general, the system 200 includes one or more computing devices 204, each of which are interconnected to a global perspective information server 216 and end recipient 220 by one or more communication networks 212.”), 14:8-11 (“The bridge 222 may further include a communication interface 386 for communicating with another computing device 204, an end recipient 220, a global perspective information server 216, and/or the communication network 212.”), 14:48-51 (“The global perspective information server 216 may further include a communication interface 468 for communicating with another computing device 204, an end recipient 220, and/or the communication network 212.”)).⁷

61. The end recipient device being a separate device is further underscored by the context of the claim language. Claim 10, which is dependent from claim 8,

⁷ The end recipient device may include dashboards and other features that are not present in other portions of the system. (EX-1001 15:53-61; *see also id.* at 23:14-19 (“Fig. 13 generally illustrates exemplary user interfaces, or dashboards 1304, 1332, and 1336, which may be displayed at an end recipient 220 to a user at an user output device 412. The user interface generator 444 may create the one or more dashboards 1304, 1332, and 1335 to provide information from an event to a user of the end recipient device 220.”)).)

requires a sensor agent that is stored on a device, a global perspective information server, and an end recipient device. Claim 10 states that “the sensor agent transmits the event line to the end recipient device.” The sensor agent and the end recipient device cannot be on the same device or the event line would not be “transmit[ed]” from one device to the other. And, the claim requires both a global perspective information server and an end recipient device. As addressed above, the specification makes clear that these are separate devices.

62. Accordingly, “end recipient device” should be construed as “a device that is separate from the device that includes the sensor agent and the global perspective information server.”

C. Level Of Ordinary Skill In The Art

63. I understand that certain issues relating to the validity of the '045 patent must be judged from the perspective of a person of ordinary skill in the 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 of the invention as claimed was made. In deciding the level of 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.

64. A POSITA at the time of the invention of the '045 patent had a bachelor's degree in an accredited program of electrical engineering, computer engineering, or computer science, or a similar discipline, and have 2–3 years of practical work experience in the general fields of electrical engineering, computer science, networking, communications, and/or device and network security. More advanced degrees and/or training in a related discipline can compensate for shorter work experience.

65. I understand that Petitioner and its expert have defined the level of ordinary skill in the art as:

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 substitute for professional experience, or significant experience in the field could substitute for formal education.

(Petition 6; EX-1003 ¶33.)

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

67. Based on either standard, I was at least a person of ordinary skill in the art at the time of invention of the '045 patent.

VI. THE CHALLENGED CLAIMS ARE PATENTABLE.

68. Petitioner presented two grounds challenging the claims of the '045 patent:

Ground 1	Section 103(a)	Obvious over Morris or, alternatively, Morris in view of van Oorschot	Claims 1-4, 8-10, 12 and 16
Ground 2	Section 103(a)	Obvious over Capalik or, alternatively, Capalik in view of van Oorschot	Claims 1-4, 7-10, 12, 15-18

69. It is my opinion that Petitioner fails to establish that Morris, Morris in view of van Oorschot, Capalik, or Capalik in view of van Oorschot make obvious every limitation of the challenged claims.

70. Grounds 1 fails because (1) as to all challenged claims, the GenesisActor in Morris is not the “originating object” as that term is used in the '045 patent; (2) as to independent claims 8 and 16 and their dependent claims, Morris does not include a “sensor agent” that meets all of the required limitations; (3) as to dependent claim 10, Morris does not disclose the sensor agent transmitting the event line to an

end recipient device; and (4) as to dependent claim 3, van Oorschot does not disclose the claimed event priority.

71. Grounds 2 relies on Capalik, which fails to disclose the “across a network” limitation. The ’045 patent teaches obtaining information about computer objects operating across a network, *i.e.*, not on the same device. Capalik merely observes virtual machines running virtual operating systems on the same device. In its institution decision, the Board found:

Petitioner has not shown how the network limitation is met in Capalik. The discussion of this limitation in the Petition relies on general statements in Capalik that its methods “are able to follow a chain of unauthorized activity across multiple virtualized operating systems” and enable “identifying and thwarting advanced threats that involve multiple attackers spread across multiple systems,” but we fail to see how that teaches or suggests the recited limitation “wherein the global perspective for one or more related events to the at least one event across a network,” which according to Petitioner, requires a showing of “information about events across the network and that those events occur on ‘other devices’ on the network.” *See* Pet. 50. Petitioner does not explain how an ability to follow unauthorized activity relates to the claimed global perspective, and the cited language discusses tracking activity across multiple “operating systems,” not across a network.

* * *

[N]othing in the description of that step indicates that any of the information used relates an event related to the event across a network. *See* Ex. 1008 ¶¶ 75–96, 106. Instead, Capalik describes determining that activity is malicious if it performs unauthorized activities on the decoy network device, not because it is related to another event across a network. *See id.* ¶ 106 (“By identifying that the process A 308 performed unauthorized activities, the process A 308 is determined to be affected by the unauthorized activities, and the

association between the socket 306 and the process A308 is determined to be malicious (i.e., unauthorized).”). . . .

[A]s Patent Owner observes, the objects and targets all exist within one virtual machine (113–1). Petitioner does not explain how that teaches or suggests obtaining a global perspective that relates to one or more other events related to the event “across a network.”

* * *

On the record before us, we are not persuaded that Petitioner has shown sufficiently for purposes of this Decision that Capalik teaches or suggests the recited “one or more related events to the at least one event across a network” of claims 1 and 8, or the similarly recited “one or more related events to the one or more events across a network” of claim 16. Nor are we persuaded that Petitioner has sufficiently shown why an ordinarily skilled artisan would have considered modifying Capalik’s system to provide these limitations. . . .

In view of the foregoing, we determine that Petitioner has not demonstrated a reasonable likelihood of showing that independent claims 1, 8, and 16 would have been obvious over Capalik alone or in combination with van Oorshot.

(Paper 11 at 44-47.) Patent Owner agrees with the Board’s institution decision analysis of Capalik—which is correct and also consistent with the Examiner’s analysis allowing U.S. Patent No. 10, 257,224 (“the ’224 continuation patent”) to issue over the same prior art distinctions regarding the Kumar reference. The ’224 continuation patent resulted from a continuation of the same application as the ’045 patent. (*See* EX-2019.) Capalik fails to disclose the “across a network” limitation for this same reason. And for the same reasons that Ground 1 fails, Ground 2 fails to render

obvious claim 3's "event priority" because van Oorschot does not disclose the claimed event priority.

72. Accordingly, the Board should reject Petitioner's Grounds 1-2 and all challenged claims should be found patentable.

A. Overview of Morris

73. Morris relates to systems and methods for classifying computer objects as malware or as safe, and for assessing a remote computer's protection against malware. (EX-1005 [0001].)

74. Morris describes the deficiencies in the prior art as either "rely[ing] 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," or as using only "limited analysis of the operation of the particular object or its method of transmission to a computer is carried out to decide the likelihood of the object being malicious." (EX-1005 [0013].) The '045 patent applies a technique more similar to the former-described system: it relies on deeper analysis of objects at the local level. However, the '045 patent includes additional aspects to address the prior art deficiencies described by Morris by using additional techniques and resources, such as intelligent caching and processing on the remote computer to preserve local memory resources

and reduce transmissions of redundant events, in order to generate context for the event at the remote computer. (*See* EX-1001 2:66-3:58.)

75. To address the deficiencies in the prior art, at its base computer Morris “allows a comparison to be made between the [computer] objects and/or their effects on the different remote computers to determine whether or not a particular object should be classed as good or malware” using “pattern analysis.” (EX-1005 [0017].) This technique “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 [sic] malware and also avoids the need to generate new signatures to be used for signature matching as in the conventional prior art anti-virus software.” (EX-1005 [0017].) To perform this analysis, the base computer receives information about the computer object from the plural remote computers and then classifies the computer objects as malware or not based on this comparison. (EX-1005 [0017].)

