

Filed on behalf of: Orca Security Ltd.

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

WIZ, INC.,
Petitioner,

v.

ORCA SECURITY LTD.,
Patent Owner.

Case IPR2024-00863
Patent 11,663,031

**DECLARATION OF DR. DAVID KAEI
IN SUPPORT OF
PATENT OWNER'S RESPONSE**

Table of Contents

I.	Introduction.....	3
II.	Qualifications.....	4
III.	Summary Of Materials Reviewed And Considered	7
IV.	Legal Standards.....	8
V.	Level Of Skill In The Art And Perspective Applied In This Declaration.....	12
VI.	Background.....	15
A.	Overview of the '031 Patent.....	15
B.	Claim Construction.....	22
1.	“(i) Taking at Least One Snapshot, and (ii) Requesting Taking at Least One Snapshot of the Virtual Machine at Rest”	22
2.	“Analyzing the At Least One Snapshot”	27
VII.	Overview of the Asserted References.....	30
A.	Overview of Veselov	30
B.	Overview of Price	37
C.	Overview of Hufsmith	43
VIII.	The Claims Of The '031 Patent Would Not Have Been Obvious ...	54
A.	Ground 1: Claims 1, 3-9, and 11-16 Would Not Have Been Obvious Over Veselov and Price.....	54
1.	A Person of Ordinary Skill in the Art Would Not Have Been Motivated to Combine Veselov and Price and Would Not Have Had a Reasonable Expectation of Success in Doing So.....	55
2.	Even If Combined, Veselov and Price Do Not Disclose or Suggest Multiple Limitations in the Independent Claims.....	79
B.	Ground 2: Claims 2 and 10 Would Not Have Been Obvious Over Veselov, Price, and Hufsmith	105

1.	A Person of Ordinary Skill in the Art Would Not Have Been Motivated to Combine Hufsmith with Veselov and Price and Would Not Have Had a Reasonable Expectation of Success in Doing So	105
2.	Even If Combined, Veselov, Price, and Hufsmith Do Not Disclose or Suggest Multiple Limitations in Claims 2 and 10	110
C.	Ground 3: Claims 6 and 14 Would Not Have Been Obvious Over Veselov, Price, Hufsmith, and Huseinović.....	114
IX.	Objective Indicia of Non-obviousness.....	115
A.	Orca's SideScanning™ Uses The '031 Patent Inventions ..	116
B.	Long-Felt, Unresolved Need	116
C.	Orca's claimed inventions overcame industry skepticism to become widely accepted--and widely praised—in the industry.	122
D.	Extensive Industry Praise Supports Non-Obviousness of the Claims.....	127
E.	Evidence Shows Wiz Copied Orca's SideScanning™.....	134
1.	Evidence shows Wiz had access to Orca's invention	136
2.	Evidence shows Wiz's Workload Scanning is substantially similar to Orca's SideScanning™	153
F.	Commercial Success.....	177
X.	CONCLUSION.....	180

I. Introduction

1. I, David Kaeli, submit this declaration on behalf of Orca Security Ltd. (“Patent Owner”) to provide my opinions concerning the validity of claims 1-16 of U.S. Patent No. 11,663,031 (the “’031 patent,” Ex.1001), in support of Patent Owner’s Response to the Petition for *Inter Partes* review of the ’031 patent, which I understand was filed by Wiz, Inc. (“Petitioner”) and accompanied by a Declaration of Dr. Angelos Stavrou (the “Stavrou Declaration”), among other exhibits. I further understand that on December 9, 2024, *Inter Partes* review (IPR) was instituted (Paper 8).

2. I have previously provided my opinions in a Declaration (Ex.2001) in support of the Patent Owner’s Preliminary Response to the Petition. Although I believe that declaration is accurate in all respects, this new Declaration represents my considered opinion and supersedes my previous Declaration.

3. I am more than eighteen years of age, and I am a citizen of the United States, currently residing in the Commonwealth of Massachusetts.

4. I am being compensated for the time I spent preparing this Declaration at my customary rate. The opinions I express in this Declaration are my own. My compensation does not depend upon the outcome of this IPR, the contents of this Declaration, or any testimony I may provide in this IPR.

5. This Declaration summarizes the opinions I have formed to date. I reserve the right to modify or update my opinions, if necessary, based on my review and analysis of information or materials that I may receive after this Declaration is submitted. I further reserve the right to respond to any new or updated positions Wiz or its expert(s) may take in this IPR.

II. Qualifications

6. In formulating my opinions, I have relied upon my knowledge, training, and experience in the relevant art. My qualifications, along with a more detailed summary of my background and publications, are stated more fully in my *curriculum vitae*, which has been provided as Exhibit 2002. Here, I provide a brief summary of my qualifications.

7. I am currently the Director of Northeastern University's Computer Architecture Research Laboratory and a Distinguished Professor of Electrical and Computer Engineering. I have been a faculty member at Northeastern since September 1993, in the Electrical and Computer Engineering Department.

8. For over thirty years, I have pursued research across a broad range of computer architectures and technologies, including memory systems, storage systems, virtualization, and secure computing systems. I have published over 400 refereed journal and conference papers in the areas of computer architecture and computer security. I also have supervised over forty Ph.D. theses in these areas.

9. I am a Fellow of the Institute of Electrical and Electronic Engineers (IEEE) and a Fellow of the Association of Computing Machinery (ACM). I have served two terms as the Chair of the Executive Committee of the IEEE Technical Committee on Computer Architecture. I am serving in my second terms as the Editor-in-Chief of the ACM Transactions on Architecture and Code Optimization. I am also an Associate Editor for Elsevier's Journal of Future Generation Computer Systems.

10. From 2011-2014, I served as Chief Science Advisor of Apposable Software. From 2007-2011, I served as Chief Technology Officer of NUIC Technologies, where I worked on designing secure virtual appliances. From 2005-2011, I consulted for Akorri Networks, where, among other things, I helped develop intellectual property relating to storage virtualization. In 2000, I served as a consultant for InCert Corporation.

11. Before I joined Northeastern's faculty in 1993, I spent twelve years working for IBM. From 1981-1986, I worked on the design and delivery of the IBM System/370 3090 mainframe system. From 1986-1993, I worked at the IBM T.J. Watson Research Center, designing, and analyzing IBM Personal Systems 2 computers and associated software. During that time, I also worked on the design of a new Application System/400 multi-processor system.

12. I am an active educator and researcher and have extensive experience in the broader field of computer architecture. I am a co-inventor on patents targeting virtualization security. *See, e.g.,* Moffie, et al., *VMM-Based Intrusion Detection System*, (US Patent 8,719,936, May 6, 2014). I am a co-inventor on patents relating to virtual machine management. *See, e.g.,* Smirnov, et al., *Systems and Methods for Determining Placement of Virtual Machines*, (US Patent 8,099,487, January 17, 2012). And, I have conducted academic research on, and published multiple papers relating to, hardware and software security, including papers discussing security on virtualized infrastructure, computer system vulnerabilities and protections, and monitoring virtualized computing environments. *See, e.g.,* Azmandian et al., *Securing Virtualized Execution Environments Through Machine Learning-based Intrusion Detection*, IEEE International Workshop on Machine Learning for Signal Processing (2015); Azmandian et al., *Virtual Machine Monitor-based Lightweight Intrusion Detection*, ACM SIGOPS Operating Systems Review (2011); Sabbagh et al., *Overdrive Fault Attacks on GPU*, IEEE/IFIP International Conference on Dependable Systems (2021); Ahn et al., *Trident: A Hybrid Correlation-Collison GPU Cache Timing Attack for AES Key Recovery*, IEEE International Symposium on High Performance Computer Architecture (2024); Arzuaga, et al., *Increasing Power/Performance Resource Efficiency on Virtualized Enterprise Servers*, IEEE

Computing Frontiers (2011); Azmandian, et al., *Workload Characterization at the Virtualization Layer*, IEEE Annual International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems, (2011).

13. I earned a Bachelor of Science degree in Electrical Engineering from Rutgers University in 1981. I earned a Master of Science degree in Computer Engineering from Syracuse University in 1985. I earned a Ph.D. in Electrical Engineering from Rutgers University in 1992.

III. Summary Of Materials Reviewed And Considered

14. The opinions contained in this Declaration are based on the documents I reviewed and my knowledge and professional judgment. In forming the opinions expressed in this Declaration, while drawing on my experience, I reviewed all of the documents cited in this declaration, as well as the Petition and the Institution Decision.

15. My opinions are additionally guided by my appreciation of how a person of ordinary skill in the art would have understood the claims of the '031 Patent at the time of the invention, which I have been asked to assume is January 28, 2019. I will sometimes refer to this time as the "relevant time."

IV. Legal Standards

16. I am not an attorney. However, I have been informed by counsel of certain legal standards and have applied those standards, described below, in formulating my opinions.

17. I understand that the claims of a patent define the limits of a patentee's exclusive rights. I understand that to determine the scope of the claimed invention, courts may construe claim terms, the meaning of which the parties may dispute. I understand that claim terms should generally be given their ordinary and customary meaning as understood by a person of ordinary skill in the art at the time of filing of the patent application.

18. I understand that claims must be construed in light of and consistent with the intrinsic evidence. In this context, I understand that intrinsic evidence includes the claims themselves, the written disclosure in the specification, and the patent's prosecution history. I understand that the specification is always highly relevant to the claim construction analysis and often is the single best guide to the meaning of a disputed term. I understand that extrinsic evidence (i.e., evidence that is not part of the patent itself or its prosecution history) may also be considered when construing claims and may include, for example, technical dictionaries, technical publications and books, treatises, and expert testimony, though the intrinsic record will be more persuasive as to the meaning of a particular term.

19. I understand that one or more prior art references can render a patent claim obvious to a person of ordinary skill in the art if the differences between the subject matter set forth in the patent claim and the prior art are such that the subject matter of the claim would have been obvious at the time the claimed invention was made.

20. I understand that for a reference to qualify as a relevant prior art reference in an obvious analysis, the reference must be “analogous”—meaning it is from the same field of endeavor as the patented technology, and if not, then the reference must be reasonably pertinent to the particular problem facing the inventors of the patent at issue. I also understand that a reference is reasonably pertinent if a person of ordinary skill in the art would have reasonably consulted those references and applied their teachings in seeking a solution to the problem facing the inventors.

21. In analyzing obviousness, I understand that it is important to consider the scope of the claims, the level of skill of a person of ordinary skill in the relevant art, the scope and content of the prior art, and the differences between the prior art and the claims. I understand that claimed subject matter can be obvious when a person of ordinary skill in the art would have been motivated to combine pre-existing disclosures in the prior art, and that combination yields no more than what a person of ordinary skill in the art would have expected. However, I further

understand that obviousness cannot be based on the hindsight combination of components selectively culled from the prior art.

22. I also understand from counsel that to guide against hindsight, obviousness must be considered in view of any evidence demonstrating certain objective factors that may show that an invention was not obvious from the prior art, even if the differences between the prior art and the claimed invention seem slight. I have been informed by counsel that “objective indicia” of non-obviousness, can include, for example, evidence of commercial success of the invention, evidence of a long-felt need that was solved by an invention, evidence that others copied an invention, evidence that an invention achieved an unexpected result, evidence of industry praise of an invention, evidence of skepticism by others, evidence of licensing an invention, or evidence that others failed to solve the same problem that an invention solved.

23. I have been informed that in order to have relevance and weight in an obviousness analysis, the patentee must show that there is a “nexus” between the objective indicia and the merits of the claimed inventions. I understand that the focus of the nexus requirement looks to the patent claims as a whole, and that nexus does *not* have to be based only on limitations that are allegedly not shown in a primary prior art reference. For example, even if every limitation of a claim is shown to be *individually* known in the prior art, that does not preclude nexus where

the objective indicia evidence is tied to the combination of features as a whole and the combination was not previously known.

24. I have been informed that commercial success is relevant as an indication of nonobviousness because the law presumes an idea would successfully have been brought to market sooner, in response to market forces, had the idea been obvious to persons skilled in the art. Thus, the law deems evidence of (1) commercial success, and (2) some causal relation or “nexus” between an invention and commercial success of a product embodying that invention, probative of whether an invention was nonobvious.

25. Similarly, I have been informed that recognition by others of skill in the art that the inventor’s contribution was an invention may support a finding of nonobviousness. This is because the law recognizes that industry participants, especially competitors, are not likely to praise an obvious advance over the known art. Accordingly, the fact that a product which embodies the patent claims garnered awards or otherwise received praise will weigh against an assertion that the same claim would have been obvious.

26. I understand that copying of the claimed inventions can also support an inference of nonobviousness, because it can both show a type of “praising” of the invention and suggesting that the prior art was insufficient. I have been informed that, like all other objective indicia, copying requires a showing of a

nexus between the alleged copying and the claimed invention. I also understand that not every competing product that arguably falls within the scope of a patent is evidence of copying, because otherwise every infringement suit would automatically confirm the nonobviousness of the patent. Rather, a nexus can be shown for copying based on, for example, that a competitor had access to the patentee's product or a disclosed embodiment in the patent and then created a product having substantial similarity to the patented product.

V. Level Of Skill In The Art And Perspective Applied In This Declaration

27. I understand that certain issues relating to validity must be judged from the perspective of a person of ordinary skill in the art. I further understand that factors that may be considered in determining the level of skill of a person of ordinary skill in the art include, but are not limited to, (1) the type of problems encountered in the art; (2) prior art solutions to those problems; (3) rapidity with which innovations are made; (4) sophistication of the technology; and (5) the educational level of active workers in the field.

28. I have been informed and understand that a person of ordinary skill in the art is a hypothetical person who is presumed to be aware of all pertinent prior art, thinks along conventional wisdom in the art, and is a person of ordinary creativity. Importantly, this hypothetical person is of ordinary skill in the art to which the claimed subject matter pertains.

29. With respect to the '031 Patent, a person of ordinary skill in the art at the time of the earliest priority date, January 28, 2019, would have been familiar with techniques for securing virtual cloud assets against cyber threats, and underlying virtualization technologies used with such techniques.

30. A person of ordinary skill in the art would have had at least a Bachelor's degree in computer science, computer engineering, or a related field, and two years of industry experience or academic research experience in cyber security and virtualization, including in cloud computing cybersecurity. In my view, more education can supplement relevant experience and vice versa.

31. At the time of the earliest priority date for the '031 Patent (January 28, 2019), I met the qualifications to be a person of ordinary skill in the art, and I believe that I also have a sufficient level of knowledge, experience, and education to provide expert opinions in the field of the '031 patent.

32. While my own level of skill exceeded that of the ordinary level of skill as of January 2019, I am well-acquainted with the actual performance and capabilities of a person of ordinary skill in the art as defined above. For example, during the relevant timeframe, I was supervising and evaluating the performance of such individuals, such as my graduate students who were performing research in this area. Furthermore, my participation in professional organizations and in

conferences and other events exposes me to the state of the industry and the level of experience of individuals who, like I did, had at least the ordinary level of skill.

33. I have applied the above-defined perspective of a person of ordinary skill in the art, as of January 2019, in forming and expressing my opinions. Unless otherwise stated, my testimony below refers to the knowledge of one of ordinary skill in the art at that relevant time.

34. In the Stavrou Declaration (Ex.1002), Dr. Stavrou sets out the following articulation of the level of ordinary skill:

In my opinion, a person of ordinary skill in the art (“POSA”) at the relevant time would have had (1) at least a bachelor’s degree in computer science, computer engineering, electrical engineering, or a related degree, and (2) would also have 2-3 years of professional experience with cyber security analysis and virtualization. Such experience includes, for example, malware analysis, security analysis of cloud computing systems, and security analysis of virtual machines. ... Additional experience can compensate for less education and vice versa.

Ex.1002 (Stavrou Decl.), ¶22. At the time of the earliest priority date for the ’031 patent (January 28, 2019), I met Dr. Stavrou’s proposed qualifications to be a person of ordinary skill in the art, and my opinions in this Declaration would not change if the Board were to adopt and apply that proposed level of ordinary skill.

VI. Background

A. Overview of the '031 Patent

35. The '031 Patent is titled “Techniques for Securing Virtual Cloud Assets at Rest Against Cyber Threats” and relates to systems and methods for securing virtual cloud assets, such as virtual machines (“VMs”) while the virtual machines are at rest. Ex.1001 ('031 Patent), Cover.

36. The '031 Patent explains that, around the time of its priority date (January 2019), public cloud service providers, such as Amazon, Microsoft, and Google, had “increasingly adapted their applications to be run from multiple cloud computing platforms,” in which “[v]irtualization,” which “allow[s] multiple applications and users to share the same cloud computing infrastructure,” “plays a key role.” Ex.1001 ('031 patent), 1:20-27. The '031 patent also provides context for “virtualization” and explains that “virtualization can be achieved by means of virtual machines,” each of which can “emulate[] a number of ‘computers’ or instances, all within a single physical device.” Ex.1001 ('031 patent), 1:28-35.

37. The '031 patent discusses problems that existed with trends in the cybersecurity field. For example, the '031 patent discusses “applications and virtual machines running on top of virtualization technologies are [] vulnerable to some cyber threats,” (e.g., from “vulnerable software applications or infected operating systems.”). Ex.1001 ('031 patent), 1:39-43. And, the '031 patent also

explains ways the industry had tried to deal with such issues. For example, the '031 patent discusses "traffic inspection," which is "[t]raditionally ... performed by a network device connected between a client and a server ... hosting virtual machines"; using "network scanner[s] deployed out of path" to query servers to determine if applications possessing security threats are executed; "traffic monitor[ing]," which "listens to traffic flows between clients and [a] server"; and using "vulnerability management and security assessment solutions ... based on agents installed in each server in a cloud computing platform or data center." Ex.1001 ('031 patent), 1:44-2:14. As the '031 patent also explains, though, these methods suffered from disadvantages. "Traffic inspection may not provide an accurate indication of the security status of [a] server due to inherent limitations, such as encryption" and lack of exposure to pertinent data. Ex.1001 ('031 patent), 1:49-52. With network scanners deployed out of path, servers "may not respond" or "expose [] necessary data," and network configuration- and credential-related issues can be problematic. Ex.1001 ('031 patent), 1:57-64. Traffic monitoring "detect[s] threats only based on the monitored traffic" and thus can miss server "misconfiguration[s]" and "vulnerabilities in software executed by the server." Ex.1001 ('031 patent), 2:1-6. And "[u]sing agents is a cumbersome solution for a number of reasons, including IT resource management, governance, and

performance.” Ex.1001 ('031 patent), 2:11-13. “For example, installing agents in a large data center may take months.” Ex.1001 ('031 patent), 2:14-15.

38. The '031 patent also recognizes that prior methods for protecting cloud computing infrastructures have not adequately addressed the “detection of vulnerabilities in data at rest,” i.e., inactive data, and the security of VMs at rest, i.e., VMs that are “powered off.” Ex.1001 ('031 patent), 2:15-26, 3:53-55. With prior methods, VMs at rest “[we]re not inspected for vulnerabilities, simply because scanners and/or installed monitoring agents cannot operate on a powered-off machine,” or only past, potentially out-of-date information for such machines was scanned. Ex.1001 ('031 patent), 2:25-32. Therefore, prior approaches did not sufficiently address “constantly evolv[ing]” threats or “undetected vulnerabilities [that] can pose cyber threats” “when data or a machine at rest becomes active.” Ex.1001 ('031 patent), 2:33-36.

39. The '031 patent aims to solve such problems. Ex.1001 ('031 patent), 2:37-55. Figure 1B of the '031 patent, reproduced below, illustrates a diagram of a cloud computing platform to which the '031 patent's techniques for “scan[ning] for embedded vulnerabilities in machines at rest,” i.e., “when the machine is powered off” and “inactive,” can apply.

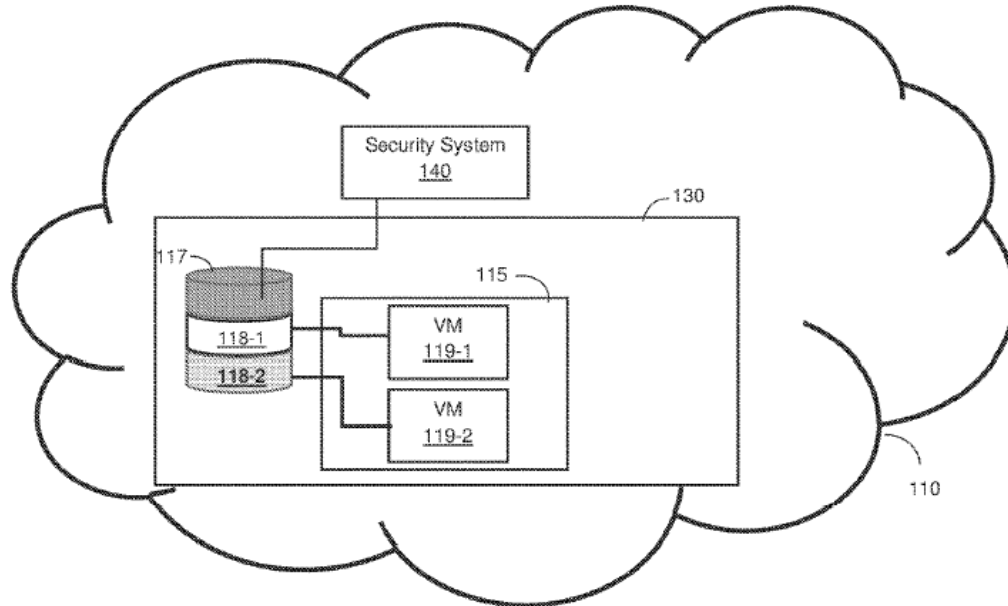


FIG. 1B

Ex.1001 ('031 patent), 3:24-25, 3:53-59, Fig. 1B.

40. As an initial matter, the '031 patent distinguishes between suspended and inactive states. Ex.1001 ('031 patent), 3:45-47 (“Various techniques disclosed herein include techniques for securing data at rest or machines at rest (collectively referred to as ‘machines at rest’). “Data at rest” can include, for example, “inactive data” stored physically in any digital form. Ex.1001 ('031 patent), 2:18-28, 3:47-48. “Machines at rest may include a virtual machine configured service(s) and/or application(s) that are inactive but can be accessed or executed as needed. The applications and/or services in such machines at rest are infrequently executed.” Ex.1001 ('031 patent), 2:18-28, 3:48-53. The '031 patent’s systems and methods are implemented to scan for embedded vulnerabilities in machines at rest (i.e., when the machine is powered off). Ex.1001 ('031 patent), 2:18-28, 3:53-55. As

an example, a machine at rest may be utilized during one month of the year and may remain otherwise remain inactive during the year. Ex.1001 ('031 patent), 2:18-28, 3:55-57. “According to the disclosed embodiments, the machine is scanned for vulnerabilities when it is in its inactive step.” Ex.1001 ('031 patent), 2:18-28, 3:57-59.

41. The cloud computing platform of the '031 patent includes a server (shown as no. 115 in Figure 1B) that hosts one or more VMs (shown as nos. 119-1, 119-2 in Figure 1B) in the cloud computing platform. Ex.1001 ('031 patent), Fig. 1B, 3:60-4:19. The cloud computing platform also has storage (shown as no. 117 in Figure 1B) which acts as storage space for the platform's server and “emulates” virtual disks (shown as nos. 118-1, 118-2 in Figure 1B) for the VMs. Ex.1001 ('031 patent), Fig. 1B, 3:60-4:19. The storage may be part of the server or, as shown in Figure 1B, separated from the server. Ex.1001 ('031 patent), Fig. 1B, 4:21-25. In the example shown in Figure 1B, a “high-speed connection” such as an “optical fiber” connects the storage with the server. Ex.1001 ('031 patent), Fig. 1B, 4:21-25. Figure 1B also shows a client environment (shown as no. 130 in Figure 1B) which interfaces with a security system (shown as no. 140 in Figure 1B) using “APIs” or through “services provided by the cloud computing platform” (not shown in Figure 1B). Ex.1001 ('031 patent), Fig. 1B, 4:51-54. The security system is configured to perform the steps of the '031 patent's invention and may

be separate from the client environment, as shown in Figure 1B, or part of the client environment. Ex.1001 ('031 patent), Fig. 1B, 4:37-50. The challenged claims of the '031 patent demonstrate the embodiment where the security system interfaces with the client environment through APIs. Ex.1001 ('031 patent), cls. 1, 9, 10.

42. The security system of the '031 patent interfaces with the client environment so that it can perform scanning operations and detect any “vulnerable software, non-secure configurations, exploitation attempts, compromised assets, data leaks, data mining” and other risks or vulnerabilities. Ex.1001 ('031 patent), 5:24-30. For example, the security system can detect vulnerable code (Ex.1001 ('031 patent), 6:35-36), cyber threats that do not represent vulnerabilities such as unencrypted sensitive data (Ex.1001 ('031 patent), 7:21-32), and changes in sensitive machine areas such as changed application files without installation (Ex.1001 ('031 patent), 7:33-39). The security system is also configured to provide “incident response, anti-ransomware, and cyber-insurance” through access to the system’s security posture (Ex.1001 ('031 patent), 5:21-30), generate reports on detected vulnerabilities and cyber threats (Ex.1001 ('031 patent), 7:40-55), and mitigate detected vulnerabilities and cyber threats. Ex.1001 ('031 patent), 7:61-8:6. Mitigation actions may include, for example, disabling a VM from execution, updating a VM with patches, blocking traffic from untrusted networks,

quarantining an infected VM, or preventing a VM from launching. Ex.1001 ('031 patent), 8:60-9:14.

43. In order to accomplish the scanning and detection operations I discuss above, the '031 patent provides a series of steps. First, the security system queries for and receives an address of a virtual disk to be assessed. Ex.1001 ('031 patent), 4:57-67. For example, the security system queries a cloud management console for the address of virtual disks that serve the VMs. Ex.1001 ('031 patent), 5:31-35. The virtual disk location may be determined before moving to steps involving snapshots. Ex.1001 ('031 patent), 8:37-44. After the security system receives the location, the system may take or request the taking of a snapshot. Ex.1001 ('031 patent), 5:31-6:16. The snapshot might be a copy of the VM's virtual disk or a copy of the VM's disk file at a point in time such as when the VM is at rest. Ex.1001 ('031 patent), 5:31-6:16, 8:42-44, cls. 1, 9, 10. The security system then analyzes the snapshot to detect vulnerabilities and cyber threats. Ex.1001 ('031 patent), 6:17-22, 8:45-59. The snapshot analysis does not involve interaction with or information from the VM and does not require agents installed in the system server or VM. Ex.1001 ('031 patent), 6:17-22. The '031 patent provides multiple techniques for analyzing snapshots to detect different types of vulnerabilities and cyber threats. Ex.1001 ('031 patent), 6:23-7:39. For example, the snapshot may be parsed and analyzed by the security system, the system may match installed

application lists to a known list of vulnerable applications, and the system may check configuration files of applications. Ex.1001 ('031 patent), 6:23-6:61. The security service then generates reports on detected vulnerabilities and cyber threats. '031 patent, 7:40-55. These reports are reported “as alerts” and may be filtered or prioritized based in part on the determined risk level of the VM. Ex.1001 ('031 patent), 7:40-55, 8:60-9:7.