76. For its comparison step, Morris uses information about the object such as “executable instructions contained within or constituted by the object; the size of the object; the current name of the object; the physical and folder location of the object on disk; 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 when the object is created, configured or runs on

the respective remote computers.” (EX-1005 [0018].) This information may be sent in the form of a key in order to reduce “the volume of data that needs to be transmitted to the base computer. Given that there may be thousands or even millions of connected remote computers and further given that each may send details about very many objects, this can be an important advantage.” (EX-1005 [0020].)

77. Based on this comparison, Morris identifies relationships between the computer object and other objects so that it may “immediately and automatically [] mark a child object as bad (or good) if the or [sic] a parent or other related object is based (or good)” so that “if at least one other object to which said object is related is classed as malware” then the object may be classed as malware. (EX-1005 [0019].)

B. Overview Of Capalik

78. Capalik is directed to “systems and method for protecting computer networks” that “analyz[e] malicious activities on a computer system in order to better protect the computer system from future malicious activity.”) To do this, Capalik generates a profile, or “fingerprint,” for a computer object after supervising its activities on a virtual machine. (EX-1008 Abstract; Claim 1.) Capalik teaches a method of monitoring activities on a virtual machine, whereby activities are identified using their source, target, and the association between the source and target. (EX-1008 [0010].) From this information, a fingerprint indicative of the activity on the virtual machine is created and is transmitted to identify the attack on other affected systems.

(EX-1008 [0010]; *see also* Figs. 6A–C (flowchart representing the method of identifying unauthorized activities according to Capalik).)

79. Capalik describes a decoy network device architecture. (EX-1008 [0016].) This is illustrated by Figure 2, which shows the host operating system on a decoy network device memory, running virtual machines that are running operating systems that are used to generate fingerprints. (EX-1008 [0033].)

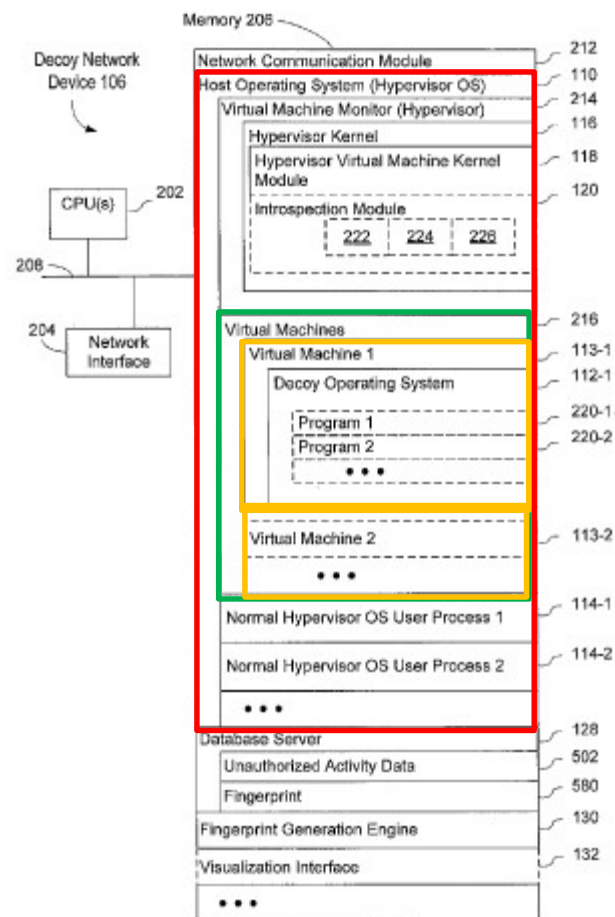


Figure 2

(EX-1008 Fig. 2.) (annotated)

Capalik's decoy network device 106 includes a CPU 106, a network communications interface 204, one or more network or other communications interfaces 204, and one or more communication buses 208 for interconnecting these components. (EX-1008 [0033].) No software is installed in the virtualized decoy operating system 112, which makes it "more difficult for the virtualized decoy operating system 112 (or an attacker who has gained access to the virtualized decoy operating system 112) to detect that its memory is being monitored." (EX-1008 [0047].)

80. Unlike malware identification and mitigation solutions that generate information from monitoring a computer object's activity on devices *across a network*, Capalik is directed to generating information about a computer object's activities through the creation of a decoy network using a decoy network device that runs multiple virtual machines that may include multiple decoy operating systems—all on *a single device*--to generate this "fingerprint." (See EX-1008 [0050]–[0053].) Capalik's decoy operating system 112 is virtualized and uses an introspection module for monitoring the activities of computer objects that runs on the same device. (EX-1008 [0051].) The virtual machines on which the virtualized operating systems run "are one or more software applications emulating one or more physical machines." (EX-1008 [0052].)

81. To create the decoy network, Capalik uses a "decoy network device" that "may include any number of virtualized decoy operating systems 112 or any

number of virtual machines” (EX-1008 [0053])—which, again, all run on a single device. (See EX-1008 [0050]–[0052].) Where Capalik teaches implementing the decoy network device on two distinct network devices, it still limits the decoy network device to a single device that is implemented on two distinct devices and that does not communicate information about computer objects across a network to generate a perspective for an event. (EX-1008 [0056] (“[T]he decoy network device 106 may be implemented on two distinct computing devices: a monitor device and an analyzer device. The monitor device includes the virtual machine monitor 214, the Virtual machines 216, and processes and modules therein. The analyzer device includes the database server 128, the fingerprint generation engine 130, and optionally the visualization interface 132. In such configuration, the monitor device collects unauthorized activity data, and the analyzer device analyzes the unauthorized activity data to generate fingerprints.”) Capalik communicates “raw attack information” to the “database server 128 as forensic data 236 for later analysis” in order to “generate a signature of the attack,” but Capalik does not disclose or teach using this for comparison with other computer objects, or to otherwise use this information to generate a perspective about an event based on other events across a network. (EX-1008 [0063].)

C. Overview of Van Oorschot

82. Van Oorschot relates to “a method for managing system events in a network.” (EX-1006 1:55-56.) Specifically, van Oorschot describes event management

tools that provide information regarding attributes of events that are initiated remotely or acted upon externally to a particular system. (EX-1006 1:44-51.)

83. Van Oorschot uses information about the attributes of an event for evaluation against a security policy in order to determine whether the event fails to conform with said security policy. (EX-1006 1:56–65.) The security policy against which the event information is evaluated “defines permitted actions for different used in accordance with the plurality of system event attributes.” (EX-1006 1:63–65.) “Significance factors” are then assigned to the data elements of the system event, and the system then determines whether the event “matches a special attention rule” that defines a prohibited action. (EX-1006 1:66-2:4.) These significance factors are then compared against a severity level for the system event based on whether the system event matches the special attention rule and whether the system event fails the security policy. (EX-1006 2:5-15.) Van Oorschot describes the types of attributes that may be used to configure a security policy, including “who,” “what,” “on what,” “where,” “where from,” and “where to” attributes. (EX-1006 7:7-32.) None of the attributes van Oorschot uses to create a security policy identify familial relationships between computer objects, such as determining the originating object of another computer object.

84. Van Oorschot’s event management tool may assign each system event a “severity level” based on exceptions or breaches during the comparison step. (EX-

1006 5:39–41.) The event management tool then reacts to the events based on the severity level of the breach. (EX-1006 5:41–49.) “For example, in the event of a severe breach, the event management tool may react by immedi-ately generating and transmitting an alert alarm or instant message to an ap-proprate security office.” (EX-1006 5:43-48.)