B. Claim Construction

44. In addition to the explicit constructions discussed below, I have applied the plain and ordinary meanings of the claim terms in the '031 patent, as a person of ordinary skill in the art would have understood them in the context of that patent in approximately January 2019.

1. “(i) Taking at Least One Snapshot, and (ii) Requesting Taking at Least One Snapshot of the Virtual Machine at Rest”

45. Independent claims 1 and 16 recite “perform at least one of: (i) taking at least one snapshot, and (ii) requesting taking at least one snapshot of the virtual machine at rest.” Independent claim 9 similarly recites “performing at least one of: (i) taking at least one snapshot, and (ii) requesting taking at least one snapshot of the virtual machine at rest.” Dr. Stavrou labels these as elements 1.5, 9.6, and 16.6. Ex.1002 (Stavrou Decl.), p. 121.

46. Dr. Stavrou does not appear to take a definitive position as to the meaning of this claim limitation. However, Dr. Stavrou addresses this limitation in Ground 1 based on an interpretation where “(i) taking at least one snapshot” does not have to be “of the virtual machine at rest.” *See* Ex.1002 (Stavrou Decl.), ¶147 (“Veselov teaches performing ‘(i) taking at least one snapshot.’”). Then, Dr. Stavrou asserts that “[t]o the extent Patent Owner argues that ‘of the virtual machine at rest’ modifies clause (i) such that it requires taking at least one snapshot of a virtual machine at rest, in my opinion this would have been obvious further in view of Budko.”¹ Thus, in his analysis, Dr. Stavrou treats the first clause of these limitations (with romanette (i)) as though it does not require a virtual machine “at rest” (*see, e.g.*, Ex.1002 (Stavrou Decl.), ¶147). I disagree with that interpretation because that is not how a person of ordinary skill in the art would understand “perform[ing] at least one of: (i) taking at least one snapshot, and (ii) requesting taking at least one snapshot of the virtual machine at rest.”

47. It is my opinion that the plain meaning of these elements require that “of the virtual machine at rest” modifies both “(i) taking at least one snapshot” and

¹ Dr. Stavrou stated in deposition that this reference to Budko was a typo and was intended to refer to Price. Ex.2077 (Stavrou Dep.) 352:14-353:10.

“(ii) requesting taking at least one snapshot.” The claim language, the specification, and the prosecution history inform this opinion.

48. First, the claim language informs a person of ordinary skill in the art that “of the virtual machine at rest” modifies both clauses. When the claim recites taking a snapshot, a person of ordinary skill in the art would ask: “snapshot of what”? This claim element provides the answer: “of the virtual machine at rest.” In contrast, under an interpretation where “of the virtual machine at rest” does not modify both clauses, then “taking at least one snapshot” would not inform a person of ordinary skill in the art about the object of the snapshot. In other words, skilled artisans would immediately recognize that there must be a target object that is subject to the snapshot, and the language “of the virtual machine at rest” provides that target object for the first and second clauses of the limitations reciting “[perform/performing] at least one of: (i) taking at least one snapshot, and (ii) requesting taking at least one snapshot of the virtual machine at rest.” ’031 patent, cls. 1, 9, 16. For this reason, Dr. Stavrou’s interpretation would be understood by a person of ordinary skill in the art to be incorrect.

49. Second, this interpretation is consistent with the specification of the ’031 patent, which further informs the person of ordinary skill in the art’s understanding. The ’031 patent is titled “Techniques for Securing Virtual Cloud Assets *at Rest* Against Cyber Threats.” ’031 patent, Cover. The ’031 patent also

makes repeated, consistent statements supporting that snapshots are of VMs “at rest” whether taken or requested. Below I enumerate several such statements from the Abstract, the figures, the problem to be solved, and the disclosed embodiments:

- “A method and system for securing virtual cloud assets at rest against cyber threats.” *Id.* Abstract.
- “Similarly, in cloud computing, some machines (e.g., virtual machines) may also be at rest.... While at rest, the machines are powered off, and are not inspected for vulnerabilities, simply because scanners and/or installed monitoring agents cannot operate on a powered-off machine.... As such, when data or a machine at rest becomes active, undetected vulnerabilities can pose cyber threats. It would therefore be advantageous to provide a security solution that would overcome the deficiencies noted above.” *Id.* 2:20-38.
- “Certain embodiments disclosed herein include a method for securing virtual cloud assets at rest against cyber threats.” *Id.* 2:56-57; *see also id.* 2:58-3:14.
- “Various techniques disclosed herein include techniques for securing data at rest or machines at rest (collectively referred to as "machines at rest").... Machines at rest may include a virtual machine configured service(s) and/or application(s) that are inactive but can be accessed or executed as needed.... The disclosed techniques are utilized to scan for embedded vulnerabilities in machines at rest, when the machine is powered off.... According to the disclosed embodiments, the machine

is scanned for vulnerabilities when it is in its inactive step.” *Id.* 3:45-59.

- “Further, the view is parsed and analyzed by the security system 140 to detect vulnerabilities. This analysis of the views does not require any interaction and/or information from the VM 119-2. In fact, the VM 119-2 is in its inactive state (at rest) during the analysis.” *Id.* 6:23-27.
- “In an embodiment, a snapshot of the VM, at rest, is accessed.” *Id.* 8:42.
- “Receive a request to scan a VM at rest for vulnerabilities” *Id.* Fig. 2.

50. Third, this interpretation is also consistent with, and informed by, the applicant’s repeated and consistent remarks made in prosecution in relation to the claims, this limitation, and distinguishing the prior art. After receiving rejections based in part on a combination of prior art references including U.S. Patent 10,536,471 to Derbeko (*see* Ex.1004 at 195 (non-final rejection), 151 (final rejection)), Orca amended its claims to recite “perform at least one of taking a snapshot *of the virtual machine at rest*, or requesting the snapshot *of the virtual machine at rest*.” Ex.1004 at 123 (claim 20), 125 (claim 30), 127 (claim 39) (emphasis added). At the same time, Orca submitted arguments that the amended claims were patentable over the prior art rejections, and three times characterized this claim element as:

“perform[ing] at least one of: (i) taking at least one snapshot, and (ii) requesting taking at least one snapshot of the virtual machine at rest, wherein the at least one snapshot represents a copy of the virtual disks of the virtual machine at a point in time”

Id. at 130-132. Orca distinguished Derbeko, arguing that “[n]or does Derbeko describe **taking** and analyzing **a snapshot while the VM is at rest** elsewhere.” *Id.* at 131. Orca further distinguished Bansal, stating that this reference does not “solve or attempt to solve the problem addressed by the pending claims, e.g., to analyze a **snapshot taken while the VM is at rest** to detect vulnerabilities.”² *Id.* at 131.

51. Thus, the intrinsic record (i.e., the claims, the specification, and the prosecution history) informs a person of skill in the art that “of the virtual machine at rest” modifies both “(i) taking at least one snapshot” and “(ii) requesting taking at least one snapshot.”

2. “Analyzing the At Least One Snapshot”

52. Claims 1 and 16 recite “analyze the at least one snapshot to detect vulnerabilities.” Similarly, claim 9 recites “analyzing the at least one snapshot to

² This passage appears to contain a clerical error, as it states that Bansal’s technique “also **does** solve...” when it is clear from the context that Orca intended to argue that it does *not* solve the problem addressed by the claims.

detect vulnerabilities.” Dr. Stavrou labels these as elements 1.6, 9.7, and 16.7. Ex.1002, p. 131.

53. Dr. Stavrou states that he was “asked to assume that ‘[analyze/analyzing] the at least one snapshot’ also encompasses analysis of a VM instantiated from the snapshot.” Ex.1002 (Stavrou Decl.), ¶83; *see also* ¶84 (“Accordingly, my discussion of Ground 1 below applies an understanding of ‘analyze the at least one snapshot’ in claims 1 and 16, as well as ‘analyzing the at least one snapshot’ in claims 9, as encompassing both direct analysis of the snapshot data and analysis of a VM instantiated from the snapshot.”). Thus, Dr. Stavrou treats limitations in the challenged claims that recite “[analyze/analyzing] the at least one snapshot” as though they “encompass[] both [1] direct analysis of the snapshot data and [2] analysis of a VM instantiated from the snapshot.” Ex.1002 (Stavrou Decl.), ¶¶81-84.

54. I disagree that this claim element encompasses “[2] analysis of a VM instantiated from the snapshot” because that is not how a person of ordinary skill in the art would understand the claim element. Dr. Stavrou does not try to justify this claim element encompassing “analysis of a VM instantiated from a snapshot,” but simply states that he was “asked to assume” this claim construction. Ex.1002 (Stavrou Decl.), ¶83; *see* Ex.2077 (Stavrou Tr.) 265:17-266:4, 266:13-267:15 (“I didn’t ask why. I was asked to assume.”), 267:16-268:3, 268:4-269:20.

55. A person of ordinary skill in the art would not understand the “[analyze/analyzing] the at least one snapshot” limitations in the ’031 patent to encompass “analysis of a VM instantiated from a snapshot.” As an initial matter, there appears to be no disagreement that an instantiated VM is not a snapshot—otherwise Dr. Stavrou would not have enumerated it as separate from “direct analysis of snapshot data” in his assumed construction. Rather, an instantiated VM is a VM, and analysis of a VM instantiated from a snapshot is an analysis of a VM—not a snapshot. Once a VM is instantiated from a snapshot, e.g., when a duplicate VM is instantiated from a snapshot for subsequent monitoring, the instantiated VM is no longer a “snapshot,” and monitoring, assessing, or evaluating it is not “analyze/analyzing” a “snapshot” as claimed. This would have been readily understood by a person of ordinary skill in the art reading the ’031 patent based on the plain language of the ’031 patent’s claims and general knowledge—an instantiated VM is *not* a snapshot including because “the at least one snapshot represents a copy of the virtual disks of the virtual machine *at a point in time.*” ’031 patent, claims 1, 9, 16; *see also* ’031 patent, 5:35-36 (“A VM’s snapshot is a copy of the machine’s virtual disk (or disk file) *at a given point in time.*”). The specification also distinguishes analyzing a snapshot by a “security system 140 to detect vulnerabilities” (’031 patent, 6:17-18), as claimed, from “another embodiment” where a system “instantiate[s] a copy of the VM” and performs

“execution of the instance of the VM in an isolated sandbox” (*id.* 6:62-7:11, 8:49-59).

56. In fact, as I discuss further below, it appears that Dr. Stavrou “assume[d] that ‘[analyze/analyzing] the at least one snapshot’ also encompasses analysis of a VM instantiated from the snapshot,” as he was “asked” to (Ex.1002 (Stavrou Decl.), ¶ 83), simply to try to buttress opinions he offers that depend on disclosures from prior art references, such as Veselov, that discuss evaluating duplicate VMs, not snapshots. *See* Sections VIII.A.2.b (discussing limitations 1.5, 9.6, 16.6 and limitations 1.6, 9.7, 16.7).

VII. Overview of the Asserted References

A. Overview of Veselov

57. Veselov is titled “Security Assessment of Virtual Computing Environment Using Logical Volume Image” and discloses systems and techniques that perform a security assessment on “virtualized reproductions of computing resources” such as VMs or instances of VMs. Ex.1007 (Veselov), Title, Abstract, 3:20-23. Figure 1 illustrates an exemplary environment and components:

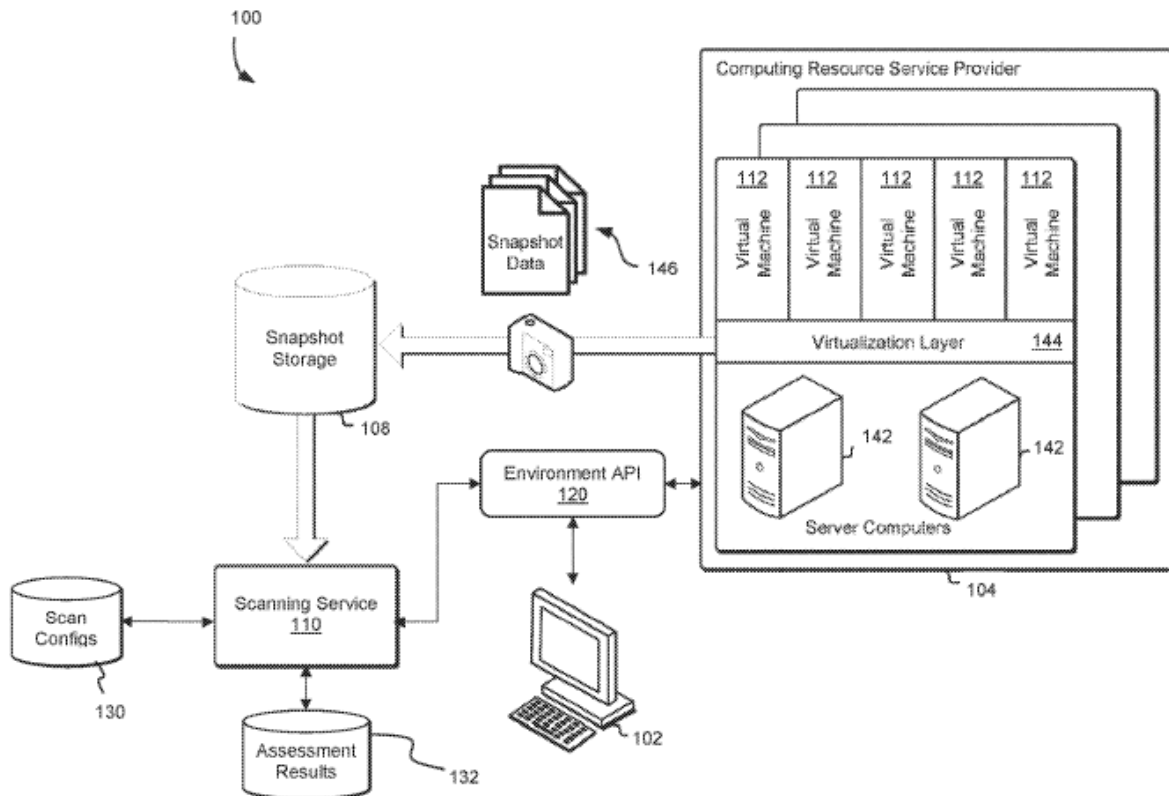


FIG. 1

58. As seen in Figure 1, Veselov discloses embodiments of a computing environment involving a computing resource service provider (shown as no. 104), in which user devices (shown as no. 102) request or manage resource allocation for server computers (shown as no. 142). Ex.1007 (Veselov), 4:19-29. These server computers are allocated as virtual computing resources. Ex.1007 (Veselov), 4:19-29. Figure 1 also depicts an API (shown as no. 120) that allows a user to connect, through, for example, a web-based application, to the computing resource service provider. Ex.1007 (Veselov), 4:29-41.

59. Veselov states that the scanning service of Figure 1 implements various disclosed steps for a security assessment. Ex.1007 (Veselov), 8:16-17. The scanning service obtains and processes snapshot data to generate assessment results. Ex.1007 (Veselov), 8:18-21. In implementing its scanning procedures, Veselov attempts to address drawbacks with resource usage during the security assessments. For example, Veselov highlights that there is a problem in the cybersecurity field where security assessments can “reduce the target computing device’s capabilities” such that the target machine “may not be capable of responding to a client as quickly as normal” or “may not be capable of providing content as quickly as normal.” Ex.1007 (Veselov), 2:56-67. Veselov also cautions that security assessments may impact machine capabilities because system components such as processors and memories are used to perform the assessment rather than to execute normal workloads. Ex.1007 (Veselov), 2:56-67. And, Veselov notes that security assessments using a target device’s computing resources “may be costly monetarily” and will require “an extended period of downtime.” Ex.1007 (Veselov), 3:8-11.

60. Veselov’s overarching goal is therefore to avoid downtime of VMs and reduce assessment complexity when performing security assessments. Throughout its disclosure, Veselov aims to preserve the activity of the original virtual machine instance. For example, Veselov provides systems that “include a

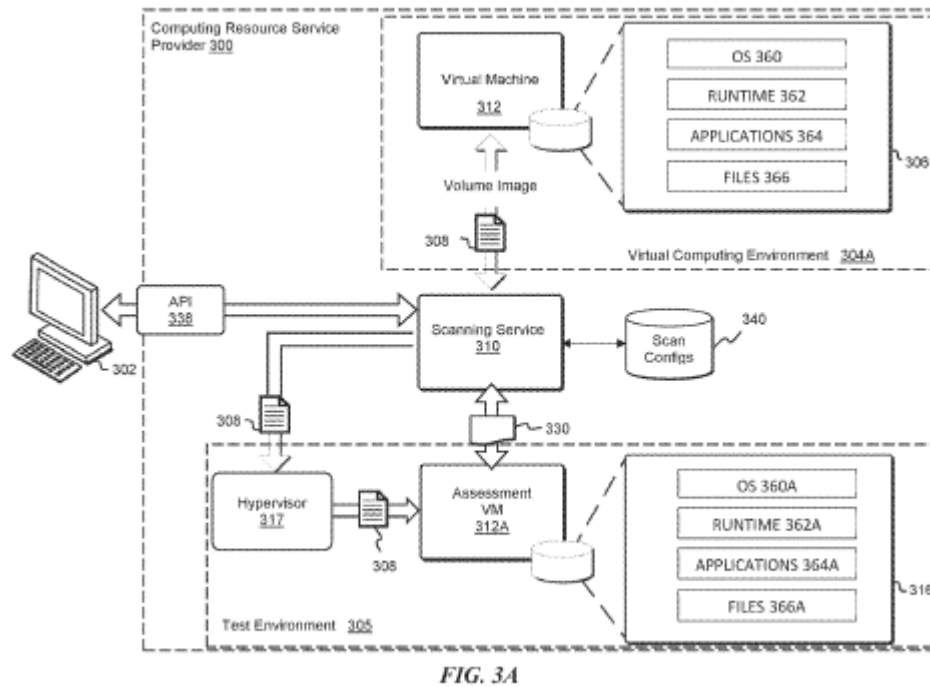
scanning service that facilitates duplication of all or a portion of the target computing resource, and then perform[] the security assessment on the duplicate computing resource”—“to avoid consuming processing time, processing power, and storage space of the target computing resource.” Ex.1007 (Veselov), Abstract; *see also* Ex.1007 (Veselov), 10:9-11 (generating a scannable volume from a snapshot includes “launching a duplicate virtual machine instance.”). And, Veselov provides that performing a robust security assessment may reduce the target computing device’s capabilities and may not be as responsive should an assessment be performed on the target machine. Ex.1007 (Veselov), 2:56-67. And Veselov cautions against “downtime.” Ex.1007 (Veselov), 3:8-11. Veselov explains that its solution allows the assessment to “be performed without impacting the customer’s virtual machine,” which can continue to “servic[e] requests from its customers or perform[] other useful work.” Ex.1007 (Veselov), 12:21-43. For the system to perform such “other useful work” Veselov must keep its VMs active during the capture of a snapshot and subsequent vulnerability analysis. Additionally, Veselov counsels solutions where the VM is configured to access one or more physical or virtual resources that are isolated in the event data dependencies need to be evaluated during an assessment. Ex.1007 (Veselov), 13:1-22, 13:33-36, 13:43-67. And the scanning service can cause “activity to occur

within the assessment virtual machine” to apply tasks in a real-time environment.

Ex.1007 (Veselov), 15:41-59, 20:57-21:13, 21:21-32.

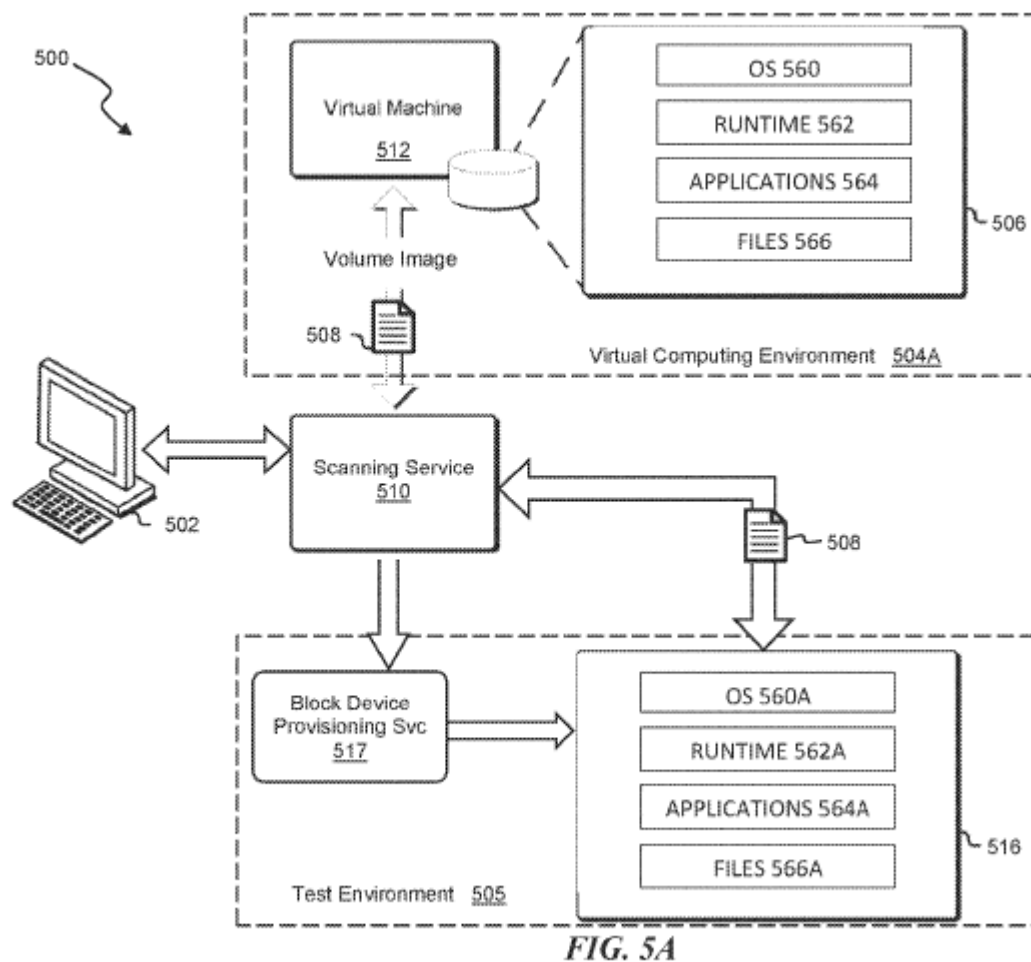
61. Veselov proposes two scanning solutions to maintain the active state of the target virtual machine. In both of Veselov’s solutions, Veselov implements snapshot technology to capture the state of the target virtual machine. *See, e.g.*, Ex.1007 (Veselov), Fig. 1, no. 146 (snapshot data). Veselov implements a virtual layer to generate or obtain access to a snapshot or snapshot data to initiate a security assessment. Ex.1007 (Veselov), 3:23-32, 6:25-49. Veselov’s snapshot includes all data necessary to “recreate the state of the computing resource” within a duplicate resource so as to preserve the active state of the machine. Ex.1007 (Veselov), 3:27-29, 3:41-34, 12:21-27.

62. In Veselov’s first solution, Veselov’s scanning service sets up and launches an active virtual machine instance (a duplicate machine for assessments). Ex.1007 (Veselov), 3:54-59. The duplicate machine has the same configuration as the original machine instance at the time of the snapshot capture. Ex.1007 (Veselov), 3:54-59. Figures 3A and 3B are informative.



63. In Figure 3A, the scanning service performs the security assessment on the assessment machine via a security agent program. Ex.1007 (Veselov), 14:13-27. The scanning service receives assessment results via the agents on the assessment machine. Ex.1007 (Veselov), 14:28-34. The assessment machine can implement processing activity in order to demonstrate real-time dynamic or static data, network connections, and communications to other computing resources. Ex.1007 (Veselov), 15:41-59. Additionally, a user or administrator may communicate instructions to the scanning service (shown inside the computing resource service provider's environment as no. 110 in Figures 3A and 3B) via the API (shown as no. 338) to configure and accomplish the security assessment. Ex.1007 (Veselov), 14:36-15:40.

64. In Veselov's second solution, Veselov provides that the scanning system may "mount" the logical volume as a hard drive with data stored thereon or may transform the snapshot into files to be read. Ex.1007 (Veselov), 3:61-66. This technique permits a security assessment on the data at the file system level. Ex.1007 (Veselov), 3:66-67. Figures 5A and 5B are informative.



65. In exemplary Figure 5A, the scanning service analyzes snapshot data in a "test environment" outside of the virtual computing environment. Ex.1007 (Veselov), 17:9-12. In both Figures 5A and 5B, the user communicates directly

with the scanning service outside the target resource computing environment. Ex.1007 (Veselov), Figs. 5A-B. 17:9-67. An API is not depicted in Figure 5A or 5B, and is not used for communication between the user and scanning service. The scanning service interfaces with VM resources through provisioning services such as a hypervisor. Ex.1007 (Veselov), Figs. 5A-B. 17:9-67.

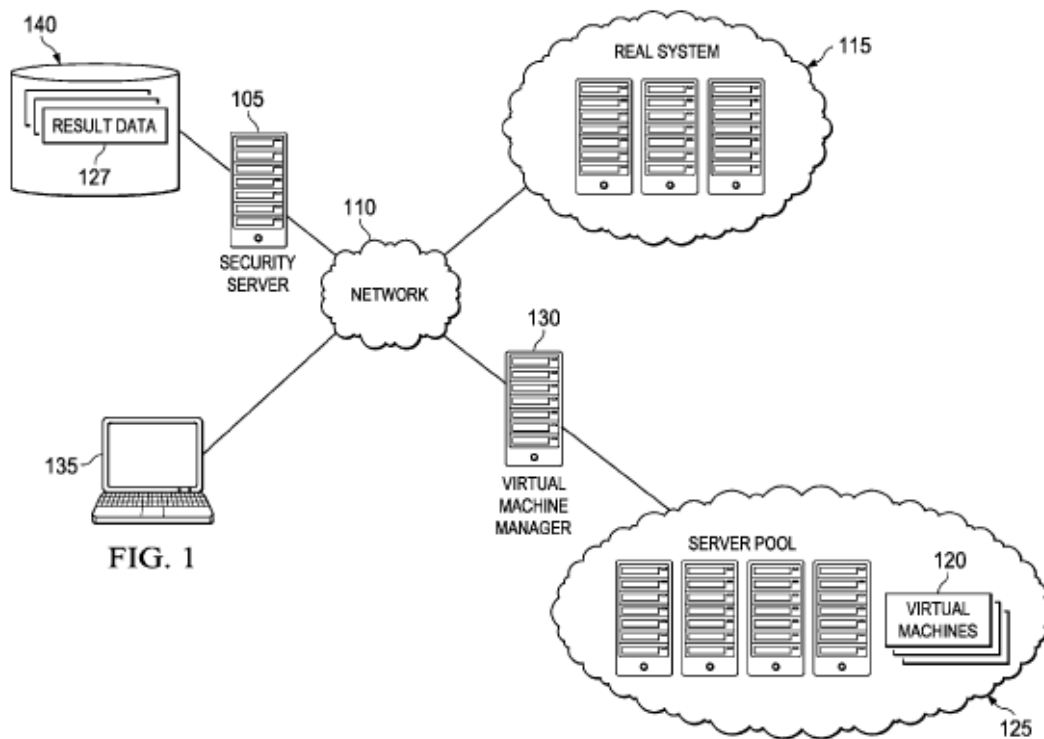
66. In both solutions, Veselov's scanning system stores the assessment results or provides them to "an owner or administrator of the virtual machine in order to evaluate the vulnerabilities of the target resources." Ex.1007 (Veselov), 4:5-8. After storing or sharing the results, the scanning system may delete the snapshot or store it for additional analysis. Ex.1007 (Veselov), 4:8-10.