85. Van Oorschot’s process for assigning a severity level to an event is depicted by Figure 11:

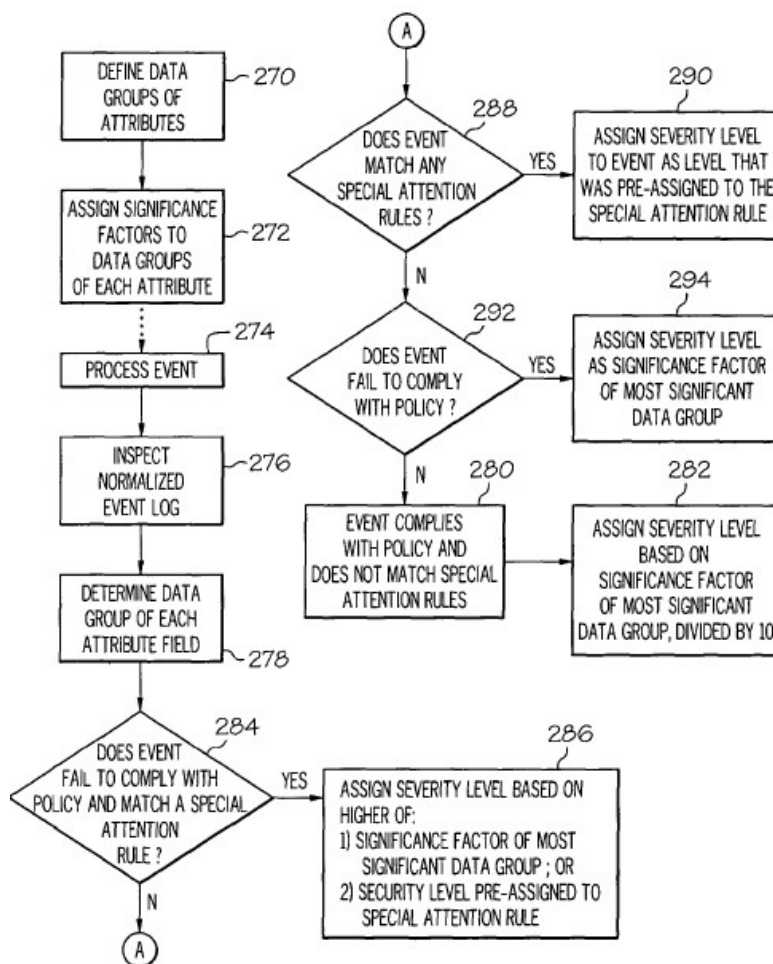


FIG. 11

(EX-1006 3:1-3, 8:12–10:2 (walkthrough of each element of Fig. 11); Fig. 11.)

86. The “data groups of the various attributes” for the event “are pre-defined 270, and significance factors are pre-assigned to the defined data groups 272.” (EX-1006 8:14–8:16.) During processing of an event 274, “the normalized event log data is inspected 276,⁸ and the data groups of each attribute field are determined 278.” (EX-1006 8:16-19.) The severity level is then assigned based on “whether the event complies with systems policies and according to whether any special attention rules are met.” (EX-1006 8:19-22.)

D. Petitioner Fails to Establish That Morris, Alone or in Combination, Renders Obvious Claims 1–4, 8-10, 12 and 16

87. Morris, whether alone or in combination with van Oorschot, does not render obvious any of claims 1-4, 8-10, 12, and 16. Specifically:

⁸ The “normalized event log” interprets event data “such that events generated at a variety of different client operating systems, applications, or network appliances, can be interpreted by the same event management tool.” (EX-1006 10:8-12.)

(1) Morris’s disclosure of GenesisActor information⁹ does not render obvious the “originating object” limitation in all challenged claims because the ’045 patent expressly distinguishes a GenesisActor from the claimed “originating object”;

(2) Morris does not render obvious the “sensor agent” limitation of independent claims 8 and 16 because Morris does not teach a single agent that meets all of the limitations recited in those claims;

(3) Morris does not disclose the sensor agent transmitting the event line to the end recipient device as required by dependent claim 10; and

⁹ The disclosure Petitioner relies on in Morris is entirely disclosed by Morris 340—and is cumulative. As Patent Owner explained in its POPR, the petition’s arguments regarding the “originating object” and GenesisActor were before the Examiner, and the Examiner allowed the ’224 continuation patent to issue. Paper 6 at 25-30. To find the claims unpatentable now, the PTAB necessarily must find that the Examiner had the relevant information during examination and erred in allowing the ’224 continuation patent. But the Examiner's decision to allow the claims of the ’224 continuation patent over the GenesisActor argument was correct for the reasons addressed herein.

(4) Morris/van Oorschot does not render obvious the event priority of claim 3 because van Oorschot’s “severity level” is not generated based on an indication of an originating object.

1. Morris Does Not Disclose Or Render Obvious The “Originating Object” Limitations

88. For Ground 1, Petitioner argues that Morris alone renders obvious the “originating object” limitations in all independent claims, i.e., 1(b), 8(e)(ii), and 16(g). (Petition 14-15, 32, 35.)

Limitation	Claim Language
1(b)	<i>“generating a contextual state for at least one of the one or more events by correlating the at least one event to an originating object”</i>
8(e)(ii)	<i>“sensor agent is operable to: ... generate a contextual state for at least one of the one or more events by correlating the at least one event to an originating object, the contextual state including an indication of the originating object of the first object and an indication of at least one of a device on which the first object is executed and a user associated with the first object”</i>

16(g)	“sensor agent is operable to: ... <i>generate a contextual state for at least one of the one or more events by correlating the at least one event to an originating object</i> ”
-------	--

Morris does not disclose or render obvious the “originating object” limitations for at least the reasons discussed below.

89. Petitioner contends that the “originating object” limitation is satisfied by Morris’s disclosure of GenesisActor information. (Petition 14-15 (“Morris discloses that additional contextual data correlated to the event is captured, including the ‘Genesisactor’ (i.e., originating object) which is the ‘key of an object that is not the direct Actor of an event but which is the ultimate parent of the event being performed. . . . Morris’s disclosure of correlating “Genesisactor” data with an event is consistent with the ’045 Patent’s disclosure of generating a contextual state.”) (emphasis and internal citation omitted).).

90. Contrary to Petitioner’s argument, Morris does not disclose or render obvious the claimed “originating object” because the ’045 patent explicitly distinguishes the GenesisActor from the claimed *originating object*. As discussed in Section V.B.1, *supra*, the ’045 patent explains that the *originating object* is the immediate parent of the current process/object and expressly distinguishes the *originating object* from the GenesisActor information. (See Section V.B.1, *supra*.) The Board’s

institution decision determined, on the then-existing record, that GenesisActor was “a species of the originating processes genus,” (Paper 11 at 26-28) but this is incorrect in view of the ’045 patent’s context. Since information about the **direct actor of an event** is **not** GenesisActor information, and the “originating object” is defined based on the “**current process and/or object**,” GenesisActor information is outside the scope of an “originating object” and cannot be a species of an “originating object” genus. (*See* Section V.B.1, *supra*.)

91. GenesisActor information is distinct from an “originating object,” and therefore cannot be a species of an “originating object” genus. (*See* Section III.B.1, *supra*.) The originating object refers to “a previous event that initiated or otherwise created the current process and/or object.” (EX-1001 17:41-42.) The GenesisActor, on the other hand, “is determined based on a chain of previous events.” (*Id.* at 17:42-43.) The “GenesisActor...*is not the direct Actor of an event* but...is the ultimate parent of the event being performed.” (EX-1001 8:42-48 (emphasis added).) Thus, the specification shows that the originating object is the object that directly created the first object. In contrast, the GenesisActor may be many steps removed from the first object through a chain of previous events. (EX-1001 17:43-44.) This distinction is significant because the GenesisActor may provide less helpful contextual state information than the originating object, which is more closely related to the first object, and thus provides a more complete audit trail for an event.

92. Petitioner does not address the distinction between originating object information and GenesisActor information. Nor does Petitioner identify any disclosure in the '045 patent or the prior art describing a “GenesisActor” as an originating object. Nor does Petitioner argue that “originating object” is a genus, or that “GenesisActor” is a species. Nor has Petitioner proposed a genus claim construction for “originating object” or otherwise explained why it should be construed as a genus that includes “GenesisActor” information, notwithstanding their mutually exclusive descriptions in the specification.