67. In both solutions, Veselov's snapshots are "prepared so that the security assessment produces the results that the scanning system otherwise would have obtained if the security assessment were performed on the original virtual machine instance." Ex.1007 (Veselov), 4:1-4. Accordingly, as detailed in this section, Veselov's techniques purport to preserve the activity of the original virtual machine instance.

B. Overview of Price

68. Price is titled "Security Assessment of Virtual Machine Environments" and is generally directed to performing security assessments for offline and online VMs. Ex.1048 (Price), Cover, Abstract. More specifically,

Price involves a set of virtual machines managed by a virtual machine manager, and for each virtual machine, Price determines whether the machine is offline. Ex.1048 (Price), Abstract. For the machines that are determined to be offline, Price's system implements techniques to collect an image from the offline machine. Ex.1048 (Price), Abstract. In contrast, for the machines that are determined to be online, Price implements agents that are loaded onto each machine and can assess security of the online machines. Ex.1048 (Price), Abstract.



69. Figure 1 depicts the environment in which Price's techniques are implemented. Ex.1048 (Price), Fig. 1. Price provides a computing system (not shown, no. 100 in Fig. 1) with security tools (shown as no. 105) that perform scans

on VMs (shown as no. 120) in connection with policy compliance, vulnerability assessment, malware protection, among other security services. Ex.1048 (Price), ¶16. Price also discloses one or more client devices (shown as no. 135) for communicating over a network (shown as no. 120) with other system devices. Ex.1048 (Price), ¶16. The VM manager (shown as no. 130) manages a set of VMs and provides administrators and customers with interfaces for deploying and maintaining VMs hosted within the server pool (shown as no. 125) in the computing system. Ex.1048 (Price), ¶16.

70. Price provides a series of steps for implementing its scanning techniques. Ex.1048 (Price), Fig. 6. First, the system identifies a VM manager that manages a set of VMs (or VMs outside of the set). Ex.1048 (Price), ¶44. The system may also identify each VM within the set with aspects and conditions of each respective VM. Ex.1048 (Price), ¶44. With this information, the system can determine whether a given VM is online or offline. Ex.1048 (Price), ¶44.

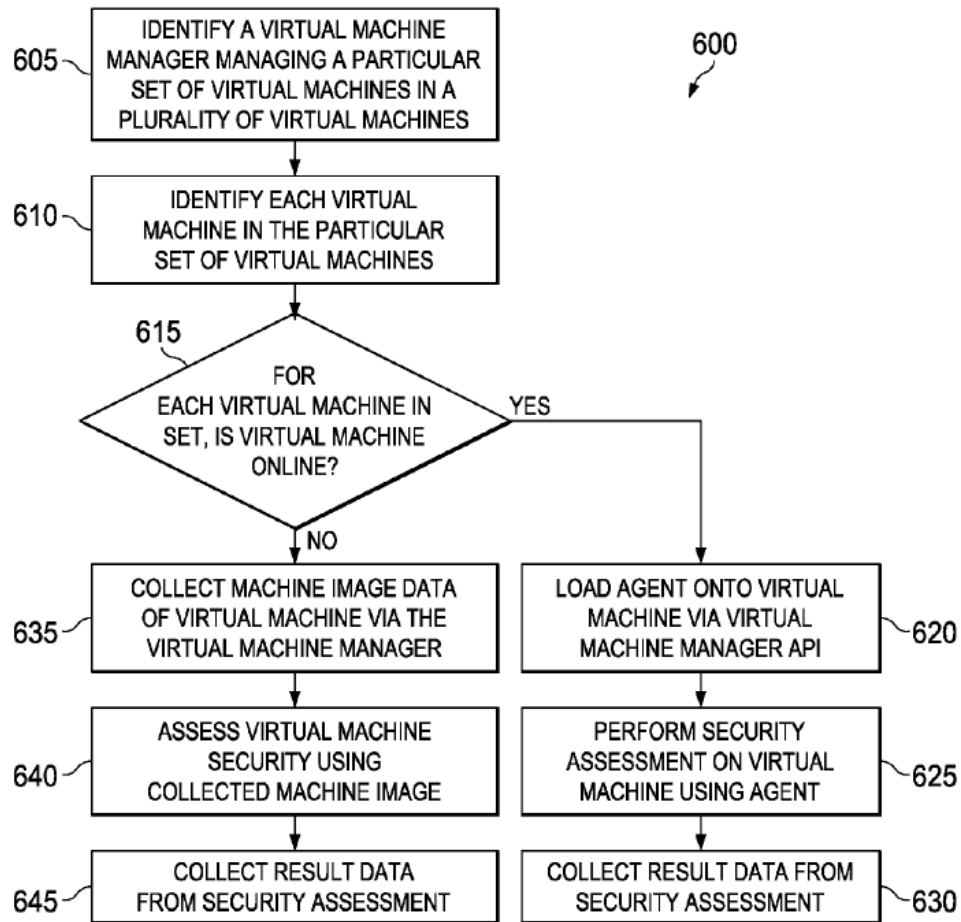
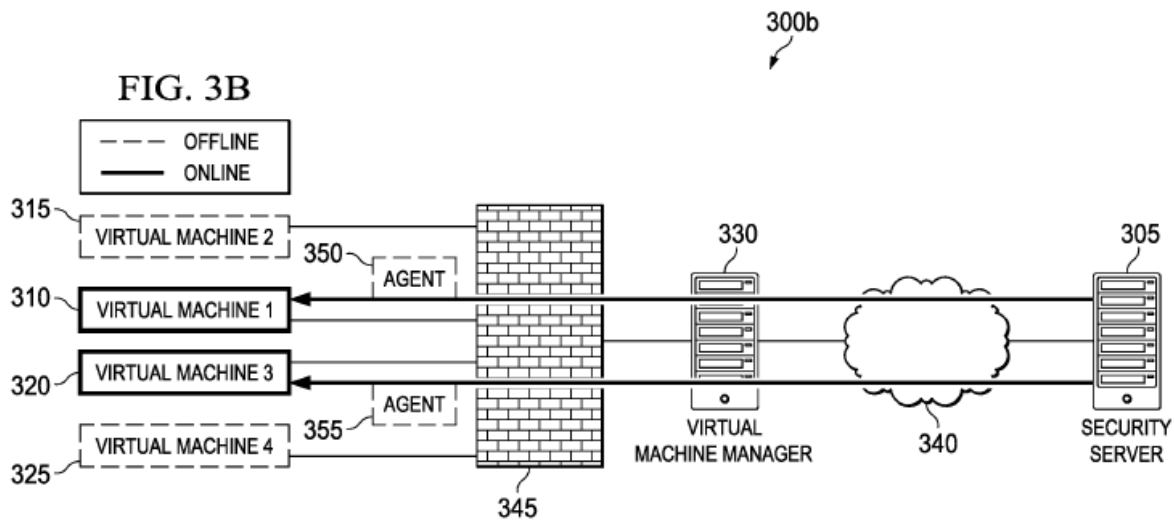
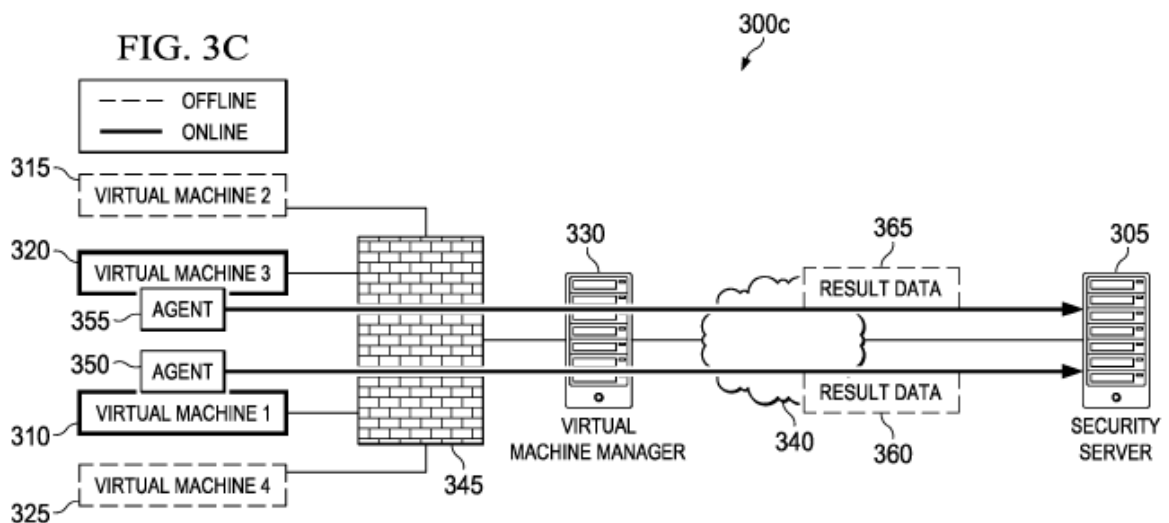


FIG. 6

71. For online machines, the system loads agents onto the target VM via an interface of the VM manager. Figures 3B and 3C are informative:



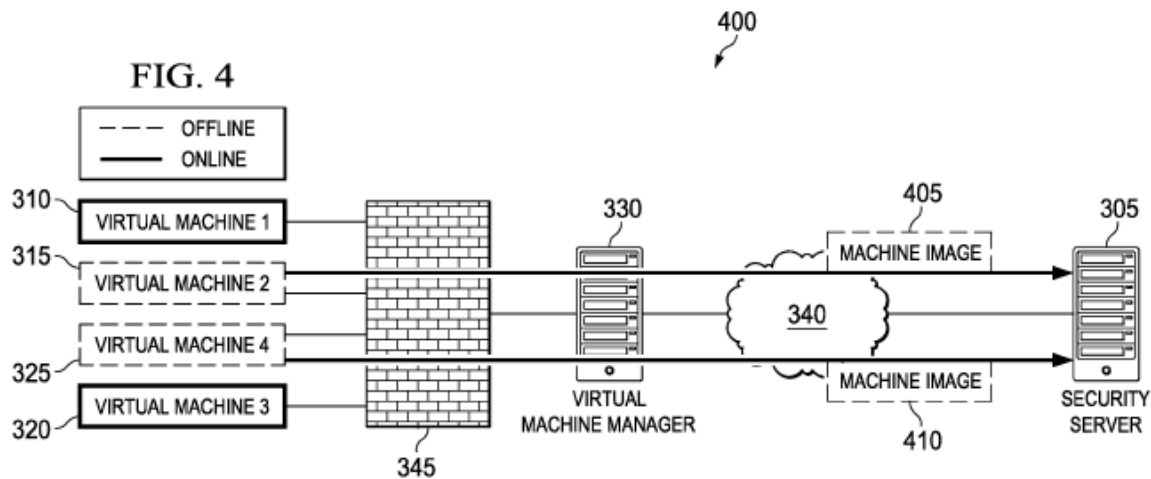
72. Figure 3B demonstrates a scenario where the security server obtains the identity and locations of VMs and loads agents onto the VMs (or loads agents based on the guest operating systems of the VMs). Ex.1048 (Price), ¶37.



73. Figure 3C of Price demonstrates a scenario where agents are scanning resources of online VMs. Ex.1048 (Price), ¶38. Through the scanning procedures,

the agents can generate results data to return to the security server. Ex.1048 (Price), ¶38.

74. For offline machines, the system collects images through the VM manager interfaces or APIs. Ex.1048 (Price), ¶41. Figure 4 is informative:



75. Figure 4 demonstrates a scenario where the system determines that given VMs are offline and collects VM images. Ex.1048 (Price), ¶41. The system can then use the image data to complete security assessments. Ex.1048 (Price), ¶41.

76. In both offline and online scenarios, the methods conclude with a determination of results from the security assessments. Ex.1048 (Price), Fig. 6, ¶¶41, 43, 45. Neither scenario utilizes nor analyzes a snapshot of a VM.

C. Overview of Hufsmith

77. Hufsmith is titled “Combined Threat Score for Container Images.” Ex.1078 (Hufsmith), Cover. Hufsmith discloses “a process for determining threat scores for container images or distributed applications that consider the results of a multitude of different scanners and other factors.” Hufsmith, Abstract. Hufsmith’s “Field” is disclosed as “techniques that combine metrics of heterogenous vulnerability scans of container images.” Hufsmith ¶2. Hufsmith is therefore directed to methods for “determining combined threat score[s] for container images,” not snapshot-based assessments of VMs. *See* Hufsmith, Title, ¶2, ¶¶4-6.

78. Hufsmith discusses problems in the field of scanning container images and teaches solutions involving its techniques for analyzing scores for container images to determine a combined threat score for the images. For example, Hufsmith discloses that components of distributed applications, called “services,” may be composed of “one or more executing bodies of code.” Ex.1078 (Hufsmith), ¶5. And, Hufsmith explains that such “bodies of code and other resources” can be difficult to secure because, for example, services arise from “diverse sets of bodies of code,” “can undergo relatively frequent version changes,” and can be “downloaded from third parties ... that may be un-trusted or accorded less trust than code built in-house.” Ex.1078 (Hufsmith), ¶5.

Additionally, Hufsmith provides that in such “potentially highly complex” environments, even where potential vulnerabilities are detected, “a degree to which they pose a threat” in an execution environment with distributed code and other resources can be “difficult to quantify.” Ex.1078 (Hufsmith), ¶6. And, Hufsmith explains that such difficulties are heightened for applications built with containers such as Docker™ containers, Open Container Initiative (OCI) containers, or with Kubernetes™ because these containers “generally virtualize at the operating system level, in contrast to virtual machines” which also emulate underlying hardware. Ex.1078 (Hufsmith), ¶26. For example, containers are generally formed from multiple layers and offer OS-level virtualization with various benefits not seen in virtual machines, “including lower computational load, faster spin up, and sharing of resources,” as well as the ability to reuse “intermediate images of read-only bodies of code and other resources” “across multiple container images.” Ex.1078 (Hufsmith), ¶¶26-27. However, such features can complicate security of the distributed application due to “false positives” in the container context, along with an inability to obtain holistic views into container exposure (i.e., because of the container’s layered nature). Ex.1078 (Hufsmith), ¶¶28-29. Also, Hufsmith explains the prior art lacks “multi-sourced vulnerability assessments” and “combined threat scores for container images.” Ex.1078 (Hufsmith), ¶¶28-29.

79. Accordingly, Hufsmith teaches techniques to combine metrics of heterogenous vulnerability scans of container images. Ex.1078 (Hufsmith), ¶2. Hufsmith teaches that its techniques are useful in the context of “software development related to distributed applications” (Ex.1078 (Hufsmith), ¶2) and its techniques are meant to assist software developers during development of container images, i.e., before containers are instantiated from a container image. Ex.1078 (Hufsmith), ¶153.

80. Hufsmith provides an architecture to implement a scanning engine to facilitate the vulnerability scans. Ex.1078 (Hufsmith), Fig. 1A.

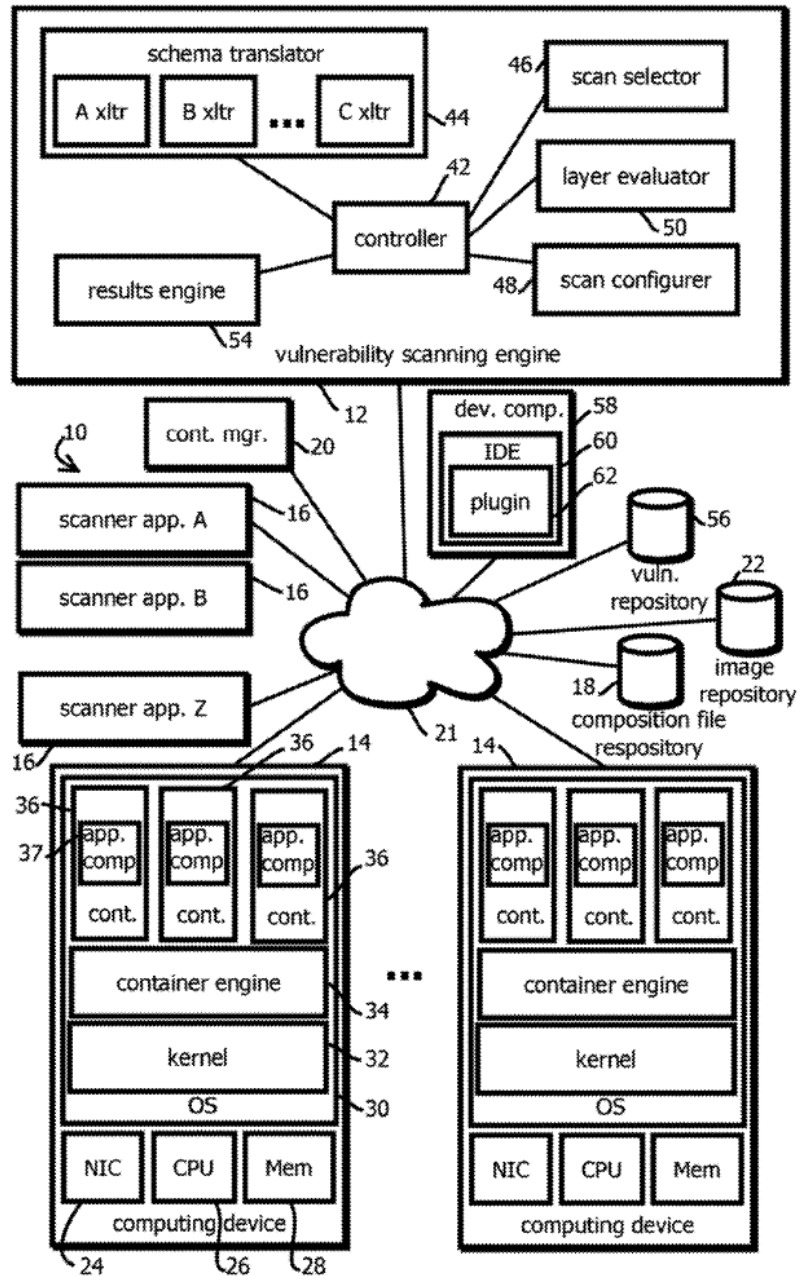


FIG. 1A

81. Figure 1A depicts a vulnerability scanning engine (shown as no. 12) that implements a controller (shown as no. 42), a schema translator (shown as no. 44), a scan selector (shown as no. 46), a layer evaluator (shown as no. 50), a scan configurer (shown as no. 48), and a results engine (shown as no. 54) that coordinate

to arbitrate which layers and which portions of layers will be scanned by one of various scanning applications (shown as no. 16). Ex.1078 (Hufsmith), ¶68. The vulnerability scanning engine decides which scanner application will perform the scan among a set of heterogeneous scanner applications that are each configured to apply a different type of scan to a different type of body of code and other resources such as images and files. Ex.1078 (Hufsmith), ¶68.

82. Figure 1B demonstrates the results engine (shown as no. 54) communicatively coupled (via a network shown as no. 21) with the container manager (shown as no. 20) and the container score database (shown as no. 102).

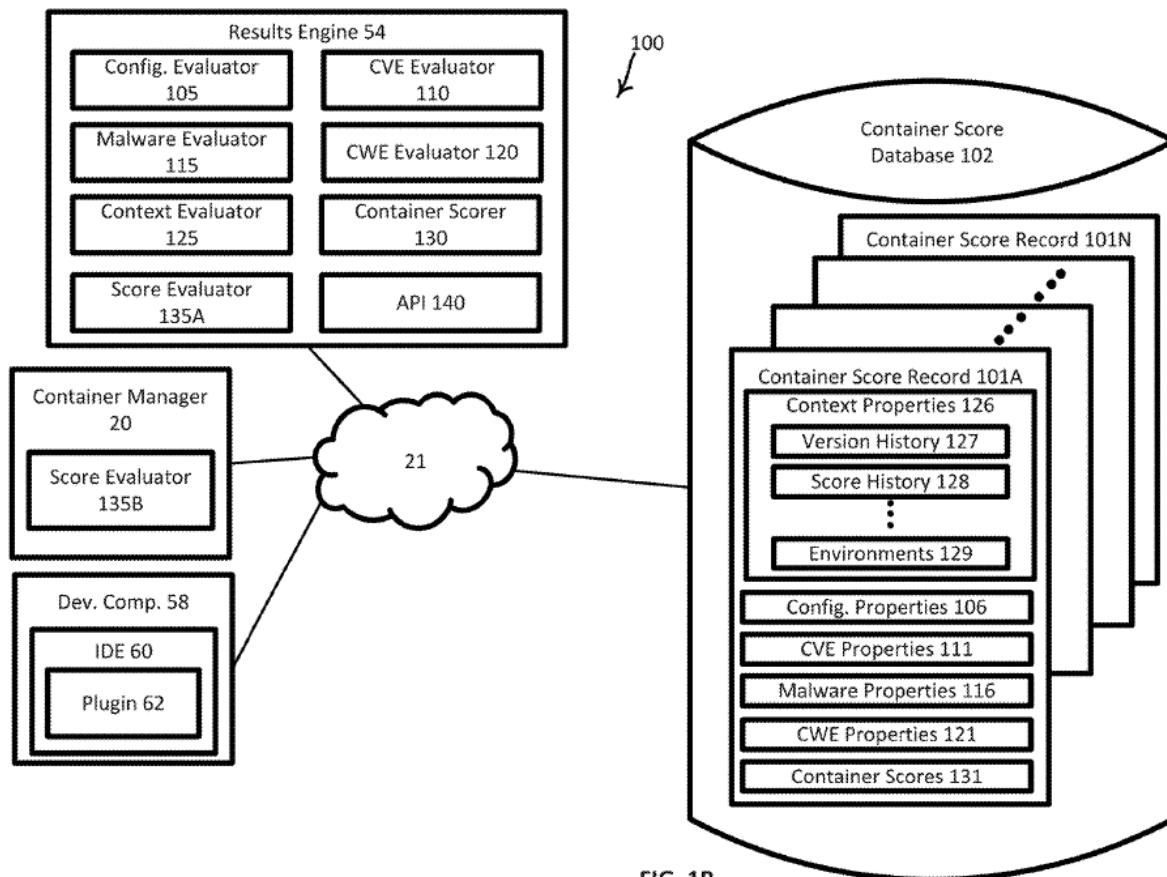


FIG. 1B

83. Hufsmith provides that the results of the vulnerability scanning engine will be translated into a unified schema of scanner results or properties, which may be stored in a database “in association with an identifier of the container image.” Ex.1078 (Hufsmith), ¶101. The database may be a container score database which comprises a number of container score records (shown as nos. 101A to 101N) that are referenced by, for example, a cryptographic hash value of the container image. Ex.1078 (Hufsmith), ¶101. In these scenarios, where the container score records correspond to container images that constitute a distributed application, a given container score record may reference one or more other container score records. Ex.1078 (Hufsmith), ¶104.

84. Figure 4, below, illustrates a process for “determin[ing] one or more score records for a distributed application having one or more container images. Ex.1078 (Hufsmith), Fig. 4, ¶¶18, 199.

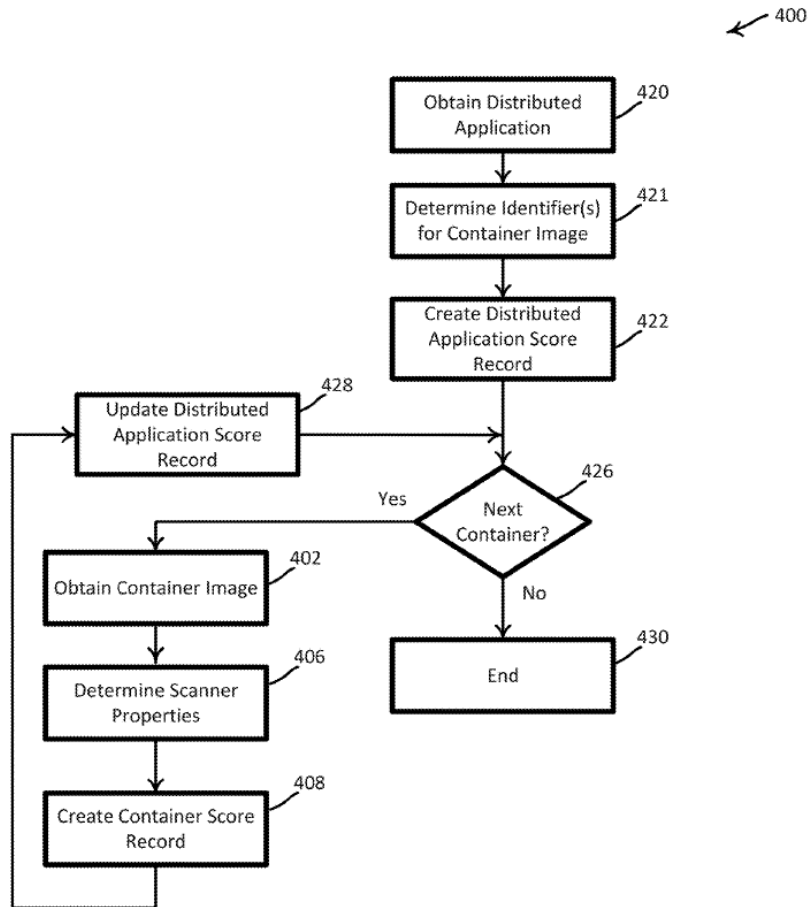


FIG. 4

85. First, Hufsmith provides that the system obtains a “distributed application” (depicted by block 420). Ex.1078 (Hufsmith), ¶201. The second step (depicted by block 421) provides that the system will identify container images associated with the distributed application “(which may be a multi-container distributed application).” Ex.1078 (Hufsmith), ¶202.

86. Then, at the third step (depicted by block 422), the system creates a distributed application score record for the distributed application. Ex.1078 (Hufsmith), ¶201-202. And, when the system determines that a container image to process exists (depicted by block 422), the system will execute steps 402-408. Ex.1078 (Hufsmith), ¶201-202. The system therefore obtains a container image (depicted by block 402). Ex.1078 (Hufsmith), ¶200. Then, scanner properties are determined for the container image (depicted by block 406). Ex.1078 (Hufsmith), ¶200. The system then proceeds to Figure 2 (below). Ex.1078 (Hufsmith), ¶200.

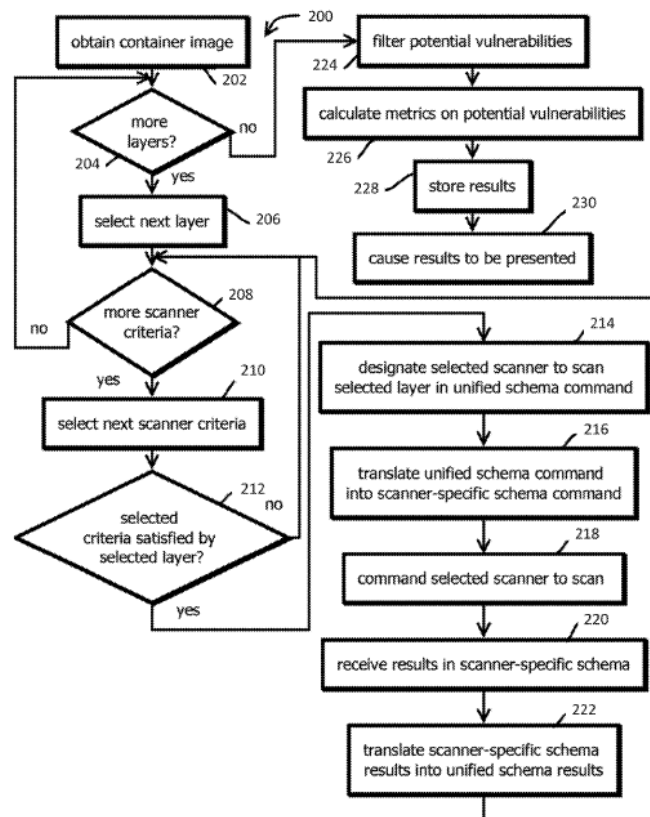


FIG. 2

87. At Figure 2, the system first obtains a container image (depicted by block 202). Ex.1078 (Hufsmith), ¶152. Then, Hufsmith provides that the system determines whether there are additional layers in the container image to be processed (depicted by block 204). Ex.1078 (Hufsmith), ¶152. For example, the system will start with the base or top layers of a multi-layer container, and, if there are additional layers to process, then the system will continue to select a layer (depicted by block 206). Ex.1078 (Hufsmith), ¶152. Then, the system determines whether there is additional scanner criteria that should be applied to a given layer (depicted by block 208). Ex.1078 (Hufsmith), ¶152. If the system determines that there are additional criteria to apply, then an option for “next scanner criteria” may be selected (depicted by block 210). Ex.1078 (Hufsmith), ¶152. The system determines whether a given layer is satisfied by selected criteria (depicted by block 212), and if so, the scanner that corresponds to the chosen criteria will be selected to scan the given layer (depicted by block 214). Ex.1078 (Hufsmith), ¶152. Next, the system executes a command telling the scanner to scan, and the results of the scanning are received and translated (depicted by blocks 216-222). Ex.1078 (Hufsmith), ¶152. Then, once there are no further layers to scan, the system will filter potential vulnerabilities and calculate metrics (depicted by blocks 224, 226). Ex.1078 (Hufsmith), ¶152. Subsequently, the system may store results and present results (depicted by blocks 228, 230). Ex.1078 (Hufsmith), ¶152. Then, the system

resumes at Figure 4, where a container score record for the container image may be created and a distributed application score may be updated. Ex.1078 (Hufsmith), ¶200, 202. The system will continue to process container images iteratively until it finds no additional container images to process. Ex.1078 (Hufsmith), ¶202 (describing blocks 426, 430 of Figure 4).

88. Figure 5 illustrates a process for “determin[ing] a combined threat score for a distributed application having one or more container images or a container image.” Ex.1078 (Hufsmith), ¶19, 204, Fig. 5.

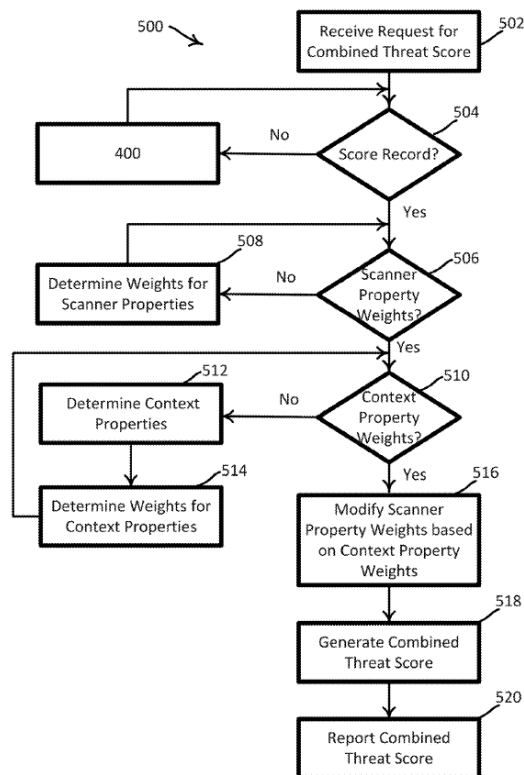


FIG. 5

89. Hufsmith's system of Figure 5 begins by receiving a request for a combined threat score (depicted by block 502). Ex.1078 (Hufsmith), ¶205. Then, the system determines whether a "score record for a distributed application or container image" exists, "such as a score record created by [the] process of FIG. 4." Ex.1078 (Hufsmith), ¶205. If no score record exists, Hufsmith provides that the "process 400 of FIG. 4 may be called to create" one. Ex.1078 (Hufsmith), ¶206. Then, after the system has a score record, the record "may be examined for scanner property weights" (depicted by block 506), which may be determined and stored to the score record if none already exist (depicted by block 508). Ex.1078 (Hufsmith), ¶206. Also, a score record "may be examined for context property weights" (depicted by block 510). Ex.1078 (Hufsmith), ¶207. And, if no context property weights exist, then the system includes an additional step of "requesting context property information, such as environment information," from which "weights are determined for the context properties at block 514." Ex.1078 (Hufsmith), ¶207. Then, weights associated with different scanner properties are modified based on weights of the corresponding context properties (depicted by block 516). Ex.1078 (Hufsmith), ¶208. And, after "each of the context property weights are applied, a combined threat score may be generated based on the modified scanner property weights" (depicted by block 518) and reported (depicted by block 520). Ex.1078 (Hufsmith), ¶¶208-209.

90. Accordingly, Hufsmith “determin[es] threat scores for container images or distributed applications that consider the results of a multitude of different scanners and other factors such as context information” for “container images.” Ex.1078 (Hufsmith), Abstract.

VIII. The Claims Of The '031 Patent Would Not Have Been Obvious

91. Petitioner and Dr. Stavrou challenge claims 1-16 of the '031 patent based on three grounds of purported obviousness, each of which depends on the combination of U.S. Pat. No. 11,216,563 (“Veselov,” Ex.1007) and U.S. Patent App. Pub. No. 2013/0247133 (“Price,” Ex.1048).

92. In my opinion, none of the claims of the '031 patent would have been obvious over Petitioner's and Dr. Stavrou's prior art, for reasons I explain below.

A. Ground 1: Claims 1, 3-9, and 11-16 Would Not Have Been Obvious Over Veselov and Price

93. In Ground 1 of the Petition, Petitioner and Dr. Stavrou challenge claims 1, 3-9, and 11-16 of the '031 patent based on the combination of Veselov and Price. *See* Ex.1002, Stavrou Decl. §XII.

94. In my opinion, Petitioner and Dr. Stavrou fail to show that any of claims 1, 3-9, and 11-16 of the '031 patent would have been obvious over the combination of Veselov and Price.

1. A Person of Ordinary Skill in the Art Would Not Have Been Motivated to Combine Veselov and Price and Would Not Have Had a Reasonable Expectation of Success in Doing So

a. Dr. Stavrou's analysis mixes different disclosures from Veselov without motivation to combine those disclosures.

95. Dr. Stavrou's analysis mixes disclosures from Figures 1, 3A, 3B, 5A, and 5B in Veselov without explaining why a person of ordinary skill in the art would have wanted to combine or modify them. *See* Ex.1002 (Stavrou Decl.), ¶122 (opining about a "communication interface"). Veselov discusses APIs only in relation to the examples in Figures 1 and 3A-3B, where computing environments are implemented by a computing resource service provider.³ *See, e.g.,* Ex.1007 (Veselov) 5:1-8 (In relation to Figure 1, "[t]he scanning service 110 may access the computing resource service provider 104, or any associated virtual computing environments and/or computing resources, via the API 120 or another API. In some embodiments, the scanning service 110 may use the API 120 to provide user interfaces to the user device 102, enabling the user to configure settings of the

³ I note that "computing resource service provider" is not typically used in the art to refer to a cloud computing environment. Additionally, Veselov does not expressly describe the APIs as cloud computing platform APIs or the computing resource service provider as a cloud computing platform.

scanning service 110 as described further below.”); *see also* 10:37-16:14 (discussing Figures 3A-3B).

96. The Figure 1 example, reproduced below, includes snapshot storage 108, also described as “snapshot storage service 108” that “may be a group of computer systems configured to store snapshot data 146 and make the snapshot data 146 accessible to one or more other computer systems, such as the scanning service 110.” Ex.1007 (Veselov) 7:34-37; *see also* 7:37-8:15 (describing snapshot storage service 108).

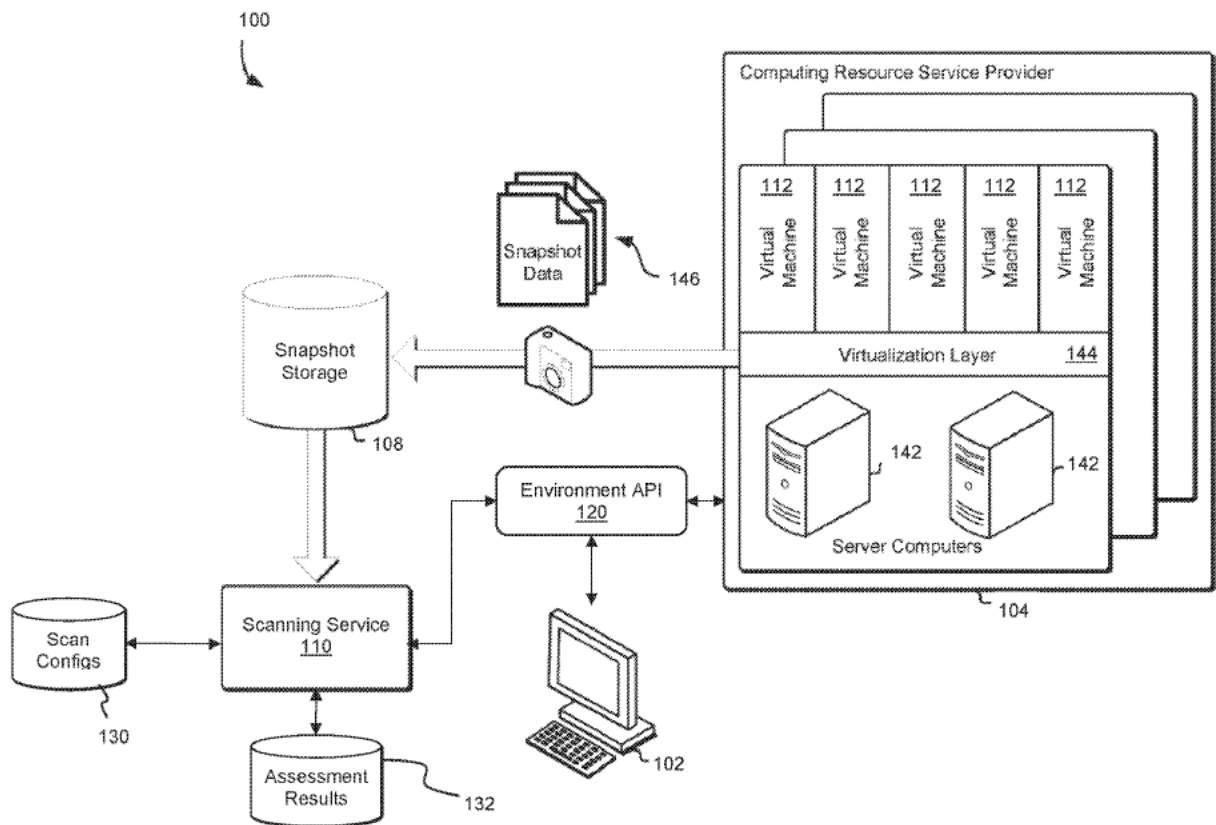


FIG. 1

Ex.1007 (Veselov) Fig. 1

97. In the context of systems that use snapshot storage, Veselov only discusses APIs in relation to a user device transmitting a request to the scanning service for a security assessment, where the “request may be an API call including information corresponding to the requestor, the user associated with the virtual machine 112, the on-demand storage service 108, and/or the particular virtual machine 112.” Ex.1007 (Veselov) 6:32-38. Veselov discloses that the scanning service “may determine the corresponding virtualization layer 144 for the virtual machine 112 identified by the request and transmit a comment to the virtualization layer 144 to provide snapshot data 146 (or to provide access to snapshot data 146) corresponding to the virtual machine 112.” Ex.1007 (Veselov) 6:38-43. The virtualization layer itself “may be configured to obtain the snapshot data 146 and send it to the scanning service 110” or store it in the storage service 108. Ex.1007 (Veselov) 6:43-46. But these disclosures do not teach or suggest identifying, querying, or returning the location of a virtual disk, let alone using APIs to do so. To the contrary, the scanning service is interacting with the virtual layer, which provides snapshot data without the scanning service needing to identify a virtual disk, let alone locate it. For further example, when snapshots are stored in the storage service, e.g., in a queryable database, Veselov discloses only that it is queryable based on the snapshot’s own identifier, or by “the identifier of the virtual machine instance from which the snapshot data 146 was derived.” Ex.1007

(Veselov) 7:50-53. Veselov's scanning service does not have any need or benefit, absent hindsight, to identify or locate a virtual disk. In summary, the Figure 1 embodiment of Veselov, on its own, cannot meet, e.g., the virtual disk limitations of claim 1, and it is thus unsurprising that Dr. Stavrou mixes disclosure of different examples to try to meet those limitations.

98. The Figure 3A/3B examples are different from Figure 1, including in that they do not include a snapshot storage service. Figure 3A is reproduced below.

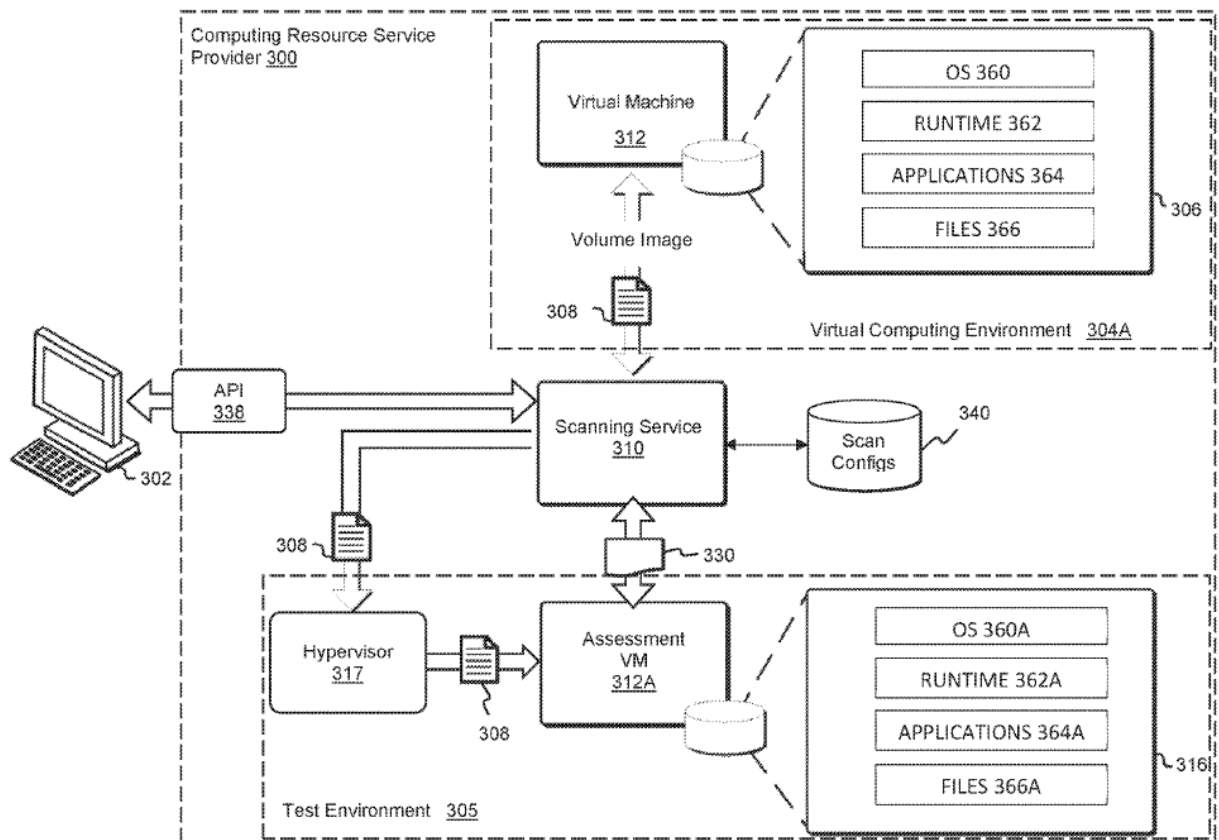


FIG. 3A

Ex.1007 (Veselov) Fig. 3A

As shown above, in Figure 3A the scanning service is part of the computing resource service provider. In this embodiment, the virtual machine may “use a logical volume 306, which is composed of a fixed or variable allocation of physical data storage resources, such as data storage blocks selected from one or more hard disk drives of one or more server computers.” Ex.1007 (Veselov)

11:5-9. The API 338 is used by a user device 302 to “access the scanning service 310.” Ex.1007 (Veselov) 14:37-41. The scanning service 310 “may obtain the volume image 308 of the virtual machine 312, use the volume image 308 to launch the assessment virtual machine 312A, B, and install the security agent program 330 in the assessment virtual machine’s 312A, B second logical volume 316 as described above and in accordance with the user input.” Ex.1007 (Veselov) 15:29-34. There is no disclosure of using—and no suggestion or need to use—the API to identify a virtual disk or query a location of a virtual disk in this Figure 3A/3B example. Further, the use of an assessment virtual machine does not teach analyzing the snapshot as required by the claims. *See* Section VI.B.2. Thus, the Figure 3A/3B embodiment on its own also cannot meet the claims.

99. In contrast, in the Figs. 5A-5B example, Veselov discusses assessing snapshots, but in the context of a different system than the system presented in Figure 3A and 3B. In the Figs. 5A-5B example, the resources are not implemented

within a Computing Resource Service Provider (i.e., what Dr. Stavrou appears to identify as a cloud computing platform), which is the entity that provides the API in Figs. 3A-3B. *See* Ex.1002 (Stavrou Decl.), ¶131 (referring to the user utilizing “the computing resource service provider’s API”). As such, it is not surprising that no API is illustrated in Figs. 5A-5B, let alone cloud computing platform APIs, or the use of such APIs as claimed in the ’031 patent. *See, e.g.*, Ex.1007 (Veselov), Figs. 5A-5B, 17:9-66.

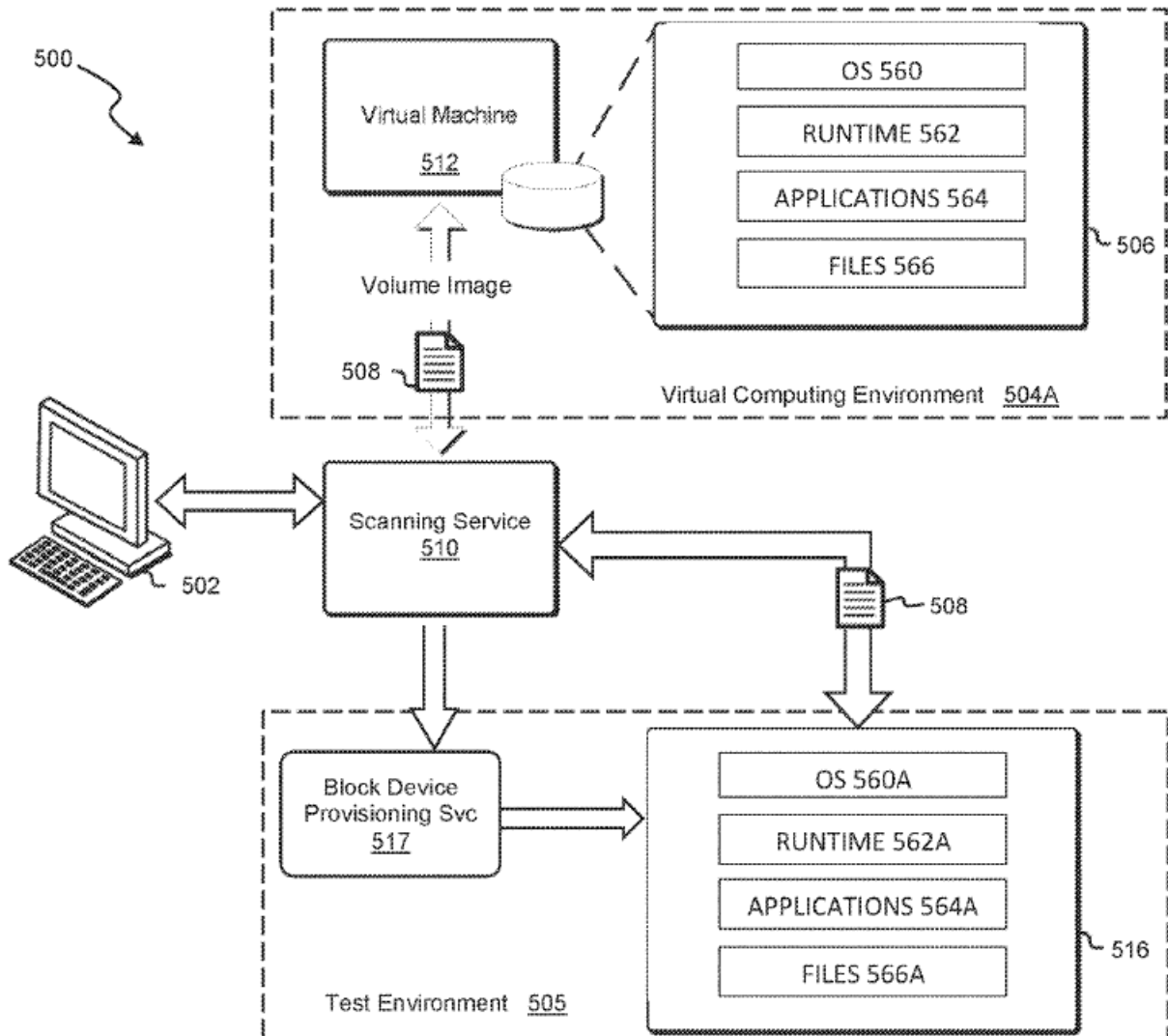


FIG. 5A

100. As shown in Fig. 5B, below, the scanning service 510 communicates with the virtual computing environment through “a provisioning service” such as “a hypervisor 518.” Ex.1007 (Veselov) 17:30:35. Hypervisors and APIs are not the same, and a person of ordinary skill in the art would have understood that they serve different purposes. *See, e.g.,* Ex.2021 [Amazon Web Service (AWS), *What is a Hypervisor?*] (“The hypervisor relays requests for processing power, memory,

storage, and other resources to the host machine in several ways, including API calls. An API is a software communication method that allows different applications to exchange data.”). Ex.2022 [Perez-Botero, *Characterizing Hypervisor Vulnerabilities*], §1 (“Virtualized environments are usually implemented with the use of a Hypervisor, which is a software layer that lies between the Virtual Machines (VMs) and the physical hardware. The Hypervisor allocates resources to the VMs, such as main memory and peripherals.”); Ex.2018 [Abraham Silberschatz et al., *Operating System Concepts* 716 (9th ed. 2013)], 63 (“Typically, application developers design programs according to an application programming interface (API). The API specifies a set of functions that are available to an application programmer, including the parameters that are passed to each function and the return values the programmer can expect”).

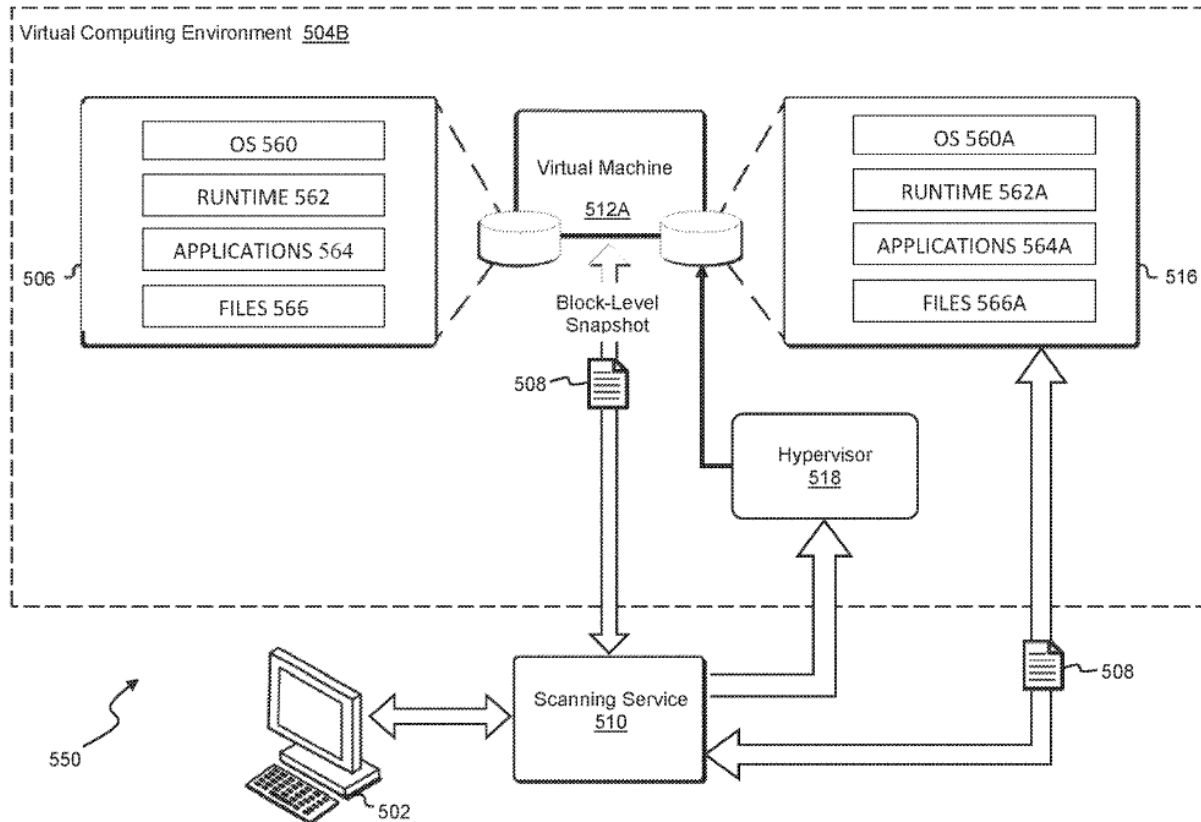


FIG. 5B

Ex.1007 (Veselov) Fig. 5B

101. Dr. Stavrou fails to explain why a person of ordinary skill in the art would have drawn from separate disclosures of these different examples, including why an API would be used in Veselov's Figure 5A-5B example—i.e., the only example Veselov discloses that is not directed to assessments of duplicate VMs. Nor does he explain why a person of ordinary skill in the art would have picked that one feature from the Figs. 3A-3B examples and not other features (e.g., assessing duplicate VMs) that do not meet the claims. The absence of any API in the Figures 5A-5B example seems to be what motivated Dr. Stavrou to just

“assume,” based on the direction of Petitioner’s counsel and without any explanation or support, that “analyzing [a] snapshot” encompasses analyzing duplicate VMs. *See* Section VI.B.2 (explaining that assessments of duplicate VMs are not analyses of snapshots and that Dr. Stavrou’s contrary assumption is incorrect). Using his improper assumption, the line between the Figures 5A-5B example, which does not have an API of any kind, and the Figures 3A-3B example, which shows an API but is directed to evaluating duplicate VMs, not snapshots, is incorrectly blurred. To put it simply, interactions between the computing device 502 and other components in Figures 5A-5B of Veselov, the only examples of Veselov that analyze a snapshot, **do not** use APIs, contrary to the express requirements of the claims, and there is no motivation or assertion in Dr. Stavrou’s declaration that Veselov would be modified to take components that are present in Figures 3A-3B and import them into Figures 5A-5B, where such components do not exist.