93. A POSITA would understand that originating object and GenesisActor information—because they are distinct—may each be uniquely important to detecting malware, such as in circumstances where the ultimate parent of an object does not provide contextual information indicating a threat but the originating object does. This is consistent with the '045 patent's discussion of enabling a complete audit trail for events. (*See* EX-1001 1:15-18; 12:55–63; 3:53-55 (“The present solution eliminates the guesswork and allows for end-to-end visibility of every event from the bottom of the stack to the top.”).) GenesisActor information is not a proxy for originating object information because jumping to the ultimate parent of an event would leave gaps in the audit trail between the ultimate parent and the event. This is contrary to the aims of the '045 patent and would not allow “an administrator, at an end

recipient 220 for example, to forensically track back any event which occurs to what triggered it.” (See EX-1001 12:55-63.)

94. Petitioner argues only that Morris discloses the claimed “originating object,” and not that Morris’s disclosure somehow teaches or otherwise renders obvious that limitation. (See Petition 14.) The fact that the ’045 patent expressly distinguishes GenesisActor information from the claimed “originating object” is therefore fatal to all of Petitioner’s Morris-based obviousness arguments.

95. In any event, there is no suggestion or teaching by Morris to modify Morris’s method to generate contextual state information by correlating an event to an originating object, nor does Petitioner identify a reason a POSITA would have been motivated to make such a modification. Although Morris discloses various types of metadata that might be captured and evaluated through its heuristic processes—including events (the actions or behaviors of an object acting upon another object); identities (the attributes of an object); GenesisActor (the ultimate parent of the event); ancillary data; and event checksums (EX-1005 [0104]-[0109])— *originating object* data is not among them. (See *generally* EX-1005.)

96. Thus, because Petitioner has not shown that Morris discloses or renders obvious the claimed “originating object,” Petitioner’s Morris-based Grounds 1 and 2 both fail.

2. Morris Does Not Disclose the “Sensor Agent” in Claims 8 and 16

97. Independent claims 8 and 16 require a “sensor agent” that is “stored on the data storage” of a “device.” (EX-1001 cls. 8, 16 (“a device including . . . a sensor agent stored on the data storage...”) The “sensor agent” must be operable to meet certain limitations, i.e., gather an event, generate a contextual state, obtain a global perspective, assemble an event line, and transmit the assembled event line. Petitioner fails to show that Morris discloses a “sensor agent” that meets these limitations.

Limitation	Claim Language
8(e)	<i>“a sensor agent stored on the data storage that is executable by the processor, wherein the sensor agent is operable to: gather one or more events... generate a contextual state for at least one of the one or more events by correlating the at least one event to an originating object... obtain a global perspective for the at least one event by obtaining information associated with... assemble an event line... and transmit the assembled event line....”</i>

16(e)	<i>“a sensor agent stored on the data storage that is executable by the processor, wherein the sensor agent is operable to:</i> <i>gather one or more events . . .</i> <i>generate a contextual state for at least one of the one or more events...</i> <i>assemble an event line including details associated with the at least one event, the details including . . . global perspective information comprising information about one or more related events to the one or more events across a network...</i> <i>and transmit the assembled event line....”</i>
-------	---

98. Both claims 8 and 16 recite “a sensor agent...wherein the sensor agent is operable to” meet the recited limitations. I understand that courts regularly hold that this phrasing requires a single instance of the element be capable of performing all the recited functionality, in view of the context of the claims and patent. Here, the context of the claims of the ‘045 patent requires that a single “sensor agent” must be operable to meet all of the recited limitations.

99. Petitioner expressly relies on the “local ‘agent’ software” on Morris’s “remote computer 2” as the “sensor agent.” (Petition 32 for limitation [8(e)] requiring “a sensor agent stored on the data storage . . . ,” arguing “*Supra*, [8(c)-(d)]

(describing *remote computer 2* configured *with “local ‘agent’ software”* for effectuating the described methods and the processor for executing the functions of the “local ‘agent’ software”); *see also* Petition at 30-32 (addressing limitations 8(a)-(e), 35 (addressing limitations 16(a)-(e); EX-2025 186:3-6 (“So what I’m saying is that the local agent software on the remote computer is a sensor agent.”) ¹⁰.)

100. However, Petitioner (a) does not explain how the “local ‘agent’ software” on Morris’s “remote computer 2” is operable to perform the limitations required of the “sensor agent”; and (b) appears to rely on devices other than Morris’s “local ‘agent’ software” on Morris’s “remote computer 2”—such as Morris’s base computer—for certain limitations that must be met by the “sensor agent.” Petitioner relies on vague contentions, or appears to split the requirements of the “sensor agent” across multiple devices to meet its burden of proving unpatentability.

101. Petitioner presents its arguments regarding claims 8 and 16 (which are system claims) by cross-referencing its arguments regarding claim 1 (which is a method claim). (Petition 32, 36.) Claim 1 does not require that a specific device perform the steps of the claimed method. However, as addressed above, claims 8 and

¹⁰ Petitioner does not oppose Patent Owner’s use of Dr. Lee’s deposition testimony taken in IPR2023-00126, which proceeding is directed to the related ’224 continuation patent. (EX-2027.)

16 require a “sensor agent” that is stored on a “device” to be operable to meet certain limitations. The petition glosses over these differences by arguing that the “sensor agent” limitations “are substantively identical to the method steps of Claim 1 and are rendered obvious for the same reasons detailed in Claim 1.” (*See* Petition 32; *see also id.* 36 n.6.)

102. As addressed in greater detail in connection with specific limitations below, the cross-referenced portions of the petition that address the method steps of claim 1 do not clearly identify Petitioner’s contentions as to which device is operable to perform the system limitations required in claims 8 and 16.

103. Second, Petitioner appears to rely on multiple devices in connection with the limitations that must be met by the “sensor agent.” At times, the petition relies on Morris’s remote computer. But, at other times the petition relies on Morris’s base computer or some combination of the remote computer and base computer. Again, Petitioner improperly split the required limitations of the “sensor agent” across multiple devices, which is inconsistent with the context of the ’045 patent. The following are several examples of the deficiencies in Petitioner’s contentions.

104. **Claims 8 and 16 state that the “sensor agent” must “generate a contextual state for at least one of the one or more events by correlating the at least one event to an originating object.”** The petition does not address this limitation in connection with claims 8 and 16. Instead, the petition refers to its discussion of

claim 1.¹¹ (Petition at 32, 35-36.) In connection with the claim 1 element “generating a contextual state for at least one of the one or more events by correlating the at least one event to an originating object,” the petition does not identify any device that performs that step. (Petition 14-15.) For example, the petition does not reference Morris’s remote computer, base computer, or any other device discussed in Morris. That alone means that the petition fails to show that Morris discloses this limitation.

105. The petition cites to paragraphs 82-85 of Dr. Lee’s declaration in connection with its discussion of claim element 1(b). However, these paragraphs also do not clearly identify any device that performs the required limitation. Dr. Lee states: “On top of the event metadata associated with running an object at one or more remote computers, *Morris* describes additional types of context generated from an event that is sent to the base computer to further inform analytical processing techniques.” (EX-1003 ¶82). With this statement, Dr. Lee appears to suggest that “correlating the event to an originating object of the first object” occurs at Morris’s base computer through “analytical processing techniques.” But, when asked on cross-examination to clarify where in Morris he contends the contextual state is

¹¹ Notably, the multiple levels of cross-referenced contentions confused Petitioner’s expert even though his declaration followed the same approach. (EX-2025 190:2-191:4 (“I am starting to get confused by all these numbers.”).)

generated, Dr. Lee was unable to provide a clear answer. (EX-2025 190:2-194:14.)