102. Dr. Stavrou acknowledges that “Veselov describes several different embodiments and figures” but dismisses any differences between them by asserting that “a POSA would have understood that the techniques and components described in the context of a particular figure can also applied [*sic*] in the context of the other embodiments and figures.” Ex.1002 (Stavrou Decl.), ¶91. But even if a technique or component “can” apply to another embodiment or figure, that does

not mean the specific techniques or components that Dr. Stavrou mixes between different examples would have been understood to apply by a person of ordinary skill in the art. Dr. Stavrou's citations of Veselov support this fact. For example, although Dr. Stavrou asserts that Veselov "explains that the techniques discussed in the context of Fig. 2 can be performed by the various systems described in other embodiments," he only quotes Veselov as teaching that "[t]he process 200 may be performed by any *suitable* system such as a scanning service or component thereof as described above in connection with FIG. 1." Ex.1002 (Stavrou Decl.), ¶91 (quoting Veselov 8:52-62). But Dr. Stavrou makes no attempt to show that, e.g., the Figs. 5A-5B systems are "suitable" for the modifications he applies from the Figs. 3A-3B examples. Nor could he, because as described above, the Figs. 5A-5B examples do not employ a computing resource service provider or APIs. *See* Ex.1007 (Veselov) 10:37-39 (Figures 3A-3B use "computing environments implemented by a computing resource service provider"), 16:56-57 (Figs. 5A-B illustrate "additional embodiments" without computing resource service providers or APIs).

103. I have been informed and understand that in an obviousness analysis, it is not appropriate to mix disclosures from multiple references, or multiple examples in one reference, without adequate explanation for why a person of

ordinary skill in the art would have done the same. Dr. Stavrou has not provided such explanation.

b. Dr. Stavrou's proposed motivations to combine Veselov and Hufsmith are not based on what a person of ordinary skill in the art would have understood at the relevant time.

104. Dr. Stavrou asserts that a person of ordinary skill in the art would have been motivated to make the combination of Veselov and Price on which all of Petitioner's grounds depend. *See* Ex.1002 (Stavrou Decl.), ¶102. I disagree. To a person of ordinary skill in the art, Veselov and Price have unrelated goals and contradictory teachings, and a person of ordinary skill in the art with Veselov in hand would not have been motivated to combine the teachings of Price as Petitioner and Dr. Stavrou assert for their challenges to the '031 patent claims. In view of these references' distinctions, which I discuss in more detail below, it appears to me that Petitioner and Dr. Stavrou are working backwards from the '031 patent's claims, looking for disclosures in disparate references, without meaningfully considering whether the references they cite would, in fact, be combined in the first place. As noted above, I understand that obviousness arguments based on this type of hindsight are improper.

105. As discussed above in Section VII.A, Veselov's security systems and methods are specifically designed "to avoid consuming processing time, processing power, and storage space" of target VMs, so the active VMs that

Veselov's systems and methods target can continue to "servic[e] requests from [] customers or perform[] other useful work" without any downtime. Ex.1007 (Veselov), 12:21-43. Price, in contrast, discloses two different processes for assessing virtual machines—(1) an embodiment where the target virtual machine is inactive or "offline," in which case a VM image is collected, and (2) an embodiment where active ("online") target virtual machines are loaded with agents to perform security assessments. *See* Ex.1048 (Price), ¶¶11-12, Fig. 6; Section VII.B.

106. Dr. Stavrou asserts that a person of ordinary skill in the art would have been motivated to combine Veselov with Price "to obtain well-known benefits of, in certain situations, taking a snapshot of the target VM at rest and performing the snapshot analysis while the VM is inactive." Ex.1002 (Stavrou Decl.) ¶105. I disagree because this supposed benefit runs contrary to Veselov's overarching goal, discussed above, of keeping VMs online and avoiding downtime. Veselov does not mention offline VMs. A person of ordinary skill in the art would have understood that, in order to combine Veselov and Price in the proposed manner, Veselov's online VMs (which Veselov explicitly aims to keep online) would need to be shut down and taken offline to use Price's offline assessments. This runs counter to Veselov's express goals, and a person of ordinary skill in the art would

not have incorporated Price's teachings for at least this reason. *See, e.g.*, Ex.1007 (Veselov), 2:58-63 12:21-43.

107. A person of ordinary skill in the art also would not have considered combining Price's teachings because its use of agents on active ("online") VMs is contrary to Veselov's teachings. Specifically, Veselov disparages "existing security assessment systems" that add "complexity of installation and monitoring of the assessment system." Ex.1007 (Veselov) 2:42-46. This is what Price does with its active VMs—adds complexity by installing agents that monitor the system. Price Abstract, Figure 6; Ex.2077 (Stavrou Tr.) 245:3-11 (Dr. Stavrou admitting Price uses agents for active "online" VMs). Such agents were prevalent in the art, but their negative impact on the performance of a VM was well-known. *See, e.g.*, Ex.2003 [Trend Micro Enterprise Security, *Virtual Datacenter*], 3 ("Resource-intensive operations such as regular anti-virus scans and pattern file updates can quickly result in an extreme load on the system. When anti-virus scans or scheduled updates simultaneously kick into action on all virtual machines on a single physical system, the result is an 'anti-virus storm'. This 'storm' is like a run on the bank, where the 'bank' is the underlying virtualized resource pool of memory, storage and CPU. Server applications and virtual desktop/VDI environments are hampered by this performance impact."); *cf.* Ex.2004 [Red Hat Virtualization 4.0], 46 (same); Ex.2026 [Wiz Marketing], 5, 7 (cautioning against

the use of agents in a cybersecurity system, providing that agent-based systems “lower system performance and lead to increased resource consumption,” and can push systems “into higher-priced deployment tiers, causing unplanned cost increases that lead to budget overruns”); Ex.2019 [Uluski Thesis], 38 (“Previous studies [] have taken measures to quantify how much overhead is introduced by the execution of a real-time anti-virus scanner and have indicated that significant amounts of pressure is place on not only the central processor, but the memory subsystem. ... The issue of pressure on the physical system becomes more immense in the ESX server environment, where multiple virtual machines are executed on one physical machine, and each virtual machine running the windows operating system requires its own instance of an anti-virus program.”); See Ex.2008 [Mrdovic, *Combining Static and Live Digital Forensic Analysis*], 2 (discussing the drawbacks to the traditional “live analysis.”). That is, a person of ordinary skill in the art would have understood that incorporating Price’s agent-based monitoring of its online VMs with Veselov’s teachings would have required installation of those agents, a slowdown of Veselov’s target VMs, and interference with those target VMs — all consequences that would not be desirable in the context of Veselov itself, and thus a person of skill in the art would be discouraged from making a combination of Veselov and Price.

108. Setting aside Price's use of agents on active VMs, a person of ordinary skill in the art reviewing Veselov would have also had no reason to combine or incorporate Price's techniques for assessing inactive VMs into Veselov's systems and methods. Veselov does not discuss the security of inactive VMs, at all. *See* Ex.2077 (Stavrou Tr.) 203:21-205:7 ("Q. There is no reference in Veselov Exhibit 1007 to a virtual machine being offline when the snapshot is taken, true? A. I believe that is correct.").

109. Rather, a person of ordinary skill in the art would have known that Veselov's snapshot-based systems and methods and Price's image-based techniques are different and would not be readily combined. Price does not mention snapshots. Ex.1002 (Stavrou Decl.), ¶104 ("Price's offline analysis is performed on VM *images*"). A VM's snapshot is, as the '031 patent states, "a copy of the machine's virtual disk (or disk file) at a given point in time." Ex.1001 ('031 patent), 5:35-36. A VM image, in contrast, is a data file that "include[s] an installed and configured guest operating system and applications" and is used for running a virtual machine. Ex.2018 [Abraham Silberschatz et al., *Operating System Concepts* (9th ed. 2013)], p.716; *see also, e.g.*, Ex.2005, [US Pat. App. Pub. No. 2008/0320583], ¶¶160-63 (distinguishing a "disk image file," which is "a virtual disk file that stores the contents of the virtual machine's hard disk drive," and a "snapshot file," which is "a file that stores the running state of the VM at a

particular moment in time, usually in terms of deltas with respect to a baseline state.”); Ex.2020 [Huawei, *Snapshots vs. Images*] (explaining that images “[c]an’t be created for running instances” whereas snapshots “record[] the state and data of the virtual machine at the moment the snapshot was taken”); *Compare* Ex.2006 [Red Hat Virtualization 4.3 – *Virtual Machine Snapshots*] (“Snapshot are a storage function that allows an administrator to create a restore point of a virtual machine’s operating system, applications, and data at a certain point in time.”) and Ex.2007 [Birk, *Technical Challenges*], 4 (“Snapshots, also referred to as forensic image, of virtual machines provide a powerful tool with which a virtual machine can be cloned by one click including also the running system's memory. This leads to the main benefit that systems hosting crucial business processes do not have to be shutdown for performing a forensic analysis.”) *with* Ex.2025 [Red Hat Open Stack], p. 5 (“A virtual machine image is a file that contains a virtual disk which has a bootable operating system installed on it.”).

110. Additionally, a person of ordinary skill in the art would have known that snapshots and images are used to address different classes of vulnerabilities. For example, a point-in-time snapshot of a computer system disk is known to allow capturing and cleaning of files infected with malware at the time when the snapshot was taken. Ex.2023 [U.S. Pat. No. 8,220,053 B1 (“Sun”)], 5:49-58. In contrast, because VM images are used to instantiate VMs, scanning, detecting, and

removing of malware from offline VM images would prevent the spread of malwares. Ex.2024 [Wei, *Managing Security of Virtual Machine Images*], Sec. 2.3 (“Anecdotal evidence shows that, if dormant VM images are not managed (e.g., scanned for worms), a virtual environment may never converge to a steady state, because worm-carrying VM images can sporadically run, infect other machines, and disappear before they can be detected”).

111. Dr. Stavrou also recognizes that snapshots and images are not the same thing. *See, e.g.*, Ex.1002 (Stavrou Decl.) ¶ 37 (“Snapshots *can be more risky than images* because snapshots contain the contents of RAM memory at the time that the snapshot was taken, and this might include sensitive information that was not even stored on the drive itself.”) (emphasis added) (quoting Ex.1018).

112. A person of ordinary skill in the art thus would have understood that Price’s techniques are **not** “analogous to those in Veselov,” as Dr. Stavrou asserts. Ex.1002 (Stavrou Decl.), ¶104. Contrary to Dr. Stavrou’s assertions, it would not have been straightforward or routine to apply Price’s teachings of analyzing VM images to Veselov’s teachings of analyzing a snapshot of a VM. Instead, a person of ordinary skill in the art would have recognized that Price’s teachings were not compatible with those of Veselov, and would not have sought to combine the teachings of the references as Dr. Stavrou asserts.

113. Thus, I disagree with Petitioner's and Dr. Stavrou's assertions that a person of ordinary skill in the art would have made the Veselov-Price combination, and that the combination "would have achieved predictable results." Ex.1002 (Stavrou Decl.), ¶104. A person of ordinary skill in the art would not have made this combination, including because the proposed combination would have been unpredictable and, as noted, contradictory.

114. Dr. Stavrou cites a number of references in addition to Veselov and Price, purportedly to show that a person of ordinary skill in the art would have been motivated to combine Veselov and Price. Ex.1002 (Stavrou Decl.) ¶¶104-107. But, as I explain in detail below, none of the references Dr. Stavrou relies on explain why a person of ordinary skill in the art—who would understand the important differences between Veselov and Price discussed above—would combine Veselov's systems and methods with Price's techniques. Critically, none of the references that Dr. Stavrou relies on establishes that a person of ordinary skill in the art would have been motivated to shut down Veselov's target VMs to take a snapshot, when shutting down Veselov's target VMs would run directly counter to Veselov's goals.

115. Dr. Stavrou cites Zhou (Ex.1066) and Jeswani (Ex.1067), arguing that "a POSA would have expected many VMs to already be at rest and inactive when [an] assessment begins." Ex.1002 (Stavrou Decl.), ¶104 (relying on Ex.1066 and

Ex.1067). But, even assuming that is true, that does not show that a person of ordinary skill in the art would have combined Veselov's snapshot-based systems and methods for assessing active VMs with Price's VM image-based techniques. This is true for many reasons, including that Veselov focuses on maintaining VM activity, such that the Veselov VMs are not "at rest and inactive" as Dr. Stavrou asserts.

116. Further, Zhou discloses the use of VM imaging for applying patches to offline VMs and does not analyze an offline VM for security vulnerabilities. Zhou is directed to the offline patching of virtual machine images and invokes materially different processes than the disputed technology. For example, Zhou explores a technology called "Nüwa" that applies patches to *images* for virtual machines in an inactive state. Ex.1066, 1. But, Zhou makes clear that Nüwa is an operative tool for virtual machine *images* – not snapshots. Ex.1066, 1. Additionally, Zhou distinguishes between online and offline patch activity, providing that whether a machine is patched online or offline depends on whether a command line impacts *memory*. Ex.1066, 5 ("Safe command lines do not depend on or impact memory. These are the commands that can and should be executed offline."). At base, Zhou's disclosures for patching online and offline machines depending on impact to memory have no bearing on Veselov's snapshot-based

analysis for vulnerability assessments of active machines – much less, the '031 patent claims' vulnerability assessment for at rest machines.

117. Additionally, Jeswani, which is directed to patch updates for dormant and active virtual machines, actually undermines Petitioner's and Dr. Stavrou's proposed combination, as it discloses that snapshots are implemented for creating manifests of *online* instances (Ex.1067, 393-94) whereas, in contrast, *images* are used for offline machines (Ex.1067, 398-99). This further demonstrates that a person of ordinary skill in the art would recognize the meaningful differences between Veselov's snapshots and Price's images discussed above. And, Jeswani's disclosure that "powering on VMs that haven't been used for several months could expose [a] datacenter to security threats" (Ex.1067, 399), which Dr. Stavrou emphasizes (Ex.1002 (Stavrou Decl.), ¶105), does not show that a person of ordinary skill in the art reviewing Veselov's snapshot-based systems and methods for active machines would turn to Price's disparate techniques for assessing offline VM images, or that a person of ordinary skill in the art reviewing Veselov would be motivated to take Veselov's VMs offline.

118. Furthermore, the fact that Zhou and Jeswani acknowledge that costs and delays can occur when an offline VM is activated, *see* Ex.1002 (Stavrou Decl.), ¶106 (citing Zhou and Jeswani), does not show that a person of ordinary skill in

the art would have combined Veselov—which does not mention inactive or offline VMs at all—with Price's image-based techniques.

119. The Budko reference that Dr. Stavrou cites likewise does not support a Veselov-Price combination; Budko does not mention snapshots, and it merely discloses that VMs that were recently exposed to vulnerabilities may require offline scanning and quarantining. Ex.1081 (Budko), ¶¶ 29-31, 37, 69.

120. Likewise, the Mirage reference (Ex.1073) and vSphere reference (Ex.1015) that Dr. Stavrou cite fail to show that a person of ordinary skill in the art would have combined, or “would have had a reasonable expectation of success in combining,” Veselov and Price, as Dr. Stavrou asserts. Ex.1002 (Stavrou Decl.), ¶107, ¶33. The Mirage reference (Ex.1073) makes no mention of taking “snapshots” of offline VMs, and it confirms that a person of ordinary skill in the art would have expected to use images for assessments of offline systems. *See* Ex.1073, 1 (“In addition, Mirage maintains a provenance tree that records how each image was derived from other images; allows operations, like patching and scanning, that normally require a VM instance to execute offline; and enables analyses such as image search and comparison.”). The vSphere reference (Ex.1015) is the only reference of these that discloses that snapshots can be taken when a VM is powered off (Ex.1015 at 164), but this further demonstrates the gap between snapshots and images, and there is no suggestion in this reference to use

such snapshots for security assessments, or a suggestion to use offline snapshotting in Veselov's context. To be clear, for reasons discussed above, it is my opinion that a person of ordinary skill in the art would not have reasonably expected to succeed in making Petitioner's and Dr. Stavrou's Veselov-Price combination.

121. It is also my opinion that Wiz's contention that a person of ordinary skill in the art would have reasonably expected to succeed in making its and Dr. Stavrou's proposed Veselov-Price combination is based on mischaracterizations of Veselov. For example, contrary to Dr. Stavrou's positions, nothing in Veselov discloses or suggests that "state of the target VM could be inactive or at rest when the snapshot is taken and while the snapshot analysis is being performed" – nor does Veselov provide that its techniques for snapshots require no interaction with the target VM. Ex.1002 (Stavrou Decl.), ¶107. Dr. Stavrou's reliance on certain disclosures of Veselov (Ex.1002 (Stavrou Decl.), ¶107 (citing Ex.1007 (Veselov), 16:45-53, 18:32-53, Figs. 3A-B (assessment VM), 5A-B (direct analysis of snapshot data)) is misplaced. For example, Veselov's snapshot-generation techniques actually interfere with the operation of the target VM. Specifically, Veselov teaches that generating the snapshot "may require that all writes to the disk are queued" and may "cause the writes to disk to be deferred or otherwise stalled until generation of the snapshot is completed." Ex.1007 (Veselov), 6:50-55. And, as above, Veselov requires that the machine maintains its active state.

See, e.g., Ex.1007 (Veselov), 2:56-67, 3:8-11, 3:22, 12:21-43. And nothing in Veselov discloses or suggests that “the original VM can be inactive or at rest ... while the snapshot analysis is being performed,” or while the snapshot is taken, as Dr. Stavrou contends. Ex.1002 (Stavrou Decl.), ¶107. Instead, Veselov’s analysis requires that the snapshot capture the active state of the machine and that the client activity not be interrupted. *See, e.g.*, Ex.1007 (Veselov), 2:56-67, 3:8-11, 3:22, 12:21-43. As I explain above, Veselov’s overarching goal is to avoid downtime of VMs and reduce assessment complexity when performing security assessments, and to preserve the activity of the original virtual machine instance. Dr. Stavrou’s contention that the “original VM can be inactive or at rest” (Ex.1002 (Stavrou Decl.), ¶107) is directly contrary to Veselov’s purpose and goals which maintain activity of the original VM such that it is not at rest.

122. For the foregoing reasons, it is my opinion that Petitioner and Dr. Stavrou have failed to show—because they cannot show—that a person of ordinary skill in the art would have made the Veselov-Price combination on which every ground in the Petition depends. A person of ordinary skill in the art would not have made and would not have reasonably expected to succeed in making, the Veselov-Price combination.

2. Even If Combined, Veselov and Price Do Not Disclose or Suggest Multiple Limitations in the Independent Claims

123. Even if a person of ordinary skill in the art would have made the Veselov-Price combination that Dr. Stavrou and Petitioner urge, it is my opinion that none of the claims of the '031 patent would have been obvious over that combination (or other combinations depending on this combination) at least because Veselov and Price do not disclose or suggest multiple limitations in the claims.

a. Veselov Does Not Render Obvious Using Cloud Computing Platform APIs to Identify and Locate Virtual Disks (Limitations 1.2-1.4, 9.2-9.4, 16.2-16.4)⁴

124. Limitations 1.2-1.4 (and their corresponding limitations in claims 9 and 16) recite features relating to using “cloud computing platform APIs” to identify virtual disks of a virtual machine, as well as to query and receive the location of at least one such virtual disk. The use of the cloud computing platform APIs for these functions facilitates the taking or requesting the taking of snapshots and the analysis of such snapshots for vulnerabilities as recited in other limitations, discussed in subsequent sections.

⁴ In this declaration, I use the claim limitation labels that Dr. Stavrou and Petitioner used. A listing of the claims of the '031 patent, with those labels, is presented at the beginning of the Petition, in a “Listing of Challenged Claims.” Pet. at iii-vi.

125. Specifically, limitations 1.2, 9.2, and 16.2 recite, “using the interface, utilize cloud computing platform APIs to identify virtual disks of a virtual machine in the client environment.” The “interface” referred to in these limitations is the “interface between a client environment and security components” recited in limitations 1.1, 9.1, and 16.1. Limitations 1.3, 9.3, and 16.3 recite, “use the computing platform APIs to query a location of at least one of the identified virtual disks.” And limitations 1.4, 9.4, and 16.4 recite, “receive an identification of the location of the virtual disks of the virtual machine.”

126. Dr. Stavrou and Petitioner do not rely on any disclosures from Price for these limitations (*see, e.g.*, Ex.1002 (Stavrou Decl.), ¶¶121-141), and it is my opinion that Veselov does not teach or suggest these limitations.

127. First, Veselov does not disclose “utilizing cloud computing platform APIs to identify virtual disks of a virtual machine.” Dr. Stavrou asserts that “Veselov teaches that [its] scanning service uses [a communication] interface to ‘identify virtual disks of a virtual machine in the client environment’ in order to facilitate its generation of a snapshot of such disks.” Ex.1002 (Stavrou Decl.), ¶122. But the disclosures on which Dr. Stavrou relies simply do not disclose or suggest an *identification of a virtual disk* of a virtual machine made using a cloud computing platform API, as claimed. *See* Ex.1002 (Stavrou Decl.), ¶122 (citing Ex.1007 (Veselov), 6:30-38, 8:60-9:41). For example, Dr. Stavrou cites the

following passage in column 6, which refers to instances of a VM but says nothing about identifying a virtual disk:

“In some embodiments, a desired security assessment may be initiated by receipt of a request. For example, a user device 102 may transmit a request to the scanning service 110 for a particular security assessment of one or more instances of *a virtual machine 112*. The request may be an API call including information corresponding to the requestor, the user associated with the virtual machine 112, the on-demand storage service 108, and/or the particular virtual machine 112.”

Ex.1007 (Veselov) 6:30-38 (italics added). And the nearly column-long passage in columns 8 and 9 that Dr. Stavrou cites discusses “a virtual machine instance,” that “instance itself,” or “a logical volume attached to the corresponding virtual machine and/or containing the instance.” Ex.1007 (Veselov) 8:60-9:41. And although Veselov refers to an identifier for a VM, it never mentions any type of identifier for, or identification of, a “logical volume,” and a VM identifier does not identify a virtual disk.

128. Additionally, as I discussed above in Section VIII.A.1, Dr. Stavrou mixes disclosures from different examples in Veselov without providing actual motivation to do so, and a person of ordinary skill in the art would not have viewed the various features and disclosures of those different examples to be simply interchangeable. Examples (Figures 1, 3A, 3B) employ “computing environments

implemented by a computing resource provider 300, 350” which may use an API to provide a user with access to scanning services within virtual computing environments, but do not locate disks or analyze snapshots, and a person of ordinary skill in the art would have had no reason to add such features to those examples as I discussed previously. Ex.1007 (Veselov) 10:37-41, 14:37-41, 15:23-27, Figs. 3A-3B; *see also* Section VIII.A.1, above. And the use of APIs in these examples relate specifically to providing interfaces to a user, conveying user-specific data, or allowing users to specify information for the system, but never for identifying or locating a virtual disk—nor is there reason to use such APIs to identify or locate a virtual disk at all in these examples.

129. By contrast, the Figs. 5A-5B examples are not implemented with computing resource service providers⁵ or an API, but rather implement scanning services through a provisioning service such as a hypervisor. Ex.1007 (Veselov) Figs. 5A-5B, 15:56-17:66. But as I discussed above, Dr. Stavrou does not explain why a person of ordinary skill in the art would have viewed elements of those two separate examples as interchangeable and compatible or would have wanted to combine or modify them. Veselov therefore does not render these limitations

⁵ As stated above, I note that “computing resource service provider” is not typically used in the art to refer to a cloud computing environment.

obvious for the same reasons I discuss in Section VIII.A.1 relating to a person of ordinary skill in the art not mixing these examples in the manner Dr. Stavrou proposes.

130. Veselov does not disclose “utiliz[ing] cloud computing platform APIs” for identifying, querying, and receiving virtual disk location information. To the contrary, Veselov’s APIs provide interfaces to a user or otherwise convey user-specific data. *See generally* Ex.1007 (Veselov) 4:29-48, 5:1-8, 5:13-22, 6:34-38, 6:59-60, 8:46-51, 14:37-53, 15:23-38. For example, relating to Fig. 1, Veselov describes that “computing resource service provider 104 may provide, or otherwise be compatible with, an environment application programming interface (API) 120 through which a user device 102 can connect to one or more virtual computing environments of the computing resource service provider.” Ex.1007 (Veselov) 4:29-48. Additionally, “the scanning service 110 may use the API 120 to provide user interfaces to the user device 102, enabling the user to configure settings of the scanning service 110.” Ex.1007 (Veselov) 5:1-8. The user can “provide some or all of the snapshot data 146 to the scanning service” using the API 120. Ex.1007 (Veselov) 5:13-15. Dr. Stavrou handwaves over these descriptions of Veselov’s APIs and Veselov’s usage of the terms “environment” or “web interface” APIs to describe them, which is consistent with their limited usage in Veselov, and instead attempts to broadly attribute API usage to multiple unrelated aspects of Veselov,

which is not how a person of ordinary skill in the art would have understood Veselov. Ex.1007 (Veselov) 4:29-48, 13:37-41.

131. Dr. Stavrou asserts that “[u]sing the interface to utilize cloud computing platform APIs to identify the virtual disks is further taught or suggested by Veselov’s discussion of tailoring snapshots.” Ex.1002 (Stavrou Decl.), ¶128. Dr. Stavrou cites two examples of what Dr. Stavrou asserts are explanations by Veselov “that the scanning service can determine what elements will be included in the snapshot.” Ex.1002 (Stavrou Decl.) ¶128 (citing Ex. 1007 (Veselov) 8:16-45, 14:37-57). Neither of these passages disclose using cloud computing platform APIs to identify (let alone query or receive a location of) virtual disks. Rather, they disclose identifying “data elements of the virtual machine” that are needed to “perform the security assessment” (Ex.1007 (Veselov) 8:16-45) and sending “assessment information” to the “user device 302 to be displayed, selected, changed, etc.” (Ex.1007 (Veselov) 14:53-57), neither of which suggest to a person of ordinary skill in the art to modify Veselov to perform the claimed functions utilizing cloud computing platform APIs.

132. Dr. Stavrou also relies on a handful of references other than Veselov, including Ex.1051 (Workstation Pro), Ex.1053 (Prahlad), and Ex.1020 (Sancheti), purportedly to show that identifying virtual disks was “routine[]” or “commonplace.” Ex.1002 (Stavrou Decl.), ¶¶122, 127-128. But Dr. Stavrou does

not identify anything in those references that would explain why a person of ordinary skill in the art would have been motivated to change Veselov's systems and methods to identify virtual disks of a VM utilizing cloud computing platform APIs, as claimed, as opposed to just identifying a VM generally. Veselov's systems and methods lack such identifications, and I have been informed and understand that, in an obviousness analysis, merely collecting disparate disclosures from a handful of references, to purportedly show that a person of ordinary skill in the art *could have* made the claimed inventions because such disclosures allegedly show that each piece of a claim was known or routine or commonplace is not enough. Rather, I understand that obviousness analyses look to what a person of ordinary skill in the art *would have* been motivated to make or do. In my opinion, nothing in the other references Dr. Stavrou cites show that a person of ordinary skill in the art would have been motivated to change Veselov's systems and methods, at all, let alone to modify those systems and methods as claimed in the '031 patent's claims.⁶ In fact, this is an example of where, in my view, Dr. Stavrou is improperly working backwards from the '031 patent's claims.

⁶ Dr. Stavrou's tertiary references do not demonstrate that the claimed features are routine or commonplace, and Dr. Stavrou does not explain how or contend that these disclosures would be used to modify Veselov. The disclosure Dr. Stavrou

133. Dr. Stavrou also asserts that in Veselov's system and methods, "the user's request may identify a VM that will be the target of the scan, and the scanning service may then be tasked with taking a snapshot containing multiple virtual disks associated with that VM." Ex.1002 (Stavrou Decl.), ¶124. But the fact that Veselov's systems and methods work with and from the identification of

cites from the Workstation Pro reference (Ex.1051) simply discloses excluding virtual disks from snapshots and makes no mention of using location data or APIs. Ex.1051 (Workstation Pro), 125. The VirtualCenter reference (Ex.1052), relied on in Ex.1002, Stavrou Decl. ¶ 127, shows that there may be an identification for a virtual machine location, but does not demonstrate using cloud computing APIs to identify virtual disk locations. Ex.1052 (VirtualCenter), 445-46. The disclosures Dr. Stavrou cites from Prahlad (Ex.1053) similarly does not demonstrate the use of the '031 patent's cloud computing APIs, nor is it relied upon in Dr. Stavrou's grounds. Ex.1053 (Prahlad), ¶¶81-82, 87-88, 92. Nor does Ex.1020 (Sancheti) demonstrate routineness of identifying virtual disks—in fact, its disclosure (which is not part of Dr. Stavrou's proposed grounds), supports that merely identifying a VM does not identify a virtual disk. *See* Ex.1002 ¶127 (quoting Sancheti 21:36-41). And Sancheti does not disclose identifying virtual disks for the purpose of security assessments.

VMs (*not virtual disks*) belies Dr. Stavrou's assertions about the virtual disk identification limitations. Dr. Stavrou offers no explanation for why a person of ordinary skill in the art would have changed those systems and methods to instead identify *virtual disks* of a VM, let alone identify such virtual disks using cloud computing platform APIs, as claimed. And, even if a Veselov VM can "contain[] multiple virtual disks" as Dr. Stavrou states, *see* Ex.1002 (Stavrou Decl.), ¶124; *see also* Ex.1007 (Veselov), 6:7-10, 11:28-33, 17:35-37, that still fails to show that the claimed virtual disk identification would have been obvious. As the '031 patent explains and shows, virtual disks in cloud computing platforms associated with VMs can be located apart from those VMs. *See* Section VI.A, *see also, e.g.*, '031 patent, Fig. 1B, 4:21-25 ("[S]torage emulat[ing] virtual discs for the VMs" "may be part of the server" hosting VMs, or, as depicted in Figure 1B, separate from but "connected to" that server). Thus, the identification of a VM does not necessarily or obviously identify a virtual disk of that VM.

134. Dr. Stavrou further asserts that Veselov "suggests" the claimed identification of virtual disks because Veselov's "API-based UI" can be used "to select assessment targets and other assessment parameters of [a] scan" and "remotely manage the user's resources." Ex.1002 (Stavrou Decl.), ¶¶129-131. As an initial matter, Veselov discusses the user identifying "the data elements of the virtual machine 112 that are needed to perform the security assessment" – not that

API elements may be used for virtual disk locations. Ex.1007 (Veselov), 8:16-45. Nothing in Veselov suggests that a “data element” is a virtual disk. And, in fact, Veselov suggests otherwise, because in other disclosures, virtual disk storage is referred to as a “virtual computer component.” So, the tailoring snapshots discussion does not teach that Veselov identifies virtual disks. But, in any event, none of the disclosures Dr. Stavrou cites actually teaches or suggests the claimed identifying virtual disks using cloud computing platform APIs. Dr. Stavrou cites Veselov at 2:22-3:19, but that disclosure merely cautions that security assessments on distributed environments (such as cloud environments) could detract from the target machine’s “normal workload” and explains the need to provide for security assessments that do not interfere with the activity of the target virtual machine. Ex.1007 (Veselov), 2:22-3:29. Dr. Stavrou also cites Veselov at 19:30-37, but that disclosure only addresses Veselov’s definition of “data store” in that it involves accessing and retrieving data with various components. Ex.1007 (Veselov), 19:30-37. Dr. Stavrou also cites Veselov at 19:60-64, but that disclosure merely provides that single or multiple devices may perform operations. Ex.1007 (Veselov), 19:60-64. Dr. Stavrou also cites Veselov at 20:46-56, but that disclosure merely states that one embodiment of Veselov is a distributed or virtual environment. Ex.1007 (Veselov), 20:46-56. Dr. Stavrou also cites to Veselov at 4:19-48, 5:1-8, 14:41-53, and 14:53-15:29, but these disclosures merely highlight Dr. Stavrou’s improper

combination of distinct embodiments. That is, Veselov at 4:19-48, 5:1-8, 14:41-53, and 14:53-15:29 make clear that its APIs exist for a user to provide input to the scanning environment or to allow a scanning service, outside of the computing resource environment, to access resources within the computing resource environment. Ex.1007 (Veselov), 4:19-48, 5:1-8, 14:41-53, 14:53-15:29. Accordingly, Veselov's disclosures, taken individually or in an unclear amalgamation (i.e., as Dr. Stavrou has done), do not teach or suggest identifying virtual disks using cloud computing platform APIs.

135. Regarding limitations 1.3, 9.3, and 16.3, Dr. Stavrou does not identify any disclosures in Veselov that actually relate to using cloud computing platform APIs to query a location of at least one virtual disk. Nor could he. Veselov only discusses queries in the context of querying a "snapshot storage service" to find a stored snapshot. Ex.1007 (Veselov), 7:34-61. Veselov does not contain any other disclosures relating to queries. Additionally, Veselov teaches that, with reference to the snapshot storage service, "volume images" are "non-queryable" or queryable only based on the VM's identifier or the snapshot's identifier—neither of these teach or suggest querying the location of a virtual disk. Ex.1007 (Veselov), 7:55-61.

136. Dr. Stavrou asserts that "[i]t would have been apparent to a POSA that, for the scanning service to generate a snapshot of certain virtual disks... the

scanning service would first determine the location of the disks it will be including in the snapshot.” Ex.1002 (Stavrou Decl.) ¶134. I disagree. Dr. Stavrou cites to Veselov at 9:9-25, but there is no mention at all of the use of APIs and no indication that “executing the snapshot itself” would use any APIs. Specifically, Veselov discloses:

At step 204, the scanning service may optionally obtain scan data describing the parameters of the security assessment to be performed. For example, the signal may include an identifier of a security assessment rules package, and the scanning service may retrieve the rules package from a scan data store. At step 206, the scanning service may obtain snapshot data representing a state of the target resource at the time the snapshot was captured. In various non-limiting examples, the scanning service may obtain the snapshot data by *executing the snapshot itself*, by requesting another service to capture the snapshot and provide it to the scanning service, by accessing a data store (e.g., the snapshot storage service) containing the snapshot, etc.; alternatively, the scanning system may obtain access to the snapshot data, such as at a storage location of the snapshot data, but may not obtain (i.e., make a copy of) the snapshot data.

Ex.1007 (Veselov) 9:9-25.

137. Veselov only discloses querying a “snapshot storage service” to find a *stored* snapshot based on one or more of either a “VM identifier,” “identifier of the virtual machine instance,” “file name,” or “key” (Veselov 7:43-55), none of

which require any query for a location of a virtual disk. The claim requires using “the computing platform APIs to query a location of at least one of the identified virtual disks.” In other words, because Veselov queries the snapshot storage service for a stored snapshot, no further querying for a location of a **virtual disk** is needed to take a snapshot or obvious and thus Veselov does not suggest using the claimed cloud computing platform API to query for the location of a virtual disk.

138. Dr. Stavrou again cites a handful of references other than Veselov, asserting that querying for locations of virtual disks was “standard,” “routine[,],” and “commonplace.” Ex.1002 (Stavrou Decl.), ¶¶134-135. But, as with limitations 1.2, 9.2, and 16.2, with limitations 1.3, 9.3, and 16.3, Dr. Stavrou does not identify anything in the references he cites that would explain why a person of ordinary skill in the art would have been motivated to change Veselov’s systems and methods to use cloud computing platform APIs to query a location of at least one virtual disk, as claimed. Veselov’s systems and methods lack such querying, and in my opinion, nothing in the other references Dr. Stavrou cites show that a person of ordinary skill in the art would have been motivated to change Veselov’s systems to include it.⁷ In my view, this is another example of where Dr. Stavrou

⁷ As I note in the preceding discussion, the VirtualCenter reference (Ex.1052) discusses properties for VMs but does not demonstrate queries for virtual disk

is improperly working backwards from the '031 patent's claims, and attempting to fill gaps in Veselov by asserting limitations are "standard" or "routine" despite his references not supporting those assertions.

139. Dr. Stavrou also asserts that Veselov teaches or suggests querying a virtual disk's "more general location" because it discusses an "endpoint/virtual network address of the VM." Ex.1002 (Stavrou Decl.), ¶136. Veselov, however, merely discloses that a "virtual machine 312 may be associated with a virtual network address." Ex.1007 (Veselov), 10:48-11:4. Veselov does not mention a virtual network address anywhere else, let alone querying for such an address, or querying for a location of a virtual disk. A person of ordinary skill in the art would

locations. Ex.1052 (VirtualCenter), 445-46. The Virtual Disk reference (Ex.1055) makes no use of *queries* for virtual disk *locations*. Ex.1055 (Virtual Disk), 55-56. Ex.1020 (Sancheti) discloses a "management component" that "discovers and identifies the virtual disks" in a manner unlike anything in Veselov, and Dr. Stavrou does not propose to combine Sancheti with Veselov or otherwise modify Veselov according to Sancheti. Sancheti 9:36-41. Finally, the VMware Infrastructure reference (Ex.1074) excerpted by Dr. Stavrou only refers to a "virtual machine location" and thus does not teach *queries* for *virtual disk locations*.

have known that a virtual network address associated with a VM is not the same as a location of a virtual disk, including for reasons I discussed above (e.g., a VM and a virtual disk may not be located at the same location or address). Thus, simply identifying a virtual machine does not teach or correspond to identifying or locating a virtual disk. And Dr. Stavrou has not explained how or why Veselov's systems and methods would be modified to incorporate the querying of a location of a virtual disk, let alone modified to require such querying using computing platform APIs, as claimed.

140. Furthermore, for at least the same reasons Veselov does not teach or suggest the querying of limitations 1.3, 9.3, and 16.3, Veselov does not teach or suggest "receiv[ing] an identification of the location of the virtual disks of the virtual machine," as required by limitations 1.4, 9.4, and 16.4.

141. Here too, Dr. Stavrou pulls in the references other than Veselov that he cited for limitations 1.3, 9.3, and 16.3. But, as discussed above, nothing in the other references Dr. Stavrou cites shows that a person of ordinary skill in the art would have been motivated to change Veselov's systems and methods to include the claimed querying. Likewise, nothing in those references shows that a person of ordinary skill in the art would have been motivated to change Veselov's systems and methods to receive an identification of the location of a virtual disk of a VM virtual machine in response to such querying. In my view, this is yet another

example of where Dr. Stavrou is improperly working backwards from the '031 patent's claims and trying (unsuccessfully) to fill in gaps in Veselov's disclosure with respect to the limitations of the claim.

142. Dr. Stavrou also asserts that limitations 1.4, 9.4, and 16.4 would have been obvious because Veselov discusses a “virtual network address” of a VM, including in connection with the VM's “communication with other systems.” Ex.1002 (Stavrou Decl.), ¶140. But as I explained above, a person of ordinary skill in the art would have known that a virtual network address associated with a VM is not the same as a location of a virtual disk. And Dr. Stavrou has not explained how or why Veselov's systems and methods would be modified to incorporate the querying and receiving of location information for virtual disks.

143. For at least the foregoing reasons, it is my opinion that Petitioner and Dr. Stavrou have not shown that limitations 1.2-1.4, 9.2-9.4, and 16.2-16.4 would have been obvious.

b. Veselov and Price Do Not Render Obvious the Taking or Requesting the Taking of Snapshots of VMs At Rest, or Analyses of Snapshot When VMs Are Inactive (Limitations 1.5-1.6, 9.6-9.7, 16.6-16.7)

144. Limitations 1.5, 9.6, and 16.6 recite, “[perform/performing] at least one of: (i) taking at least one snapshot, and (ii) requesting taking at least one snapshot of the virtual machine at rest, wherein the at least one snapshot represents a copy of the virtual machine disks of the virtual machine at a point in time.”

Limitations 1.6, 9.7, and 16.7 recite, “[analyze/analyzing] the at least one snapshot to detect vulnerabilities, wherein during the detection of the vulnerabilities by analyzing the at least one snapshot, the virtual machine is inactive.”

145. Dr. Stavrou asserts that these limitations would have been obvious over Veselov and Price. Ex.1002 (Stavrou Decl.), ¶¶146-169. I disagree. It is my opinion that Veselov and Price do not teach or suggest these limitations.⁸

146. Dr. Stavrou asserts that Veselov discloses “(i) taking at least one snapshot.” Ex.1002 (Stavrou Decl.), ¶ 147. This assertion, however, is based on Dr. Stavrou’s and Petitioner’s incorrect interpretation of limitations 1.5, 9.6, and 16.6. As I explained above, limitations 1.5, 9.6, and 16.6 require performing at least one of taking or requesting taking at least one snapshot of a VM *at rest*, see Section VI.B.1 (“Claim Construction”), and Veselov is only focused on snapshots of executing VMs. See Sections VII.A, VIII.A.1. As Petitioner admits in the

⁸ Dr. Stavrou introduces Budko as rendering it obvious for Veselov’s snapshot capture to take place while the target virtual machine is at rest but provides no citations for Budko. Ex.1002, ¶149. Dr. Stavrou admitted that this reference to Budko was a typo during his deposition. Ex.2077 (Stavrou Tr.) 352:14-353:10. Budko does not even mention a snapshot. Instead, Budko involves “files” that hold data for an offline machine. Ex.1081, ¶¶29-30.

Petition, “Veselov does not expressly describe requesting taking a snapshot of the VM ‘at rest’ or detecting vulnerabilities while the VM is ‘inactive,’” and “does not expressly describe the state of the VM at the time the snapshot is captured,” *see* Pet.22, 38; *see also* Ex.1002 (Stavrou Decl.), ¶ 149 (“Veselov does not expressly describe the state of the VM at the time the snapshot is captured.”). Accordingly, Dr. Stavrou’s reliance on disclosures from Veselov discussing taking snapshots of live (i.e., active) VMs (see Ex.1002 (Stavrou Decl.), ¶¶147-148) are misplaced. Veselov does not teach or suggest taking or requesting taking a snapshot of a VM *at rest*. Nor does Dr. Stavrou assert that Veselov would be modified to do so (i.e., Veselov would not be modified to take a VM offline for snapshot taking purposes), as he asserted in his deposition that he does not advocate for modifying Veselov.

147. Additionally, Dr. Stavrou cannot fill this hole in his analysis with disclosures from Price. *See* Ex.1002 (Stavrou Decl.), ¶¶150-152. Price is directed to assessments of VM *images*, not snapshots. *See* Section VII.B. Price does not even mention snapshots. And taking snapshots of a VM at rest was counter to conventional wisdom. Ex.2117 at 1 (Orca “transcends traditional solutions by inspecting data at rest”);

REDACTED

see also Section IX.D (industry praise of Orca’s ability to scan unavailable/at rest

VMs). Thus, neither Veselov nor Price discloses or suggests taking or requesting taking a snapshot of a VM at rest, as claimed.

148. Furthermore, for reasons I explained above, Dr. Stavrou fails to show that a person of ordinary skill in the art would have been motivated to combine Veselov and Price to apply to VMs “at rest” as he suggests, including in view of the other references he relies on (e.g., Zhou, Jeswani, and Budko). *See* Section VIII.A.1. A person of ordinary skill in the art would **not** have been motivated to combine Price’s VM image-based techniques with Veselov’s snapshot-based systems and methods in the proposed manner. *See* Section VIII.A.1. As established above, this is the case because Veselov’s goals focus on maintaining activity of its VMs, and it would be contrary to Veselov’s goals to purposefully take its VMs offline, as would be required by Petitioner’s and Dr. Stavrou’s analysis.

149. Further, regarding Limitations 1.6, 9.7, and 16.7, Dr. Stavrou asserts that these limitations would have been obvious over Veselov because Veselov’s “analysis can be performed by analyzing an assessment VM instantiated based on the snapshot, or the snapshot data can be analyzed directly (i.e., without instantiating an assessment VM),” and that each type of analysis “satisfies the construction of analyzing the at least one snapshot.” Ex.1002 (Stavrou Decl.), ¶161. I disagree. As I explained above, Dr. Stavrou’s proposed claim construction

for the “[analyze/analyzing] the at least one snapshot” terms in the ’031 patent’s claims is incorrect because a person of ordinary skill in the art would not understand those terms to encompass evaluations of VMs instantiated from snapshots, and Dr. Stavrou’s contrary assertion is based on only an unsupported assumption Dr. Stavrou was “asked” to make—nothing more. *See* Section VI.B.2.

150. Additionally, as the Petitioner admits in the Petition, Veselov does not disclose these limitations. *See* Pet.22 (“Veselov does not expressly describe ... detecting vulnerabilities while the VM is ‘inactive’”). Dr. Stavrou asserts that “most of Veselov’s snapshot-based assessment techniques do not require interaction with the original VM,” and persons of ordinary skill in the art “would have expected many target VMs to already be inactive.” Ex.1002 (Stavrou Decl.), ¶165; *see also* Ex.1002 (Stavrou Decl.), ¶¶166-168 (similar). But, as discussed, Veselov is directed to assessments of live VMs and keeping those live VMs running without interference. *See* Sections VII.A, VIII.A.1. Thus, Dr. Stavrou’s assertion that “a POSA would have recognized that in many cases the target VM would be inactive both before and during the analysis of the snapshot” (Ex.1002 ¶165), directly contradicts Veselov’s teachings and express goals of, among other things, allowing the VM to “perform[] other useful work” during the analysis. *See* Sections VII.A, VIII.A.1.

151. Moreover, Dr. Stavrou cannot use Price to fill the gaps in his analysis relating to limitations 1.6, 9.7, and 16.7 for similar reasons as for limitations 1.5, 9.6, and 16.6. Namely, as Price does not mention snapshots or describe analyses of snapshots, it does not teach or suggest “analyzing [] at least one snapshot to detect vulnerabilities,” or doing such an analysis while “the virtual machine is inactive,” as claimed. *See* Section VII.B (“Overview of Price”). And, as noted, a person of ordinary skill in the art would *not* have been motivated to combine Price’s VM image-based techniques with Veselov’s snapshot-based systems and methods in the proposed manner. *See* Section VIII.A.1. For example, Price uses different assessment techniques for online and offline VMs, and specifically uses *images* for offline VMs. Nothing in either reference would teach or suggest to a person of ordinary skill in the art to ignore Price’s teachings and instead apply Veselov’s online-based snapshots to offline VMs.

152. For at least the foregoing reasons, it is my opinion that Petitioner and Dr. Stavrou have not shown that limitations 1.5-1.6, 9.6-9.7, and 16.6-16.7 would have been obvious in view of Veselov and Price.

c. Veselov Does Not Render Obvious the Claimed Reporting Detected Vulnerabilities as Alerts (Limitations 1.7, 9.8, 16.8)

153. Limitations 1.7, 9.8, and 16.8 recite, “[report/reporting] the detected vulnerabilities as alerts.”

154. Dr. Stavrou and Petitioner do not rely on any disclosures from Price for these limitations (*see, e.g.*, Ex.1002 (Stavrou Decl.), ¶¶170-175), and it is my opinion that Veselov does not teach or suggest these limitations.

155. Veselov does not mention alerts. Veselov discloses that its “scanning service 110 may be configured to perform security assessments and produce assessment results based at least in part on snapshot data.” Ex.1007 (Veselov), 5:9-13. And, Veselov provides that the scanning service may “process the snapshot data” and “generate one or more assessment results.” Ex.1007 (Veselov), 8:19-21, 18:53-56 (similar).

156. Dr. Stavrou incorrectly equates “communicating [] assessment results” with “reporting [] vulnerabilities as alerts.” Ex.1002 (Stavrou Decl.), ¶¶171-173. But a person of ordinary skill in the art would have understood that communicating results does not require or suggest reporting detected vulnerabilities “as alerts,” as claimed. *See, e.g.*, Ex.1001 ('031 patent) 8:60-61 (“[T]he detected cyber threats and/or vulnerabilities are reported, *for example, as alerts*”) (italics added), cls. 1, 9, 16 (“[report/reporting] the detected vulnerabilities *as alerts*.”). Veselov’s disclosure of simply “communicating” results is not enough to constitute an alert. For example, Veselov’s “assessment results” are simply “findings” resulting from “applying the rules of the assessment” that do not suggest the urgency implied by an alert. Ex.1007 (Veselov) 10:18-22. Veselov’s

“findings” would include benign data—e.g., “findings” that a VM does not have a certain vulnerability or is configured in a certain way after applying the “rules” of the assessment—that further confirms to a person of ordinary skill in the art that these results and findings are not reported as alerts. Ex.1007 (Veselov) 10:18-22. Rather, simply “communicating” results in Veselov suggests a standard report. Persons of ordinary skill in the art would have understood that alerts and reports are different and can involve different system components, capabilities, and purposes. *Compare* Ex.2010 [U.S. Pat. No. 9,467,465 (“Hibbert”)], 20:34-44 (“The **alert** module 516 may **generate an alert** based on the assessment. If one or more applications or files identified within the records of the assessment request are classified as vulnerable or highly suspicious, the alert module 516 may generate an alert. The alert may flag the vulnerability, potentially identify the exploit, indicate the degree of danger, or provide any other information. The alert may be provided in any manner. For example, the alert module 516 may provide one or more alerts via email, SMS text message, MMS, web page, intranet alert, extranet alert, or any other way.”) *with id.*, 20:12-21 (“The **report** module 514 may **generate any kind of report** indicating the results of one or more assessments. Information contained within the report may include the digital device, one or more users of the digital device, applications and files identified by the records which may be safe, vulnerable, suspicious, or unknown, or any other kind of information. The report

module 512 may provide the report to the digital device associates with the assessment, the security administrator system 220, a network administrator, or any digital device(s) or individual(s).”).⁹

157. Alerts serve to notify stakeholders of situations that require attention, and not all reported results are alerts. Dr. Stavrou agreed in his deposition that “an alert is different than a report.” Ex.2077 (Stavrou Tr.) 286:9-287:3. Dr. Stavrou’s opinions about Veselov for these limitations, however, contradict how a person of

⁹ See also, e.g., Ex.2011 [US Pat. No. 10,298,599 (“Zhang”)], 9:50-67 (providing that a system may “**report the results without** causing the browser to notify, request permission from, or otherwise **alert**, a user using the browser” because “[p]resenting an alert [] interrupts the user’s experience with the web page”); Ex.2012 [US Pat No. 9,594,912 (“Thioux”)], 10:41-48 (illustrating a specific “alert generation logic” as a subset of “reporting logic” to “generate an alert for the client device”); Ex.1023 (ITL Bulletin), at 4 (“Agencies should use CVE in their internal reports of vulnerability scans, notifications to system owners of observed vulnerabilities, and alerts identifying the vulnerabilities targeted by active exploits.”); Ex.1017 (US Pat. No. 11,106,784), 13:12-19 (“generate a report to be shown in the report interface”), 13:26-38 (“transmitting an alert (e.g., to a system admin)”).

ordinary skill in the art would understand limitations 1.7, 9.8, and 16.8, as those limitations do not simply recite “report[ing] the detected vulnerabilities.” Ex.1002 ¶171 (“Veslov further teaches reporting these vulnerabilities as alerts with its discussion of reporting ‘assessment results.’”). To the contrary, they recite that the detected vulnerabilities be reported “*as alerts*.” ’031 patent, cls. 1, 9, 16.

158. For at least the foregoing reasons, it is my opinion that Petitioner and Dr. Stavrou have not shown that limitations 1.7, 9.8, and 16.8 would have been obvious.

d. Dependent Claims 4 and 12

159. Claims 4 and 12 depend from claims 1 and 9, respectively, and recite “wherein the identification of the location of the virtual disks of the virtual machine includes a virtual address of at least one of the virtual disks.”

160. Veslov and Price do not render obvious this additional limitation for the same reasons I discussed above relative to the virtual disk identification in claims 1 and 9. *See* Section VIII.A.2.a.

161. Dr. Stavrou cites to Veslov’s disclosure of a “virtual network address of the VM” and Price’s disclosure of “location (e.g., IP or MAC address) of each enumerated virtual machine in system” and asserts that “[a] POSA would have understood such an address to be ‘a virtual address of at least one of the virtual disks.’” Ex.1002 (Stavrou Decl.) ¶180. I disagree. An address of a VM is not a

“virtual address of at least one of the virtual disks” for similar reasons that identifying a VM is not identifying a virtual disk. *See* Section VIII.A.2.a.

162. Dr. Stavrou further points to his claim 1 analysis for querying a location of a virtual disk as support for disclosure of the additional limitation in claims 4 and 12. Ex.1002 (Stavrou Decl.) ¶181. I disagree for the same reasons discussed above for that limitation. *See* Section VIII.A.2.a. Although Dr. Stavrou additionally asserts that obtaining a virtual address of a virtual disk would have been obvious “to facilitate generation of the snapshot,” he does not identify any disclosure of obtaining such an address. Ex.1002 (Stavrou Decl.) ¶181. Nevertheless, Dr. Stavrou appears to rely on an *active* VM to provide any such data about the virtual disk and does not explain how such information would be obtained in a system that takes or requests to take snapshots of VMs *at rest*.