The following exchange is representative:

Q. So, in Morris, is the contextual state information generated at the remote computer, the base computer, or somewhere else?

A. So as I said here -- yeah. I talk about Morris describes additional type of contact generated from an event that is sent to the base computer. So some of this information, obviously you get it from the base computer.

Q. Okay. Let me ask the question one more time to try to get a clear record. In Morris, is the contextual state information generated at the remote computer, the base computer, or somewhere else?

A. I -- let me see here. So the -- the base -- the remote computer sends contextual information to the remote -- to the base computer, that's for sure. So meaning that the remote computer, whatever it does, as we describe here in the declaration, meets the limitation. It gathers contextual information, metadata, and so on, and sends the remote -- and sends to the base computer.

(EX-2025 192:4-193:1.)¹²

106. In sum, the petition does not identify any device in Morris that includes a sensor agent operable to generate contextual state information for the event by correlating the event to an originating object of the first object. Although Petitioner contends that the “sensor agent” is located at Morris’s remote computer, neither the petition nor Dr. Lee provide any credible testimony that Morris’s remote computer correlates an event to an originating object. Furthermore, Dr. Lee’s testimony supports that Morris’s base computer, not its remote computer, performs this operation.

107. **Claim 8 states that the “sensor agent” must “obtain a global perspective,” and Claim 16 states that the “sensor agent” must assemble an event line that includes “global perspective information.”** As an initial matter, the petition does not contend that “obtain a global perspective” in claim 8 should be

¹² The excerpts from Dr. Lee’s deposition cited herein are indicative of numerous exchanges in which Dr. Lee either could not (or refused to) provide direct answers to basic questions regarding his opinions. Instead, Dr. Lee frequently read non-responsive excerpts from his declaration into the record without responding to the question asked. (*See, e.g.*, EX-2025 167:12-180:11, 204:2-206:2.) Patent Owner asks that the Board take Dr. Lee’s evasive cross examination testimony into account when weighing the credibility of his opinions in this case.

interpreted differently than the process by which the “global perspective” is obtained for purposes of assembling an event line for claim 16. Again, the petition does not address these limitations in connection with claims 8 or 16. Instead, the petition refers to its discussion of claim 1. (Petition 32 (“The remaining limitations of Claim 8 are substantively identical to method steps of Claim 1.”), *id.* 35-36 (citing back to claims 1 and 8 as support for this limitation of claim 16).)

108. In connection with the claim 1 element “obtaining a global perspective for the at least one event,” the petition identifies Morris’s base computer (not its remote computer) as performing the step of “obtaining a global perspective for the event by obtaining information concerning a determination as to whether the first object/originating object is malware.” (Petition 22.) Petitioner further states that Morris’s “[b]ase computer 3 compares the data about the computer object received from the plural computers and classif[ies] the computer object as malware on the basis of said comparison.” (Petition 21 (emphasis added, quotations omitted)). In other words, Petitioner argues that Morris discloses obtaining a global perspective by determining whether an object is malware and that Morris makes that determination at the base computer.

109. The arguments in the petition align with Dr. Lee’s declaration, which states that Morris discloses obtaining a global perspective by determining whether a first or originating object is malware based on familial relationships (EX-1003

¶¶101-106 (emphasis added)) and that Morris discloses “*comparing in the base computer* the data about the computer object received from the plural computers; and *classifying the computer object as malware on the basis of said comparison.*” (EX-1003 ¶101 (emphasis added)). When asked about his opinion on cross examination, Dr. Lee did not offer any credible testimony to the contrary. The following exchange is representative:

Q. All right. So in paragraph 88,¹³ are you saying that it’s the base computer that’s determining the global perspective?

MR. HORTON: Object to form.

A. You have to talk about how the -- in paragraph 88 when I talk about explicitly -- specifically how the base computer, based on the information received from the remote computer, you know, as highlighted text says “classifying the computer object as malware on the basis of said comparison” and also determining whether a particular object should be classified as malware.

So, basically, this here lists what kind of comparison analysis you would do on the base computer. So...

¹³ This question references paragraph 88 of Dr. Lee’s declaration in IPR2023-00126, which corresponds to paragraph 101 of Dr. Lee’s declaration in tis case.

BY MR. BLYTHE: Q. Okay.

A. But again, like I said, you know, at the end, you know, the local computer get updated with all this information, and my Figure 2 shows -- it teaches -- he actually does, you know, try to obtain that, try to draw a conclusion based on the local search. Again, based on global information. So that's why it's gaining a global perspective. It doesn't mean that global computer -- I mean, the remote computer performs data with everybody else, no. That's base computer does that. But -- yeah.

(EX-2025 202:19-204:1.)

110. In sum, although the petition contends that the “sensor agent” is located at Morris’s remote computer, the petition does not include any contention that Morris’s remote computer obtains or determines a global perspective. On the contrary, the petition points to Morris’s base computer, not its remote computer, in connection with this operation.

111. Claims 8 and 16 state that the “sensor agent” must “gather one or more events.” Again, the petition does not address these limitations in connection with claims 8 or 16. Instead, the petition refers to its discussion of claim 1. (Petition 32 (“remaining limitations...are substantively identical to method steps of Claim “), id. 36.) In connection with the claim 1 element “gathering one or more events defining

an action of a first object acting on a target,” the petition states: “Morris discloses that ‘the base computer’ gathers ‘events initiated by or involving the object when the object is created, configured or runs on the respective remote computers.’ Morris, [0018].” (Petition 12 (emphasis added).) The petition goes on to discuss a combination of both the base computer and remote computers in connection with this limitation. (Id. 12-14.) Dr. Lee similarly refers to a combination of Morris’s base computer and remote computers in connection with this limitation. (EX-1003 ¶¶77-81.) And, Dr. Lee’s cross examination testimony provided no clarity on what device in Morris he believes discloses gathering an event. (EX-2025 186:7-190:1.) Petitioner cannot rely on an ambiguous combination of the base computer and remote computers for the limitation requiring a “sensor agent” operable to “gather an event.”

112. Because Petitioner fails to identify a single sensor agent on a device that is operable to perform the required limitations 8(e)(i)-(v) and 16(f)-(i), claims 8 and 16, and their challenged dependent claims 9, 10, and 12 should be held patentable.

3. Morris Does Not Render Obvious a Sensor Agent Transmitting the Event Line to the End Recipient Device

113. Morris does not disclose “the sensor agent transmit[ting] the event line to the end recipient device” as required by Claim 10. Petitioner’s argument fails for two reasons: (1) Morris’s “base computer” that aggregates information about

computer networks across the monitored network is not an end recipient device; and
(2) copying information about object signatures is not the same as transmitting an event line.

Limitation	Claim Language
10	“The system of claim 8, further comprising: a global perspective information server; and an end recipient device, wherein the information associated with the one or more of the first object and the originating object is obtained from the global perspective information server and the sensor agent transmits the event line to the end recipient device.”

114. Petitioner argues that the “end recipient device” is Morris’s base computer. (Petition 35 (“Morris discloses its ‘agent software’ running on remote computer 2 (i.e., sensor agent) transmits the key comprising the event and contextual data (i.e., event line) to base computer 3 (i.e., end recipient device) to perform the above-described malware analysis.”).) However, Petitioner’s argument is based on an improper construction of “end recipient device” under which the “end recipient device” and “global perspective information server” (which is also required by claim 10) are the same device. (Petition 33 (“A PHOSITA would have understood base

computer 3 in conjunction with database 7 constitutes the global perspective information server and base computer 3 constitutes the end recipient device.”.)

115. The ’045 patent distinguishes the “end recipient device” from the device that includes the sensor agent and the global perspective information server. (Section III.B.2, *supra*.) Petitioner’s argument that the base computer is an “end recipient device” is grounded on nothing more than conclusory statements in the petition (Petition 33-35) and Dr. Lee’s declaration (EX-1003 ¶¶ 150-153.) Thus, the petition fails to show that Morris’s “base computer” discloses or renders obvious the claimed “end recipient device.”