163. For at least the foregoing reasons, it is my opinion that Petitioner and Dr. Stavrou have not shown that dependent claims 4 and 12 would have been obvious.

e. Summary of Ground 1 Opinions

164. For at least the foregoing reasons, it is my opinion that Petitioner and Dr. Stavrou have not shown that any of the independent claims of the '031 patent—claims 1, 9, and 16—as well as dependent claims 4 and 12, would have been obvious. It follows that Petitioner and Dr. Stavrou have not shown that any of the

claims depending on those independent claims, including but not limited to the dependent claims challenged in Petitioner's ground 1 (claims 3-8 and 11-15), would have been obvious. Thus, Petitioner and Dr. Stavrou have not shown that any of claims 1, 3-9, and 11-16 of the '031 patent would have been obvious.

B. Ground 2: Claims 2 and 10 Would Not Have Been Obvious Over Veselov, Price, and Hufsmith

165. Claim 2 depends from claim 1 and recites “wherein reporting the detected vulnerabilities as alerts includes indicating priority levels associated with the detected vulnerabilities.” Claim 10 depends from claim 1 and recites “wherein reporting the detected vulnerabilities as alerts includes providing indications of the detected vulnerabilities based on priority levels associated with the detected vulnerabilities.”

166. Dr. Stavrou proposes to combine Veselov, Price, and Hufsmith to render claims 2 and 10 obvious. Ex.1002 (Stavrou Decl.) ¶¶201-207. I disagree.

1. A Person of Ordinary Skill in the Art Would Not Have Been Motivated to Combine Hufsmith with Veselov and Price and Would Not Have Had a Reasonable Expectation of Success in Doing So

167. A person of ordinary skill in the art would not have been motivated to combine Hufsmith with Veselov and Price. As an initial matter, I discussed above in relation to Ground 1 that a person of ordinary skill in the art would not have been motivated to combine Veselov and Price in the proposed manner, and that

discussion applies here to Ground 2 because Ground 2 assumes the combination of Veselov and Price. *See* Section VIII.A.1.

168. Additionally, even assuming that Veselov and Price are combined, a person of ordinary skill in the art would not have combined them with Hufsmith to render the claims obvious. Hufsmith is directed to “Combined Threat Scores for *Container Images*” not *snapshot*-based assessments of *VMs*. Ex.1078 (Hufsmith), Title. Hufsmith explicitly teaches that containers and VMs are different. Ex.1078 (Hufsmith), [0026] (“Containers generally virtualize at the operating system level, in contrast to virtual machines that emulate the underlying hardware as well.”). Although a container may “execute[] within a virtual machine,” they are not the same, and an image for a container is not the same as a snapshot of a VM. *Id.* And Dr. Stavrou agreed in deposition that VMs and containers are “not the same thing” and “distinct entities.” Stavrou Tr. 306:1-308:20, *see also id.* 300:13-22, 302:2-5 (“Q. You don’t recall ever citing to a reference in Hufsmith to snapshot, though; is that fair? A. I would say it’s fair, yes.”). Dr. Stavrou asserts that Hufsmith teaches that its “teachings apply to VM images and other computer resources.” Ex.1002 (Stavrou Decl.) ¶201 (citing Ex.1078 (Hufsmith) ¶¶26, 149). I addressed paragraph 26 of Hufsmith above, and it does not disclose that its teachings apply to VM images.

169. Additionally, the disclosure in paragraph 149 relating to “virtual machine images” does not motivate the application of Hufsmith to Veselov’s *snapshots*. See Section VIII.A.1 (distinguishing images and snapshots). Further, a person of ordinary skill in the art would recognize that this disclosure of Hufsmith does not demonstrate that container-image techniques apply in general to virtual machine images. As an initial matter, Hufsmith’s techniques expressly apply to its “encapsulated functionality,” and Hufsmith does not further elaborate on the scope of the encapsulated functionality. Ex.1078 (Hufsmith), ¶149. As I discuss in Section VII.C above, Hufsmith implements tools specific to the analysis of container images with heterogenous layers. Accordingly, even reading the above disclosure from Hufsmith, a person of ordinary skill in the art would not understand that Hufsmith’s embodiments could simply be appended to virtual machine images. Hufsmith at most confirms that techniques for scanning *images* of containers might be applicable to “encapsulated functionality” such as *images* of virtual machines. Ex.1078 (Hufsmith), ¶149. Hufsmith does not contemplate that its analysis of container *images* could be substituted or combined with Veselov’s analysis of *virtual machines* to teach analysis of a *snapshot*, as Dr. Stavrou appears to suggest (Ex.1002 (Stavrou Decl.), ¶101). Instead of addressing or acknowledging these concepts and how they differ, Dr. Stavrou conflates them to make his assertions without attempting to explain why analysis of container images

would apply to analysis of virtual machines, or why analysis of container images would apply to analysis of snapshots. The analysis of container images versus the analysis of virtual machines involves different actions and techniques such that a person of ordinary skill in the art seeking to execute security assessments for virtual machines would not think to look to techniques focusing on container image analysis. Ex.2027 [Steve Lasker, *Key Differences between VM and Container Vulnerability Scanning*] (“The fundamental difference between containers and VMs is where the scanning occurs, with the analysis and action taken. In the container world, a single scan and detection can account for hundreds to thousands of nodes being remediated.”)

170. Additionally, Dr. Stavrou asserts that Hufsmith “teaches the benefits of alert prioritization.” Ex.1002 (Stavrou Decl.) ¶206. I disagree. Hufsmith only uses the term “alert” once, in paragraph 15 in description of Figure 3A as a flowchart executed “by a plugin of an integrated development environment to annotate code specifying container images with alerts relating to potential security vulnerabilities.” Ex.1078 (Hufsmith) ¶15. Dr. Stavrou does not cite to that passage in support of his assertion. Ex.1002 (Stavrou Decl.) ¶206. Instead, Dr. Stavrou cites to paragraphs 6, 155, and 180. Ex.1002 (Stavrou Decl.) ¶206. But paragraph 6 makes no mention of alerts and does not, contrary to Dr. Stavrou’s assertion, describe “problems caused by overwhelming vulnerability reports.” Ex.1002

(Stavrou Decl.) ¶206 (citing Ex.1078 (Hufsmith) ¶6). Paragraph 6 simply describes that the “degree to which” a detected vulnerability “pose[s] a threat in a given, and potentially highly complex, execution environment of distributed application code and other resources is difficult to quantify or express in a usable manner.” Ex.1078 (Hufsmith) ¶6.

171. A person of ordinary skill in the art also would understand that paragraphs 155 and 180, cited by Dr. Stavrou, relate to developers working in an “integrated development environment” or “IDE.” Ex.1078 (Hufsmith) ¶¶153-197. As detailed above, Hufsmith’s techniques disclosed in these paragraphs are meant to assist software developers during development of a container image. In the paragraphs Dr. Stavrou cites, Hufsmith describes ways to “annotate code specifying container images” for developers actively working in an “integrated development environment” (or “IDE”), e.g., to reduce the “cognitive load of developing container images.” *See* Ex.1078 (Hufsmith), ¶¶153-197 (describing Hufsmith’s “IDE”). These actions occur before a container is instantiated from a container image, and are not similar to the assessments contemplated by Veselov or the ’031 patent. To persons of ordinary skill in the art, Hufsmith’s IDE disclosures are vastly different from the security assessment systems and methods disclosed by Veselov (or the ’031 patent). Persons of ordinary skill in the art, for

instance, would not be motivated to look to teachings in support of container development for analyzing an image of a container and reporting results as alerts.

172. For the above reasons, it is my opinion that a person of ordinary skill in the art would not have combined Hufsmith with Veselov and Price, and would not have had a reasonable expectation of success in doing so.

2. Even If Combined, Veselov, Price, and Hufsmith Do Not Disclose or Suggest Multiple Limitations in Claims 2 and 10

173. Because claims 2 and 10 depend, respectively, on independent claims 1 and 9, they are not obvious over Veselov, Price, and Hufsmith for the same reasons that claims 1 and 9 are not obvious in view of Veselov and Price. *See* Section VIII.A.

174. Dr. Stavrou relies on Hufsmith for disclosure of the additional limitations of claims 2 and 10. Ex.1002 (Stavrou Decl.) ¶¶208-213. But Hufsmith does not render obvious the additional limitations of claims 2 and 10, which recite, “wherein the reporting the detected vulnerabilities as alerts includes indicating priority levels associated with the detected vulnerabilities.”

175. Dr. Stavrou asserts that Hufsmith discloses these limitations in four ways. Ex.1002 (Stavrou Decl.) ¶¶208-213. First, Dr. Stavrou asserts that several “evaluators” (such as “CVE and CWE evaluators”) report results in “preliminary stages” including “detected vulnerabilities... along with weights associated with each vulnerability.” Ex.1002 (Stavrou Decl.) ¶209 (citing Hufsmith ¶¶120, 134.

According to Dr. Stavrou, “[t]hese weights indicate the severity of the risk associated with a given security issue and are therefore ‘priority levels’ as recited.” Ex.1002 (Stavrou Decl.) ¶209. I disagree. These weights do not indicate the severity of the risk associated with a given security issue. Hufsmith’s weights are applied to evaluate “context properties” “identified within a container score record associated with a container image.” Ex.1078 (Hufsmith), ¶135; *see also* Ex.1078 (Hufsmith), ¶¶135-143. The weights are used to calculate a weighted sum of vulnerabilities and therefore are used to calculate a type of “count of the number of potential vulnerabilities detected.” Ex.1078 (Hufsmith) ¶90. Further, the weights are not reported as alerts, nor are they “priority levels associated with the detected vulnerabilities” as claimed. In paragraphs 120 and 134, cited by Dr. Stavrou, the weights relate to metrics and values, not to a detected vulnerability itself. *See* Ex.1078 (Hufsmith) ¶112 (“the configuration evaluator 105 determines a weight for each metric or value of a metric based on whether the configuration properties implicate that the container is more, less, or neutrally susceptible to that metric or specific value associated with that metric.”).

176. Second, Dr. Stavrou asserts that Hufsmith teaches a “report sent to the user” that “indicates the priority levels represented by” metrics. Ex.1002 (Stavrou Decl.) ¶210. Dr. Stavrou seems to imply that a weight is a metric. Ex.1002 (Stavrou Decl.) ¶210 (“These metrics reported by the results engine include the

weights....”). But as I discussed above, the weight is a “weight for each metric or value” and is not itself a metric. And Dr. Stavrou does not assert that the “metric or value” constitutes the claimed “detected vulnerabilities” and therefore the “weight for each metric or value” does not constitute a priority level for detected vulnerabilities. *See* Ex.1078 (Hufsmith) ¶96 (distinguishing between the metrics and “potential vulnerabilities”).

177. Third, Dr. Stavrou cites to “annotations in a development environment”, i.e., displaying information about a vulnerability in order to show a developer particular lines of source code in a pre-deployment container image that should be addressed. Ex.1002 (Stavrou Decl.) ¶211 (citing Ex.1078 (Hufsmith) ¶¶154, 156, 180-183). This is inapplicable to Veselov, which only analyzes deployed VMs or snapshots where such source code would be unavailable for such annotating. Dr. Stavrou does not explain how or why persons of ordinary skill in the art could have or would have implemented this feature in Veselov’s systems or the proposed Veselov-Price combination. Although he asserts that this disclosure is “broadly applicable to other types of reporting,” Dr. Stavrou explicitly relies on “annotations” of the source code that persons of ordinary skill in the art would understand *not* to apply to situations, such as in the proposed Veselov-Price combination, where *source code is not available to be annotated*. Ex.1002 (Stavrou Decl.) ¶¶211-212. Even so, Dr. Stavrou does not explain how such

annotations of source code (see Figure 3B, below) are included in “reporting the detected vulnerabilities as *alerts*.” *Id.* Simply making it “available to the user even when a partial report is presented” does not constitute an alert as discussed above. *See* Section VIII.A.2.c.

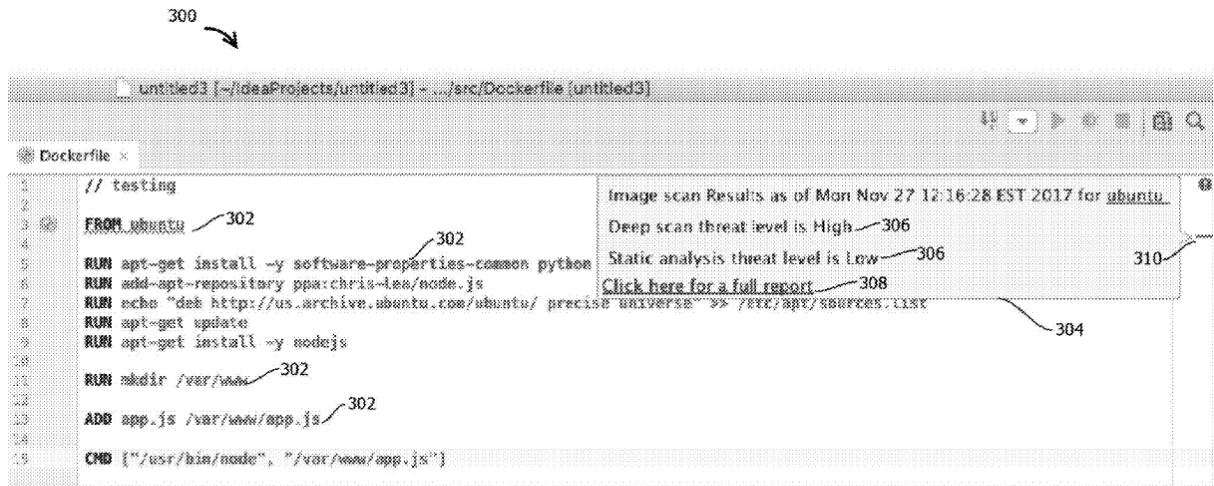


FIG. 3B

Ex.1078 (Hufsmith), Figure 3B

178. Fourth, Dr. Stavrou asserts that Hufsmith teaches that “the reporting includes indicating various types of priority levels associated with the detected vulnerabilities.” Ex.1002 (Stavrou Decl.) ¶213. Again, simply reporting results does not teach or suggest that the alleged priorities are provided as part of reporting *alerts*. Section VIII.A.2.c. Additionally, to the extent Dr. Stavrou intended to add “global weights” as the claimed priority levels, I disagree. *See* Ex.1002 (Stavrou

Decl.) ¶213. Hufsmith explains that its global weights are not associated with any particular detected vulnerability but rather apply equally to all detected vulnerabilities for given container images. *See* Ex.1078 (Hufsmith) Figs. 3a/3b, ¶138 (global weight reflects whether an updated version of a container is available), ¶139 (global weight based on trends in scores). Such weights are not the claimed “priority levels” (plural) “associated with the detected vulnerabilities” (plural).

C. Ground 3: Claims 6 and 14 Would Not Have Been Obvious Over Veselov, Price, Hufsmith, and Huseinović

179. Claims 6 and 14 depend from claims 1 and 9, respectively. Dr. Stavrou proposes to combine Veselov, Price, Hufsmith, and Huseinović to render claims 6 and 14 obvious. I disagree, however, that a person of ordinary skill in the art would have been motivated to combine these references or had a reasonable expectation of doing so, including for the reasons I provided for the combinations of Veselov and Price (Ground 1) and Veselov, Price, and Hufsmith (Ground 2). Additionally, Huseinović explicitly counsels *against* collecting offline data. Huseinović at 1 (describing the “traditional procedure” of “digital forensic investigation” with an “offline analysis” to have “drawbacks” as it is “obvious that data that was in physical memory is lost.... If the computer is set to off, the data are lost and cannot be retrieved for later analysis.”).

180. But even assuming that a person of ordinary skill in the art would combine these four references, they do not render claims 6 and 14 obvious. Ground 3 addresses only additional limitations from dependent claims 6 and 14 of the '031 patent. Ex.1002 (Stavrou Decl.), ¶215. Independent claims 1 and 9, however, on which claims 6 and 14 depend, are still challenged based on only Veselov and Price. I discussed above for Ground 1 that the combination of Veselov and Price does not render claims 1 and 9 obvious to a person of ordinary skill in the art, including because of multiple missing non-obvious claim limitations and the objective evidence of non-obviousness. Consequently, it is my opinion that, for at least the same reasons Petitioner and Dr. Stavrou have failed to show that claims 1 and 9 would have been obvious over Veselov and Price, they have failed to show that claims 6 and 14 would have been obvious over Veselov, Price, Hufsmith, and Huseinović.

IX. Objective Indicia of Non-obviousness

181. I have considered objective evidence that supports that a person of ordinary skill in the art would not have considered the claims of the '031 patent to be obvious, as discussed below. It is my opinion that, even assuming that Petitioner and Dr. Stavrou made a threshold showing of obviousness (they have not for the reasons explained above), there is objective evidence of non-obviousness that

shows that the claimed combination of features would not have been obvious to a person of ordinary skill in the art.

A. Orca's SideScanning™ Uses The '031 Patent Inventions

182. I have reviewed Section III of Exhibit 2074, which is the Declaration of Avi Shua. I understand that Mr. Shua is a co-founder of Orca and the named inventor of the '031 patent. In that section of his declaration, Mr. Shua maps Orca's SideScanning™ to the claims of the '031 patent. I have further reviewed the Exhibits cited in that Section of Exhibit 2074. Mr. Shua's mapping is consistent with my review of those Exhibits. Thus, I adopt and incorporate by reference Section III of Exhibit 2074. Based on my review of those Exhibits and Mr. Shua's mapping, it is my opinion that Orca's SideScanning™ practices the claims of the '031 patent.

B. Long-Felt, Unresolved Need

183. Between 2010-2014, there was a major shift of commercial computing services to cloud computing platforms. Ex.2076, 5-6 (describing the "shift to cloud" beginning in 2010-2014). Amazon introduced the first commercial cloud in 2006, followed by Google in 2008, and Microsoft in 2010. Ex.2134, *The History of Cloud Computing*, Cyber Magazine (Nov. 28, 2021) at 2. By 2014, cloud adoption had "take[n] off" with huge year-over-year increases in adoption. Ex.2134 at 2 ("Cloud adoption takes off" around 2012). As the '031 patent

describes, “[o]rganizations [] increasingly adapted their applications to be run from multiple cloud computing platforms.” ’031 patent, 1:20-23. Virtualization technologies such as virtual machines, software containers, and serverless functions helped enable this shift to commercial cloud computing. *See* ’031 patent, 1:36-39.

184. As applications moved to the cloud, security lagged behind. “In the early day of cloud, [security] vendors rushed to deploy their existing solutions into this new environment.” Ex.2076 at 4. But even as vendors realized that cloud computing posed new security challenges, new solutions had a number of drawbacks. *See* Ex.2136, Cloud Security Report, 2 (2018) (“As more workloads are shifting to the cloud, cybersecurity professionals remain concerned about security of data, systems, and services in the cloud.”); Ex.1029, *Overview of Attacks on Cloud Computing* (April 2012) at 1 (“[S]ecurity is the biggest problem of” cloud computing systems). As the ’031 patent describes, the cloud security industry developed a number of methods for protecting cloud computing infrastructures, including: (1) “traffic inspection”, (2) using “network scanner[s] deployed out of path,” (3) “traffic monitor[ing],” and (4) using “agents installed in each server,” but these methods suffered from multiple disadvantages. ’031 patent, 1:44-2:14.

185. Generally, these problems relate to (1) limited visibility and (2) limited coverage. The problem of limited visibility includes problems with inspecting encrypted traffic, scanning servers that either don't respond or risk exposing sensitive data, and inability to detect misconfiguration based on only observing, e.g., network traffic. '031 patent, 1:49-2:1-6. The problem of limited coverage, which was especially true with agent-based systems, includes difficulties associated with installing agents on every VM. Thus, while an agent may have better visibility into a VM's security posture, the difficulty in installing agents throughout a platform left big gaps in coverage. *See* Ex.2076, 11 ("Traditional agents struggle with complex deployment and maintenance."); '031 patent, 2:14-15 ("[I]nstalling agents in a large data center may take months."); Ex.2103, *Security Agents Don't Belong In Your Cloud* (Aug. 18, 2021) at 3 ("Customers tell me that agent-based approaches are only able to cover up to 50% of the cloud real estate even after several months of deployment.").

186. Lack of visibility and coverage was particularly a problem for VMs that are at rest or inactive. Ex.1001 ('031 patent), 2:15-26, 3:53-55. VMs at rest "[we]re not inspected for vulnerabilities, simply because scanners and/or installed monitoring agents cannot operate on a powered-off machine," or only past, potentially out-of-date information for such machines was scanned. Ex.1001 ('031 patent), 2:25-32.

187.

REDACTED

188. Thus, even by 2019, “[a]lthough cloud security has evolved over the years and is becoming more mature, the endless journey to obtain the right level of visibility and control over the cloud workloads [was] still a challenge.” Ex.2137, Diogenes, *The Quest for Visibility and Control in the Cloud*, ISSA Journal, 1 (March 2019); *see also* Ex.2136, Cloud Security Report, 2 (“Organizations continue to adopt cloud computing at a rapid pace,” but “traditional security tools

are often not suited for the challenges of dynamic, virtual and distributed cloud environments.”), 3 (“Eighty-four percent [of respondents] say traditional security solutions either don’t work at all in cloud environments or have only limited functionality.”).

189. Accordingly, there was a long-felt, unmet need for a cloud security solution that could provide both visibility and coverage to adequately address “constantly evolv[ing]” threats and “undetected vulnerabilities [that] can pose cyber threats,” including for “data or a machine at rest.” ’031 patent, 2:33-36.

190. Orca’s claimed inventions solved this need. As discussed in the following sections, Orca’s claimed approach, embodied in its SideScanning™ product, was a “genuinely genius innovation” such that cloud security posture management “can be defined by pre and post-Orca.” Ex.2076 at 7. I have also reviewed articles by Wiz and Palo Alto Networks that describe agentless scanning, and its benefits, in a manner that closely matches the claimed invention. For example, I have reviewed Exhibit 2138, which appears to be a Wiz article from September 28, 2021, titled “Agents Are Not Enough.” Ex.2138 at 1. In this article, Wiz describes: “[A]gentless deep scanning approaches [] achieve a level of comprehensiveness in terms of coverage and features that is not possible with agent-based approaches due to the tradeoff between performance and impact to the environment.” Additionally, I have reviewed

Exhibit 2135, which appears to be a Palo Alto Networks article from November 16, 2021, titled “Palo Alto Networks Shifts Left with Prisma Cloud 3.0.”

Ex.2135 at 1. In this article, Palo Alto Networks describes: “agentless scanning... works off cloud provider APIs and disk snapshots” to “extend[] coverage to both active *and dormant* virtual machines”. Ex.2135 at 3.

191. “Orca launched the first version of **agentless side scanning**, which allowed deep visibility into cloud workloads”—including for VMs at rest¹⁰—“without the need to install agents.” Ex.2076 at 11.

Orca Security made a *big leap* in cloud security with the introduction of its patented agentless SideScanning technique in 2019. This innovation introduced workload visibility without needing security teams to install an agent on endpoints or installing software on workloads for assessments. They operate out-of-band by accessing cloud workloads at the runtime storage level, gathering crucial data through the cloud provider’s APIs, while scanning disk snapshots.¹¹

Ex.2076 at 31.

192. **REDACTED**

¹⁰ See '031 patent, cls. 1, 9, 16 (limitations 1.6, 9.7, 16.7).

¹¹ See '031 patent, cls. 1, 9, 16 (limitations 1.2-1.5, 9.2-9.4, 9.6, 16.2-16.4, 16.6).

REDACTED

C. Orca's claimed inventions overcame industry skepticism to become widely accepted--and widely praised—in the industry.

193. As discussed below, I have reviewed objective evidence that there was industry skepticism of Orca's claimed inventions, but that they were quickly accepted and widely praised by the industry. This evidence further supports that skilled artisans at the time of the invention would not have viewed Orca's claims as obvious.

¹² See '031 patent, cls. 1, 9, 16 (limitations 1.7, 9.8, 16.8).

194. A 2024 report on “the Future of Cloud Security” states “[t]here is no doubt, the pivotal moment was the introduction of agentless scanning by companies like Orca and Wiz”—of course, “Orca launched the first version of agentless side scanning” and, as shown below, Wiz later copied Orca. Ex.2076 at 8, 11. It was a “turning point – but anyone who tells you they knew it at the time is lying.” *Id.* at 8.

195. After Orca “introduced the concept of agentless side scanning,” it was “difficult” for “older providers to come to terms with [its] success.” Ex.2076 at 7. That 2024 report states that these providers were “skeptics” because “from a security perspective, agentless scanning was nothing compared to what attacks they were able to detect and protect.” *Id.* “They missed however that security teams were not yet in the weeds of these architectures,” so before Orca, they “didn’t know what they didn’t know.” *Id.* 7. And in 2021, Palo Alto Networks released an article about how Orca’s solutions “fall short.” Ex.2106, *How Agentless Approaches Sideline Cloud Workload Protection Strategies*, Palo Alto Networks (Feb. 23, 2021) at 1 (“Agentless Solutions Fall Short”). By November, Palo Alto Networks announced their own agentless scanning solution. Ex.2135, *Palo Alto Networks Shifts Left with Prisma Cloud 3.0* at 2 (announcing “Agentless Security for Flexible Cloud Workload Protection” that “Provide[s] visibility into an organization’s cloud workload and application risks”).

196.

REDACTED

REDACTED

REDACTED

197. Eventually, “everyone needed [agentless scanning] in order to stay competitive.” Ex.2076, 7; Ex.2103, *Security Agents Don’t Belong In Your Cloud* (Aug. 18, 2021) (“[A] new ‘agent-less’ approach to cloud security has recently become possible due to cloud-native APIs now available from CSPs. This approach provides security in a fundamentally different way.”). As I mentioned above, Palo Alto Networks quickly reversed course on its public criticisms of the approach, and Wiz entered the market a year after Orca with the same technology—copied from Orca.

D. Extensive Industry Praise Supports Non-Obviousness of the Claims

198. I have reviewed evidence of industry praise of Orca's inventions that further shows that skilled artisans did not view (and a person of ordinary skill in the art would not have viewed) the inventions as obvious. This evidence includes numerous industry awards, including:

- CRN's "20 Coolest Tech Startups of 2020." Ex.2119, *The 20 Coolest Tech Startups of 2020*, CRN (Nov. 19, 2020) at 9-10. CRN touted "Orca's SideScanning technology" which "automatically assesses the security state of every discovered asset" to provide "complete visibility." *Id.*
- Forbes's "Top 20 Cybersecurity Startups to Watch in 2021." Ex.2120, *The Top 20 Cybersecurity Startups to Watch in 2021 Based on Crunchbase*, Forbes (Nov. 20, 2020) at 6-7. Forbes found Orca "noteworthy for its innovative approach to providing instant-on, workload-deep security" for cloud computing systems, emphasizing Orca's "patent-pending SideScanningTM technology" that "reads cloud configuration and workloads' runtime block storage out-of-band." *Id.*
- SC Media's Best Cloud Workload Protection Solution. Ex.2123, *SC Award Winners 2023 Orca Security – Best Cloud Workload Protection Solution*, SC Media (Aug. 21, 2023) ("After years of dealing with agents, then seeing how easy it

is to install and use Orca, I knew that its agentless approach was both a major innovation and a game changer”)

199.