Petitioner argues that Morris’s remote computer may also be an “end recipient device.” (Petition 24, 35.) In particular, Petitioner argues that Morris’s “key” (which Petitioner says is the “event line” in the ’045 patent) can be “‘copied to’ the local database of a remote computer that first ran the object.” (Petition 24, citing Morris [0084].) However, this argument fails to account for the actual language of the claim limitation. Claim 10 of the ’045 patent requires that “the *sensor agent* transmits the event line to the end recipient device.” Under Petitioner’s argument, the “key” is copied to the remote computer from the base computer. (EX-2025 211:18-212:6.) And, as addressed above, Petitioner contends that the “sensor agent” is “local ‘agent’ software” on Morris’s “remote computer 2.” (Petition 30-32 (addressing limitations

8(a)-(e), 35 (addressing limitations 16(a)-(e); EX-2025 186:3-6).) Any copy of a “key” coming from the base computer to the remote computer is not being transmitted by the “sensor agent.”

116. Petitioner’s argument that the “key” is copied back to “the local database of a remote computer that first ran the object” would mean that the device that includes the sensor agent (i.e., the remote computer) and the end recipient device are the same. As addressed above, the ‘045 patent distinguishes the “end recipient device” from the device that includes the sensor agent and the global perspective information server. (Section III.B.2, *supra*.) The petition does not include any argument that Morris discloses transmitting an event line from one remote computer to a different remote computer. And, on cross examination, Dr. Lee was unable to identify any disclosure in Morris that suggested transmitting an event line from one remote computer to another. (EX-2025 212:7-213:22.)

117. Because Petitioner fails to show that Morris’s base computer discloses or renders obvious the claimed “end recipient device,” or that Morris discloses or renders obvious a remote computer transmitting an event line to another remote computer, claim 10 is patentable.

**4. Morris in View of van Oorschot Does Not Render Obvious
“Obtaining an Event Priority” in Claim 3**

118. Morris in view of van Oorschot does not render obvious the claim 3 limitation “obtaining an event priority based on one or more of the generated context and the global perspective for the at least one event.” Petitioner concedes that Morris does not include an event priority. (Petition 27.) Petitioner argues that van Oorschot’s disclosure of a “severity level” is an event priority based on “generated context.” (Petition 27.) However, Van Oorschot’s “severity level” cannot be the claimed event priority because it is not based on generated contextual state information that includes an indication of the originating object, as required by the claims.

Limitation	Claim Language
3	<i>“obtaining an event priority based on one or more of the generated contextual state information and the global perspective, wherein the event line further includes the event priority”</i>

119. Petitioner argues that van Oorschot renders obvious an event priority based on the generated contextual state. Contextual state information must include correlating an event to an originating object and an indication of the originating object. (EX-1001 cl.1; *see* Section IV.D.1, *supra*.) In particular, independent claim 1 of the ’045 patent, from which claim 3 depends, requires that the contextual state be

generated by “correlating the at least one event to an originating object,” and the contextual state must include “an indication of the originating object of the first object.” (EX-1001 cl. 1; *see* Section IV.D.1, *supra*.) Van Oorschot’s “severity level” is not based on information that is indicative of, or correlated with, an originating object.

120. Van Oorschot assigns “event data and contextual data associated with an event” into groups that are used to define system rules regarding permitted and unpermitted actions. (Petition 27-28; EX-1006 7:7-36.) According to Petitioner, these rules disclose the “generated contextual state.” (Petition 28 (“These rules focus on permitted (or abnormal) activity related to certain attributes, such as ‘who’ (e.g., ‘only ‘system administrators’ are permitted to modify files’) or ‘where’ (e.g., ‘types of clients at which the events can occur’)—i.e., the generated contextual state discussed above in claim 1.”).) But Petitioner does not identify any disclosure from van Oorschot of a contextual state that identifies an originating object. (See Petition 27-28; cf. EX-1006 7:7-57; *see also* EX-1006 8:12-10:1; EX-1003 ¶¶ 125-130.) The portions of van Oorschot cited by Petitioner are directed to generating policy rules that define a system’s permitted actions (EX-1006 7:7-57), which has nothing to do with indicating an originating object.

121. Van Oorschot discusses policy rules based on criteria relating to, for example, who may access or modify a file. (EX-1006 7:7-57) But van Oorschot does not discuss any indication of whether a file or process is an originating object. (Compare EX-1006 7:7-57 with EX-1001 39-42 (“[T]he originating process or object may [be] determined based on a previous event that initiated or otherwise created the current process and/or object.”) Petitioner does not argue that POSITA would find the ’045 patent’s event priority obvious in view of van Oorschot’s policy rules. Accordingly, Petitioner fails to show that Morris/van Oorschot renders obvious the event priority required by claim 3.

E. Petitioner Fails To Establish That Capalik, Alone or in Combination, Renders Obvious Claims 1–4, 7–10, 12, 15-18

122. Petitioner’s Ground 2—based on Capalik, alone or in combination with van Oorschot—does not render claims 1-4, 7-10, 12, and 15-18 obvious because Capalik does not teach or suggest “wherein the global perspective relates to one or more other events related to the event across a network.” (See also Paper 11 at 44 (“We find that Petitioner has not shown how the network limitation is met in Capalik.”).) Petitioner resurrects and relies entirely on an argument that the Examiner considered and ultimately (correctly) rejected during prosecution of the ’224 continuation patent (EX-2019 293–94), and Petitioner fails to provide an adequate reason for revisiting the Patent Office’s determination that “[t]here is simply no teaching of

events relating to application integrity for other devices on a network” from a prior art reference directed to “the integrity of a single application running on a single device.” (*Id.*; *see generally* Petition 48-49.)

123. Further, for the same reasons that Ground 1 fails, Ground 2 fails to render obvious claim 3’s “event priority” because van Oorschot does not disclose the claimed event priority.

124. Petitioner challenges claims 7, 15, 17, and 18 only under Ground 2 (Petition 7) so those claims are patentable if the Board finds, consistent with its institution decision, that Capalik does not meet the network limitation.

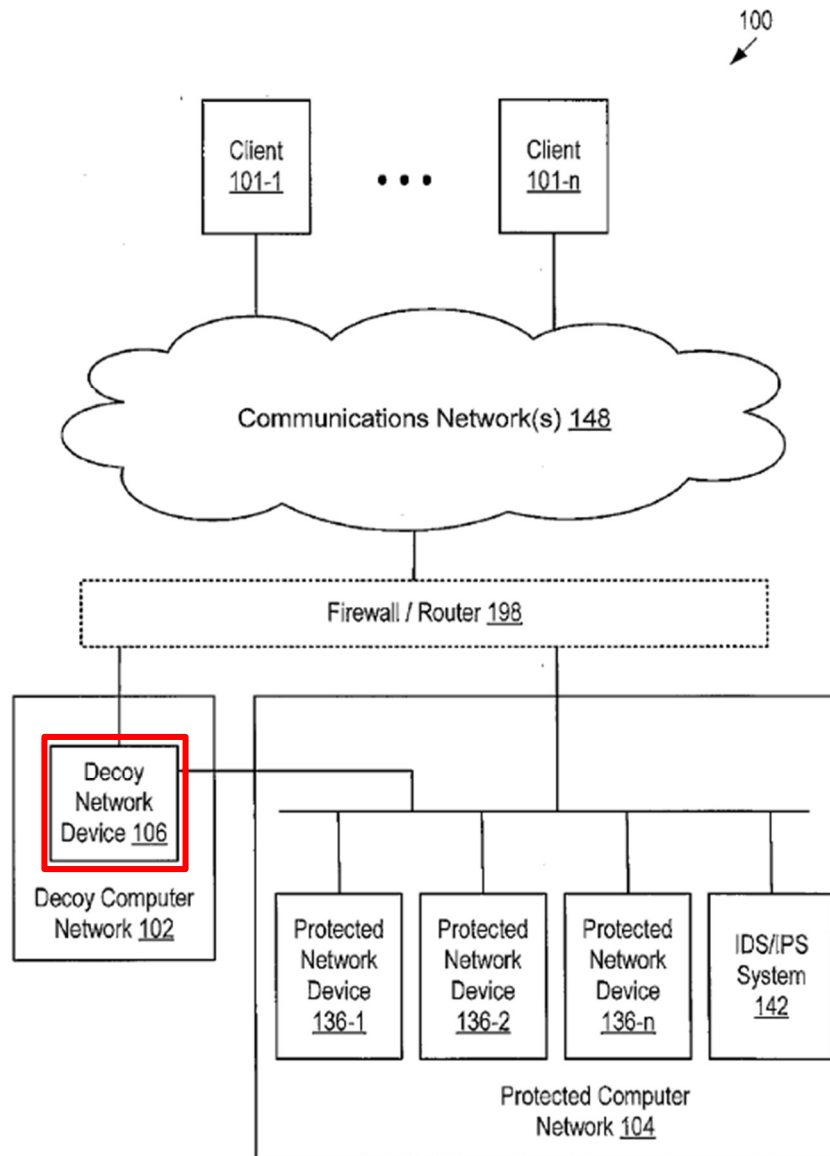
1. Capalik Does Not Disclose Or Render Obvious A Global Perspective Related to One or More Other Events related to the Event Across a Network