REDACTED

REDACTED

S

REDACTED

REDACTED

REDACTED

201. I have also reviewed evidence of public praise of Orca's inventions, including:

- Exhibit 2076, which I have been informed is a 2024 report published on Substack by Software Analyst Cyber Research (founded by Francis Odum) and James Berthoty that heaped praise on Orca's inventions. Select quotes include:

Orca: Introduced the concept of **agentless side scanning**, which allowed for deep visibility into cloud environments, and workloads, without the need to install agents on each workload. CSPM can be defined by pre and post-Orca: the ability to view workload data without an agent was a **genuinely genius** innovation - utilizing the nature of cloud APIs being accessible to vendors.

Id. at 7

Orca Security made a big leap in cloud security with the introduction of its patented **agentless SideScanning** technique in 2019. This **innovation** introduced workload visibility without needing security teams to install an agent on endpoints or installing software on workloads for assessments.

Id. at 31

They operate out-of-band by accessing cloud workloads at the runtime storage level, gathering crucial data through the cloud provider's APIs, while scanning disk snapshots. Their technology evaluates block storage bytes and reconstructs file systems, scans operating systems, and applications in a read-only mode, ensuring that virtual assets remain unaffected. This **unique** method allows companies to deploy Orca across their entire cloud environment without worrying about performance impacts or operational disruptions. For security teams, this means getting visibility into engineering workloads, without needing to bother them at all along the way.

***Id.* at 31**

Orca's SideScanning technology has proven **highly effective** for vulnerability assessments and scanning with a fast and unified database for attack paths especially for non-container assets. While Orca's advantages in agentless remains a key strength, it also has shown some limitations as cloud security teams start looking for more runtime data and visibility into their workloads. Critically, agentless scanning has always had difficulties being applied to ephemeral workloads, where what's on disk is changing minute by minute.

***Id.* at 31**

- **Quick deployment and rapid-time to value:** Agentless solutions are easier to deploy since they do not require installing software on each workload. They **leverage cloud APIs to snapshot underlying instance hard disks**, copy them into the security vendors account, where they're hooked up on the vendor's side and scanned. This creates an **elegant "like magic" approach** - where you get full visibility into the workload, without any instrumentation.

***Id.* at 11-12**

- Exhibit 2117, which appears to be an article by Panorama IT, an Orca Security Partner, noting that "Orca "transcends traditional solutions by inspecting data at rest." Ex.2117 at 1-2. "This unique capability reveals

hidden vulnerabilities and critical data exposures that other tools may ignore.” *Id.*

- Exhibit 2118, which is titled “Vendor Landscape and TCO Analysis for Achieving Cloud-Wide, Workload-Deep Security and Compliance for Public Cloud Environments.” Ex.2118 at 1. This document describes “Orca uses a patent-pending technique called SideScanning™, ... which unobtrusively scans the run-time block storage layer—out of band—to extract valuable data about the environment and the assets communicating in it.” Ex.2118 at 5. The report states that a Senior Cloud Security Engineer from a company called Sisense described Orca’s solution as “unique in that it locates vulnerabilities with precision and delivers tangible, actionable results—without having to sift through all the noise.” *Id.* at 4.

202. In summary, Orca’s extensive praise for the claimed combination of features in the ’031 patent is objective evidence that persons of ordinary skill in the art would not have viewed Orca’s claims as obvious at the time of invention.

E. Evidence Shows Wiz Copied Orca’s SideScanning™

203. I have reviewed evidence that Wiz copied Orca’s SideScanning™. This evidence, which I discuss below, further demonstrates that a person of ordinary skill in the art, at the time of the invention, would not have found the ’031 patent’s claims obvious.

REDACTED

REDACTED

207. In the following section I discuss additional evidence that Wiz copied Orca's product.

1. Evidence shows Wiz had access to Orca's invention

208. I have been informed that Wiz had access to Orca's invention at least because Orca was founded before Wiz and released its SideScanning™ before Wiz. Specifically, I have been informed that Orca and Wiz were both startup companies, that Orca was formed before Wiz, and that Orca presented its SideScanning™ technology to Wiz's founder when he (Wiz's founder) was still at Microsoft in May 2019.

209. REDACTED

210. REDACTED

REDACTED

REDACTED

REDACTED

REDACTED

REDACTED

REDACTED

REDACTED

REDACTED

REDACTED

REDACTED

REDACTED

REDACTED

REDACTED

REDACTED

REDACTED

REDACTED

REDACTED

227. As the above-discussed evidence shows, Wiz accessed and copied Orca's claimed inventions.

2. Evidence shows Wiz's Workload Scanning is substantially similar to Orca's SideScanning™

228. Wiz's Cloud Security Platform uses what Wiz calls "workload scanning," which is substantially similar to Orca's SideScanning™.

229. REDACTED

230. REDACTED

REDACTED

REDACTED

REDACTED

231. I have also reviewed some of Wiz's public documents, which show a substantial similarity with Orca's SideScanning™, including:

- Exhibit 2026, which I am informed is a webpage on Wiz's website. This webpage touts the benefits of agentless scanning and states that agentless analysis is "based on two fundamental principles" including "Privileged access to customer's cloud environment *via APIs*" and "*Snapshot scanning*." Ex.2026 at 3. "By connecting to cloud APIs, they can discover new resources as they're created, without requiring manual installation of an agent process. This maximizes security coverage from day one, improving the visibility of security issues." Ex.2025 at 8.

What is agentless security?

Agentless security provides visibility into the threats in your environments, without requiring the installation of software-driven agents.

Under the agent-based model, each of your hosts must run a monitoring process that collects data from the host's environment and sends it to your security service. Agentless security removes this requirement by having the service collect data itself, using cloud provider APIs and metadata.

Agentless analysis approaches are based on two fundamental principles:

- Privileged access to customer's cloud environment via APIs
- Snapshot scanning

Privileged access assigned to the security vendor enables discovery of all resources and services used in the cloud environment. This information is then processed to determine the list of workloads to be scanned.

Ex.2026 at 3

Advantages of agentless security

Agentless security solves most of the problems associated with agents. Instead of running an agent in each of your environments, agentless services sit *outside* your resources. They collate security information by monitoring data provided by cloud APIs and infrastructure services. This model presents several compelling advantages for security teams and administrators.

Simple, automatic coverage

Agentless platforms automatically monitor the resources in your cloud provider accounts. By connecting to cloud APIs, they can discover new resources as they're created, without requiring manual installation of an agent process. This maximizes security coverage from day one, improving the visibility of security issues.

Ex.2026 at 8

- Exhibit 2115, which I have been informed is a webpage on Wiz's website, and is titled "Securing clouds, securely." Ex.2115 at 1. This page describes "agentless scanning approaches" as "cloud native and API-based." Ex.2115 at 1. This page also describes the Wiz approach to agentless scanning, which it calls "workload scanner." Ex.2115 at 3. For example, in a "Full SaaS Wiz-hosted workload scanner," the scanner has "limited permissions to run the scan – including creating, sharing, and deleting snapshots." Ex.2115 at 3. I note that the "limited permissions" here refer at least in part to a policy file I discussed in the previous section.

Full SaaS Wiz-hosted workload scanner — In this model, the Wiz workload scanner is hosted in a Wiz cloud environment, with limited permissions to run the scan — including **creating, sharing, and deleting snapshots**. During the scan, the scanner has read-only access to the scanned disk, so some environments are a better fit for the hosted model (highly regulated, sensitive data, etc). This model offers several benefits including ease of deployment, scalability, and lack of ongoing management.

Ex.2115 at 3

This is one of the reasons that agentless scanning approaches to securing the cloud — approaches that are cloud native and **API-based** — have taken over the cloud security market, replacing legacy agent-based practices. At Wiz, we see this firsthand: from the pain of customers who want certainty and context, to the relief they feel when they gain that much-needed visibility. As **the CISO of Fox said**, “Wiz came into the picture to allow us to feel secure and confident in how fast we’re moving, even as our cybersecurity challenges keep changing.”

Ex.2115 at 1

232.

REDACTED

233.

REDACTED

234.

REDACTED

REDACTED

REDACTED

REDACTED

REDACTED

REDACTED

REDACTED

REDACTED

240. In the discussion that follows, I discuss additional exhibits I have reviewed that further support the substantial similarity between Wiz and Orca's products as it relates to specific limitations of the claims of the '031 patent.^{REDACTED}

a. Wiz's Workload Scanner is substantially similar to Orca's SideScanning in relation to limitations 1.1, 9.1, 16.1: Establish[ing] An Interface Between A Client Environment And Security Components

241. Wiz's Workload Scanner is substantially similar to Orca's SideScanningTM in relation to limitations 1.1, 9.1, and 16.1 of claims 1, 9, and 16 of the '031 Patent (reproduced in full above).

242. For example, similar to Orca's SideScanningTM, Wiz's Workload Scanner is based on assigning privileges to access and scan an environment using cloud APIs. For example, in Exhibit 2115, which I discussed above, Wiz

characterizes “Agentless scanning approaches” as being “based on assigning privileges to security vendors to access and scan an environment using cloud APIs.” Ex.2115 at 1. Wiz then “outline[s]” its “approach to secure agentless scanning” by describing the “Wiz workload scanner.” Ex.2115 at 2. I therefore understand Wiz’s description of “Agentless scanning approaches” in this document to encompass its own Workload Scanner.

Agentless scanning approaches are based on assigning privileges to security vendors to access and scan an environment using cloud APIs. When selecting an agentless scanning technology, it’s important to assess the privileges assigned to the security vendor, to ensure **least privileges** and secure access to your data. In addition, it’s important to understand exactly which data is being shared with the vendor and how that data is shared across cloud environments.

Ex.2115 at 1.

243. Wiz’s Workload Scanner thus includes an onboarding process where a client environment provides Wiz’s Workload Scanner with “privileges” that allow access to an environment and to scan the environment using cloud APIs. *Id.* This establishes an interface between a client organization’s cloud computing environment implemented by one or more cloud computing platforms, such as AWS, Azure, or Google Cloud Platform (GCP), and Wiz’s Workload Scanner.

244. **REDACTED**

REDACTED

- b. Wiz's Workload Scanner is substantially similar to Orca's SideScanning in relation to limitations 1.2-1.4, 9.2-9.4, 16.2-16.4: Using The Cloud Computing Platform APIs To Identify Virtual Disks Of A Virtual Machine, and Query and Receive A Location Of The Virtual Disk(s)**

245. Wiz's Workload Scanner is substantially similar to Orca's SideScanning™ in relation to limitations 1.2-1.4, 9.2-9.4, and 16.2-16.4 of claims 1, 9, and 16 of the '031 Patent.

246. Wiz's Workload Scanner uses the interface between a client's environment and Wiz's security components to utilize cloud computing platform APIs to identify virtual disks of a virtual machine in the client environment, and to query and receive an identification of a location of the virtual disk(s).

The Wiz workload scanner

Traditional cloud security posture management tools are focused on the configuration layer only. When we started Wiz, we understood that in order to analyze an end-to-end risk, we would need to analyze *multiple* layers, including network, identity, data, and the workloads themselves. This is why we built a workload scanner that is designed for the cloud and uses cloud APIs, instead of relying on older agent-based approaches.

Ex.2115 at 2

REDACTED

REDACTED

- c. **Wiz's Workload Scanner is substantially similar to Orca's SideScanning in relation to limitations 1.5, 9.6, 16.6: Perform[ing] At Least One Of: (i) Taking At Least One Snapshot, And (ii) Requesting Taking At Least One Snapshot Of The Virtual Machine At Rest, Wherein The At Least One Snapshot**

**Represents A Copy Of The Virtual Disks Of The
Virtual Machine At A Point In Time**

249. Wiz's Workload Scanner is substantially similar to Orca's SideScanning™ in relation to limitations 1.5, 9.6, and 16.6 of claims 1, 9, and 16 of the '031 Patent (reproduced in full above). Wiz's Workload Scanner will take at least one snapshot of a virtual machine at rest, and/or request taking of at least one snapshot of a virtual machine at rest.

250. REDACTED

REDACTED

REDACTED

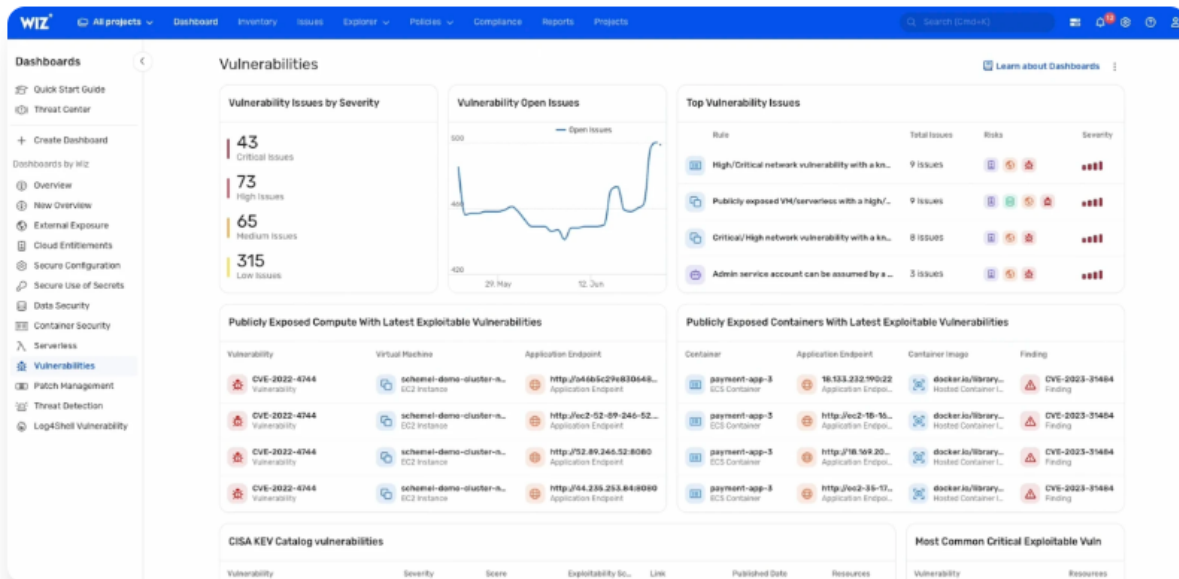
REDACTED

252. Wiz's Workload Scanner performs "Offline Workload Scanning" and therefore performs and requests snapshots of VMs at rest. Ex.2167 (Wiz CSPM Buyers Guide) at 11; *see also* Ex.2165 at 14 ("It's important to be able to scan virtual machines or containers even if the workload is offline.").

- d. Wiz's Workload Scanner is substantially similar to Orca's SideScanning in relation to limitations 1.6, 9.7, 16.7: Analyz[e/ing] The At Least One Snapshot To Detect Vulnerabilities, Wherein During The Detection Of The Vulnerabilities By Analyzing The At Least One Snapshot, The Virtual Machine Is Inactive;**

253. Wiz's Workload Scanner is substantially similar to Orca's SideScanningTM in relation to limitations 1.6, 9.7, and 16.7 of claims 1, 9, and 16 of the '031 Patent (reproduced in full above).

254. For example, Wiz's workload scanner "analyze[s] an end-to-end risk" by "analyz[ing] multiple layers, including network, identity, data, and the workloads themselves." Ex.2115 at 2. Additionally, Wiz performs "vulnerability analysis" based on the snapshot.

Stage 6: Vulnerability analysis

Example vulnerability dashboard that prioritizes issues by contextual severity

After scanning, the tool will generate a comprehensive list of identified vulnerabilities, order them based on severity, filter away false positives, and provide options for remediation.

Ex.2165 at 8

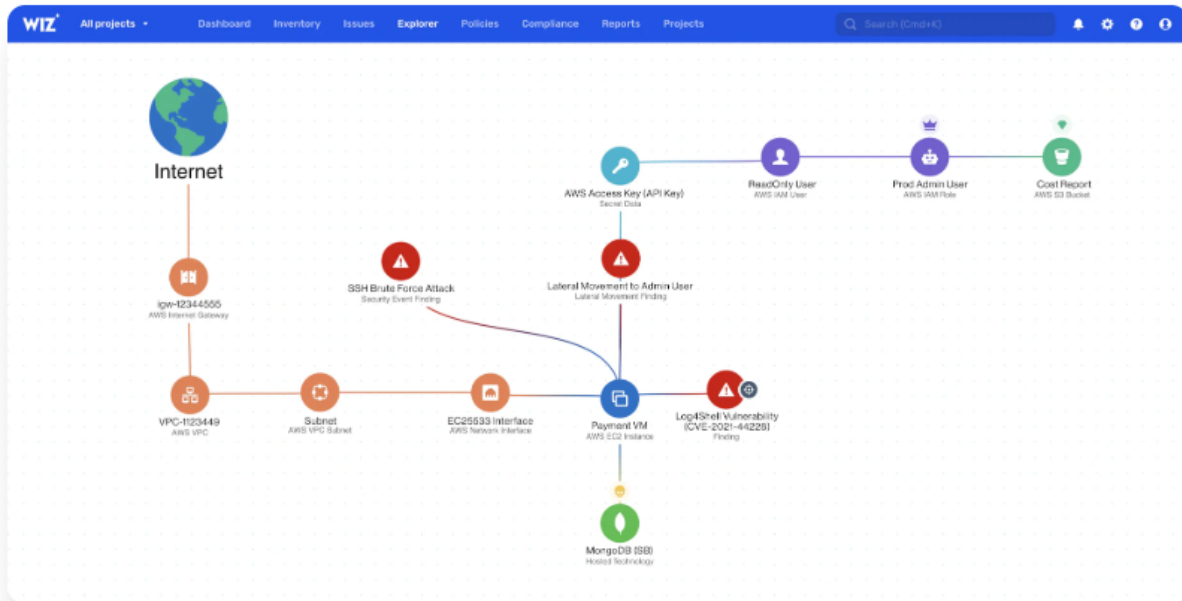
255. As discussed above, Wiz’s Workload Scanner performs “Offline Workload Scanning” and therefore performs and requests snapshots of VMs at rest. Ex.2167 (Wiz CSPM Buyers Guide) at 11.

- e. **Wiz’s Workload Scanner is substantially similar to Orca’s SideScanning in relation to limitations 1.7, 9.8, 16.8: Report[ing] Detected Vulnerabilities As Alerts**

256. Wiz’s Workload Scanner uses the features recited in limitations 1.7, 9.8, and 16.8 of claims 1, 9, and 16 of the ’031 Patent (reproduced in full above).

257. For example, Wiz's Workload Scanner performs "Risk-based prioritization" of vulnerabilities that purports to "mitigate the amount of alert fatigue you experience."

3. Risk-based prioritization



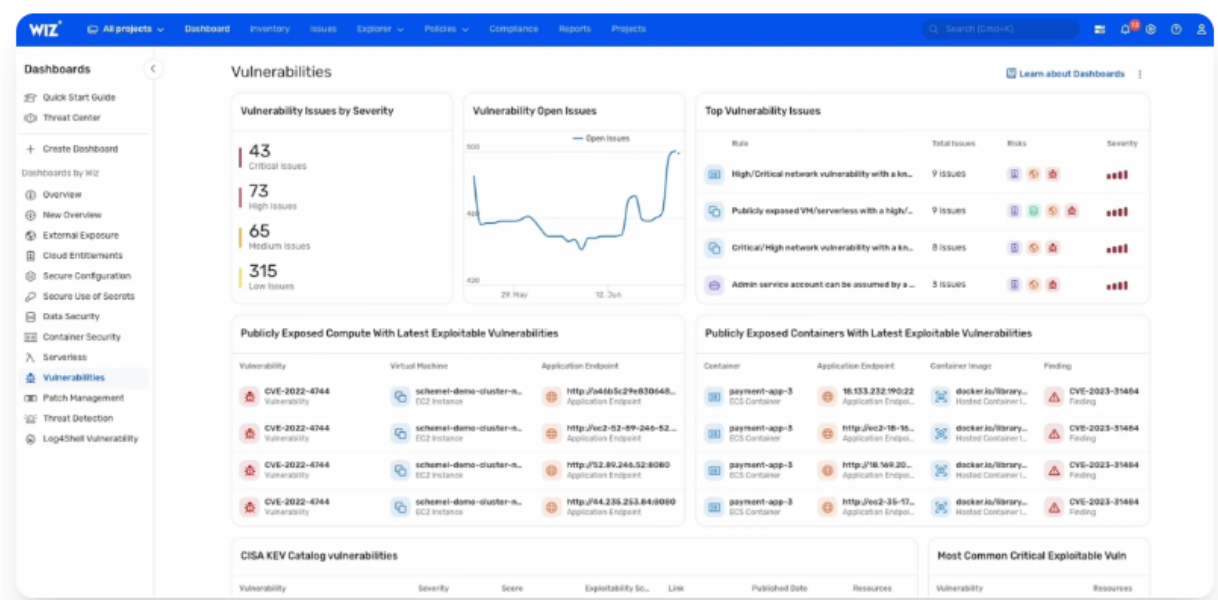
Example visualization of the risk-based context that a vulnerability scanner should provide

Choose a tool that provides risk-based prioritization of vulnerabilities, considering factors such as severity, exploitability, and asset criticality alongside elements such as external exposure, **cloud entitlements**, secrets, misconfigurations, and malware presence. A tool with this functionality can correlate your vulnerabilities that have numerous risk factors to mitigate the amount of alert fatigue you experience.

Ex.2165 at 14-15

258. Wiz's Workload Scanner reports issues as alerts.

Stage 6: Vulnerability analysis



Example vulnerability dashboard that prioritizes issues by contextual severity

After scanning, the tool will generate a comprehensive list of identified vulnerabilities, order them based on severity, filter away false positives, and provide options for remediation.

Ex.2165 at 8

f. **Wiz’s Workload Scanner is substantially similar to Orca’s SideScanning in relation to limitations 9.5, 16.5: “emulating the virtual disks of the virtual machine to generate at least one snapshot”**

259. Wiz’s Workload Scanner uses the features recited in limitations 9.5 and 16.5 of claims 9 and 16 of the ’031 Patent (reproduced in full above).

260. In some organizations’ security configurations, Wiz’s Workload Scanner emulates the virtual disks of a virtual machine. For example, Wiz describes the role of hypervisors to emulating hardware.

How VMs work

Virtual machines use virtualization to achieve resource isolation. In virtualization, host machines consist of separate VMs with separate operating systems and dedicated virtual hardware. This level of logical separation is made possible by a hypervisor, a software layer that manages the physical resources and allocates these resources to the VMs as required.

As illustrated below, hypervisors play an important role in the process of emulating underlying hardware.

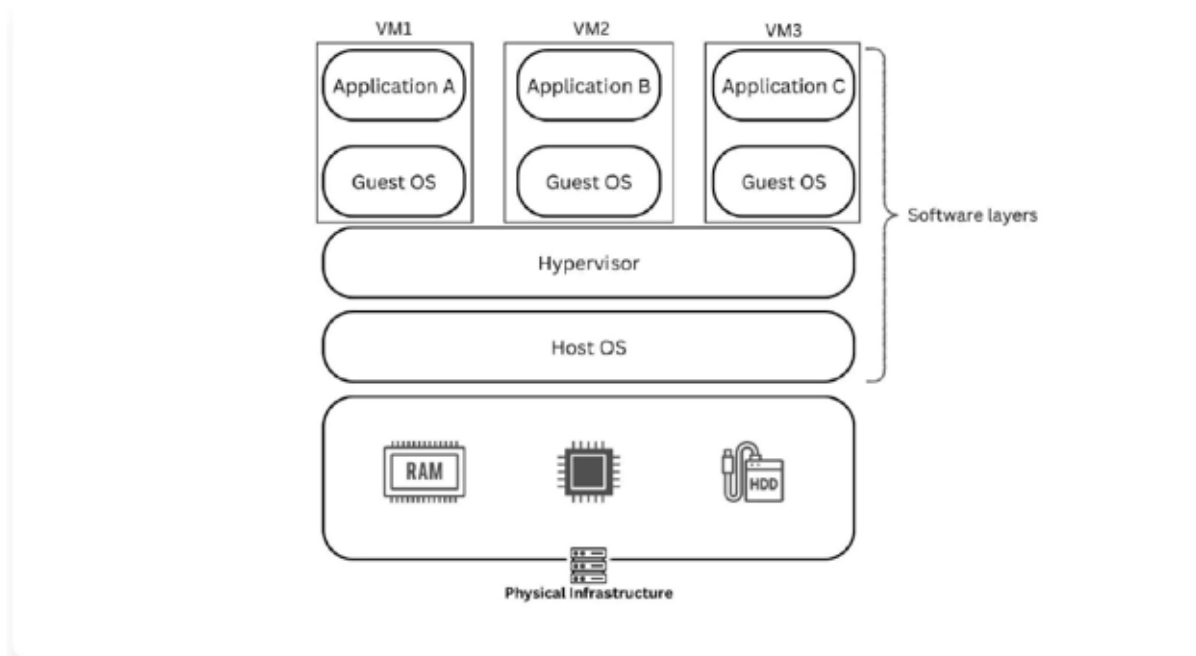


Figure 3: Virtualization

Ex.2075 at 3-4

F. Commercial Success

261. I have reviewed evidence that Orca's SideScanning™ was commercially successful, including investment funding of more than \$600 million and REDACTED Ex.2140, *Orca 2020 Recap*, 2; Ex.2112, *Upstart Orca Security Raises \$210M In Google-Backed Funding*, CRN, 1-2; Ex.2139, *Orca Security Raises \$340M To Scale Platform R&D*, *Channel*, CRN, 2; Ex.2312. I have also reviewed evidence that Orca's

innovations created a new market for agentless Cloud Security Posture

Management. *See* Ex.2076, 7 (“CSPM can be defined by pre and post-Orca”);

see also

REDACTED

262. I have also reviewed evidence that Wiz’s Cloud Security Platform, which uses what it calls “workload scanning” and which practices the claims (as shown above) has been commercially successful as well, including claims that it became the fastest-growing software company in history, and articles describing valuations of the company in the tens of billions of dollars. Ex.2116, *\$100M ARR in 18 months: Wiz becomes the fastest growing software company ever*, 2 (Aug. 10, 2022); Ex.2141, *Cloud Security Startup Wiz, Now Valued at \$10B, Raises \$300M*, 2; Ex.2142, *Wiz walks away from \$23 billion deal with Google, will pursue IPO* (July 22, 2024).

263. It is my opinion that the commercial success of both Orca’s and Wiz’s products has nexus to the claims of the ’031 patent. For example, both systems practice the claims of the ’031 patent as discussed above. REDACTED

REDACTED

X. CONCLUSION

All statements made herein of my own knowledge are true, and all statements made on information and belief are believed to be true. Further, I am aware that these statements are made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under 18 U.S.C. § 1001. I declare under penalty of perjury that the foregoing is true and correct.

Executed on 4/6/2025, in Boston, Massachusetts.



Dr. David Kaeli