125. Claims 1, 8, and 16 of the ’045 patent include a global perspective about one or more related events to the at least one event across a network. (EX-1001 cls. 1, 8, 16.) Petitioner contends that Capalik can follow a chain of un-authorized activity across multiple virtualized operating systems, and that this behavior discloses this element. (Petition 50.) But in Capalik all of the virtualized operating systems are *on the same device*, so there is no “one or more other events related to the event *across a network*.” (EX-1008 [0045]-[0046].)

Limitation	Claim Language
1[(c)(i)] ¹⁴	<i>“wherein the global perspective for one or more related events to the at least one event across a network”</i>
8[(e)(iii)]	<i>“...wherein the global perspective for one or more related events to the at least one event across a network”</i>
16[h]	<i>“...global perspective information comprising information about one or more related events to the one or more events across a network”</i>

126. Capalik discloses a system and method implemented via a decoy network device—i.e., a decoy system that is monitored to collect fingerprint data of unauthorized activities. (EX-1008 [0030], Fig. 1.)

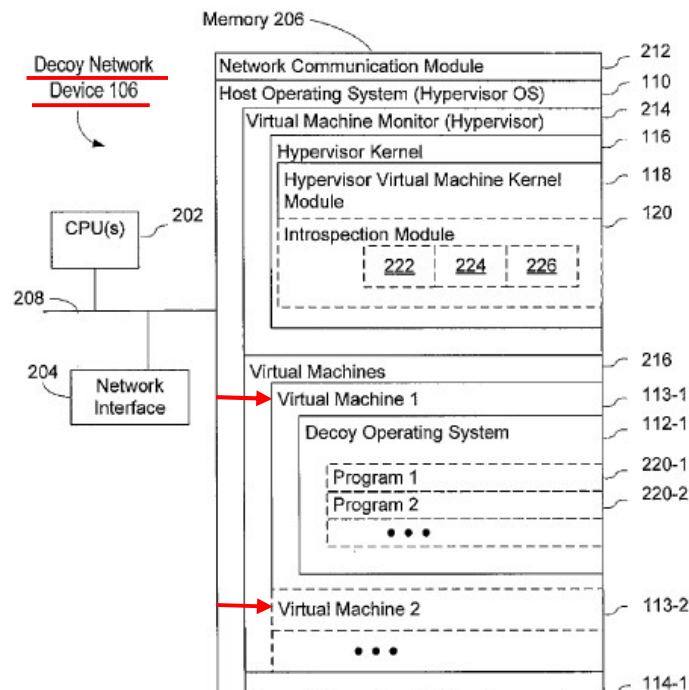
¹⁴ The Court construed this limitation of claim 1 as *“wherein the global perspective [is] for one or more related events to the at least one event across [other devices on] a network.”* (EX-2024 9.)



(EX-1008 Fig. 1 (annotated).) The decoy network device utilizes a virtual machine—i.e., one or more software applications *emulating* one or more physical machines—to establish virtual or decoy operating systems to attract attacker activity. (EX-1008 [0052]-[0053].)

127. Capalik teaches “[a] computer implemented method [that] includes monitoring activities on the virtual machine. A plurality of activities being

performed at the virtual machine is identified.” (EX-1008 Abstract.) Figure 2 of Capalik illustrates that, the virtual machines run decoy operating systems and programs in order to collect information about events occurring on the decoy network device.



(EX-1008 Fig. 2) (excerpted and annotated).

That is, while Capalik teaches that its method is “able to follow a chain of unauthorized activity across multiple virtualized operating systems,” these virtualized operating systems reside on the *same single decoy network device*. (EX-1008 [0045]-[0046]; cf. EX-2019 293 (“Kumar is focused on the integrity of a single application running on a single device. There is simply no teaching of events relating to application integrity for other devices on a network. Therefore, Kumar cannot possibly teach or suggest a ‘global perspective’ that ‘relates to one or more other events

related to the event across a network.”).) Thus, the decoy network device of Capalik on which the virtual machines run is not capable of performing this limitation because the global perspective cannot “relate[] to one or more other events related to the event *across a network*.”

128. Petitioner also argues that “Capalik discloses obtaining global perspective information in a scenario based on related events from multiple data streams across a network, shown in annotated Figure 4E.” (Petition 52.) Petitioner is wrong. Figure 4E illustrates virtual machine 113-1. (EX-1008 [0074].) But again, the events Petitioner cites all occur on one virtual machine (and one decoy network device) and *not across a network*.

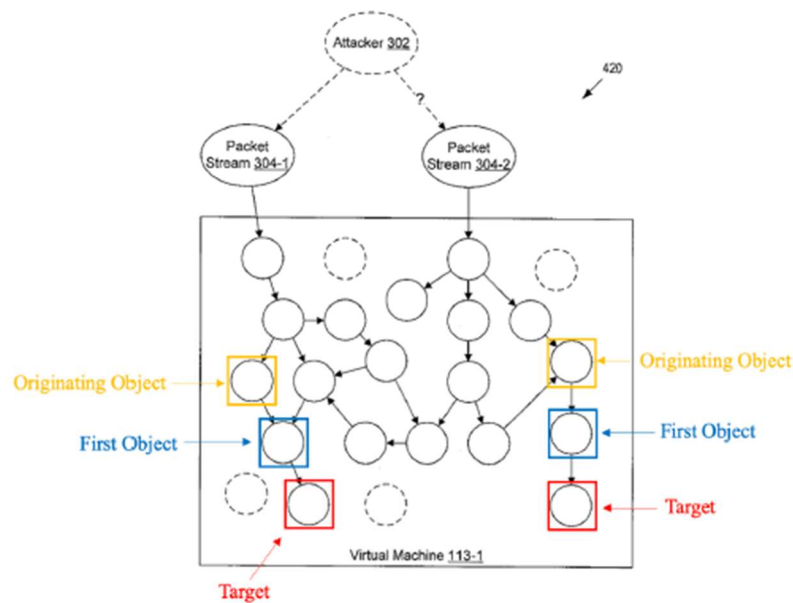


Figure 4E

(Petition 52; EX-1008 Figure 4E (annotated).)

129. Petitioner further argues that “a POSITA would have understood that in an attack across multiple virtualized operating systems, Capalik’s methods would follow the related chain of unauthorized activity across each operating system and would perform the methods identified above for obtaining a global perspective—namely, identifying an activity source and the activity target (e.g., by ‘an Inter-net Protocol address’) and determining ‘association types’ between sources and targets, including ‘malicious associations.’” (Petition 50-51 (emphasis added).) But the claim limitation is not met by following unauthorized activity across operating systems.

130. The claims requires relating events across a **network**, and a network is not the same as virtualized operating systems on a single device. Indeed, the ’045 patent illustrates distinct computing devices 204 on a network. (EX-1001 Fig. 1.) Petitioner does not explain why activity across virtualized **operating systems** on a single device would be understood by POSITA as equivalent to activity across a **network**. Capalik thus does not teach or suggest a “global perspective” that “*relates to one or more other events related to the event across a network.*”

131. Petitioner’s argument regarding Capalik repeats a reading of Capalik 570 that the Examiner considered and ultimately rejected. During prosecution, the Examiner rejected claim 1 in view of Kumar. (EX-2019 293.) To overcome this rejection, the applicant argued the following:

It is illogical to conclude that such a simplistic mapping of alerts to a specific device is analogous to the claimed “global perspective” that is “based on the contextual state information” and “relates to one or more other events related to the event across a network.” Rather, Kumar is focused on the integrity of a single application running on a single device. **There is simply no teaching of events relating to application integrity for other devices on a network.** Therefore, Kumar cannot possibly teach or suggest a “global perspective” that “relates to one or more other events related across a network.”

(EX-2019 293 (emphasis added).) Petitioner’s Capalik-based argument for the limitation “*wherein the global perspective relates to one or more events related to the event across a network*” is the same single-device argument the Examiner considered for Kumar during prosecution, and over which the Examiner ultimately allowed the ’045 patent.

132. Petitioner does not argue that a POSITA would have been motivated to modify Capalik’s teachings for this limitation, nor would a POSITA have been motivated to do so. Even if one were to utilize multiple decoy network devices, Petitioner does not identify any Capalik teaching or embodiment that includes sharing information across decoy computer network devices. To the contrary, Capalik teaches a self-contained decoy network device on which virtualized operating

systems are run, and on which computer objects activities are monitored. (EX-1008 Fig. 2.)

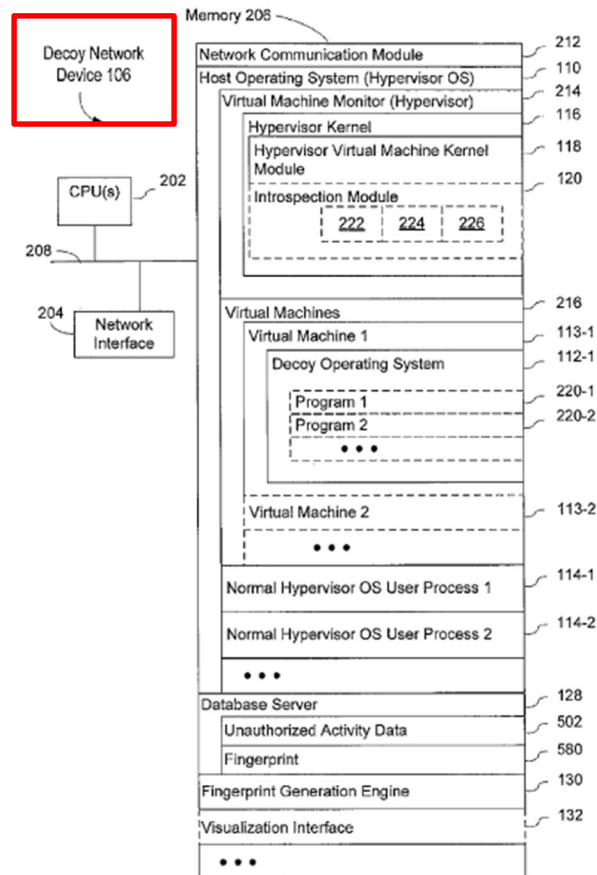


Figure 2

(EX-1030 Fig. 2 (annotated to highlight Capalik’s decoy network device, on which multiple virtual machines run decoy operating systems.) Capalik does not teach transmitting this information between decoy network devices, or the use of this information from multiple decoy network devices to create a fingerprint. (See EX-1008 [0126]–[0128].) Capalik does not instruct POSITA on how the computer object fingerprint it generates may be used by other devices on a network—only that it can

be used. (*See* EX-1008 [0126] (“The decoy network device transmits (644) the fingerprint to one or more protected network devices 136 to prevent future attacks that comprise the same or similar activities as indicated by the fingerprint.”).) Capalik does not instruct a POSITA how to modify computer systems on a network to use the fingerprint it generates. (*See* EX-1008 [0126]–[0128].) Rather, “[t]he decoy network device creates (636) the fingerprint indicative of the activity on the virtual machine from the stored activities.” (EX-1008 at [0120]; *see also id.*, [0126]–[0128].)

133. Accordingly, it is my opinion that Petitioner fails to show that Capalik discloses or renders obvious the “*wherein the global perspective relates to one or more events related to the event across a network*” limitation claimed recited in the challenged ’045 patent claims.

2. Capalik in View of van Oorschot Does Not Render Obvious Claim 3’s “Event Priority” Because van Oorschot’s “Severity Level” Does Not Include an Indication of any Originating Object

As discussed above in Section IV.D.4, *supra*, van Oorschot’s severity level does not include the claimed ‘generated context’ because it is not based on any correlation using an originating object.

As for Ground 1, for Ground 2 Petitioner relies on van Oorschot alone as disclosing “‘assign[ing] a severity level’ to ‘system events’ based on the generated context to inform the manner in which an event is processed.” (Petition 57.) Thus,

Petitioner’s argument for “modifying Capalik’s malware detection methods to incorporate Van Oorschot’s known event priority severity level assignment” fails. (*See* Petition 57.) Even if the severity level from van Oorschot were incorporated in Capalik, that severity level would not be based on the event’s “generated context”—there is no originating object used to generate context for the event. (*See* Section IV.D.4, *supra*.)

VII. PETITIONER’S GROUNDS FAIL TO ANTICIPATE OR RENDER OBVIOUS ALL DEPENDENT CLAIMS.

134. Because Petitioner has failed to establish that any of its grounds anticipate or render obvious all limitations of independent claims 1, 8, and 16, Petitioner cannot make a showing that such grounds anticipate or render obvious dependent claims 2–4, 7-10, 15, and 17-18.

VIII. SECONDARY OBJECTIVE INDICIA OF NONOBVIOUSNESS.

135. In determining obviousness of the claimed invention, I understand that objective evidence of secondary considerations pertaining to nonobviousness must be considered, including long-felt but unsolved needs, failure of others, commercial success, etc. 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.

136. Despite the evidence regarding the long-felt, unmet need for the invention and commercial success in Patent Owner's district court complaint against Petitioner, Petitioner fails to address or consider any such evidence in its petition; thus, institution should be denied. (EX-2015 ¶¶72–82.)

137. As discussed above, a long-felt and persistent need existed to improve network forensics methods before the invention of the '045 patent. (*See* Section IV, *supra*.) Prior attempts to develop reliable network security were limited in their ability to establish any sort of context or obtain useful information from within the host or system, making them insufficient to identify and address the threats faced by modern computer networks. (*See id.*) Where others failed, the inventors of the '045 patent succeeded. (*See id.*)

138. I understand that Patent Owner currently accuses at least four industry participants, including Petitioner, of infringing the technology claimed in the '045 patent, demonstrating the critical need and widespread industry reliance on these novel methods. (*See* EX-2003; EX-2016; EX-2017; EX-2018.) Petitioner, in particular, markets and provides the CrowdStrike Falcon Platform, the components of which practice the system and methods claimed in the '045 patent. (*See* EX-2018.)

IX. CONCLUSION

For the foregoing reasons, it is my opinion that Petitioner's grounds do not anticipate or render obvious any of the challenged claims of the '045 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 7th day of September, 2023 in Los Angeles, California.

Respectfully submitted,



Nenad Medvidovic, Ph.D.