

Petitioner's Demonstratives

Wiz, Inc. v. Orca Security Ltd.

IPR2024-00863, IPR2024-00864, IPR2024-00865

U.S. Patent Nos. 11,663,031; 11,663,032; & 11,693,685

Grounds

Ground	Claims	35 U.S.C. §	Prior Art
'031 Patent			
1	1, 3-9, 11-16	103	Veselov & Price
2	2, 10	103	Veselov, Price, and Hufsmith
3	6, 14	103	Veselov, Price, Hufsmith, and Huseinović
'032 Patent			
1	1-11, 13-25	103	Veselov and Hufsmith
2	12	103	Veselov, Hufsmith, and Hutchins
'685 Patent			
1	1-22	103	Veselov and Hufsmith
2	1-22	103	Veselov, Hufsmith, and Chari
3	2-3, 14-15	103	Veselov, Hufsmith, Chari, Price, and Huseinović

'031 Patent

Topic	Issues
Claim Construction	• “location” — not disputed • “analyzing” snapshot • “taking at least one snapshot”
Ground 1	• Veselov’s common disclosures • Veselov-Price Combination • Element 1.2 • Elements 1.3-1.4 • Elements 1.5-1.6 • Element 1.7
Ground 2	• Veselov-Price-Hufsmith Combination
Ground 3	• Veselov-Price-Hufsmith-Huseinović Combination

Claim Construction – “location”

Pet. 9-10

A POSA would have understood that the ordinary and customary meaning of a “location” in this context broadly encompassed a virtual location and a non-virtual location. EX1002, ¶¶78-80; *see also id.*, ¶¶30 (data locations), 45-46 (virtual disk

Claim Construction – “analyzing the at least one snapshot”

Pet. 10-11

Accordingly, the discussion below applies a construction of “analyzing the snapshot” encompassing both direct analysis of the snapshot data and analysis of a VM instantiated from the snapshot. EX1002, ¶84. Veselov describes both of these

Claim Construction – “analyzing the at least one snapshot”

POR, 17

The Figures 5A/5B embodiments have yet other problems for Wiz’s theories.

In these embodiments, Veselov discusses analyzing snapshots as “data files.”

Dr. Kaeli

Q. So it's your understanding that the embodiment starting at line 62 describing instantiating a copy of the VM is one of the various techniques that can be used to analyze the views and snapshots that is described in the paragraph starting at line 30.

Did I get that right?

A. The specification is describing various embodiments for techniques that can be used to analyze the views and snapshots.

EX1401, 36:2-11

'031 Patent

Various techniques can be utilized to analyze the views and snapshots, depending on the type of vulnerability and cyber threats to be detected. Following are some example embodiments for techniques that may be implemented by the security system 140.

6:30-34

In yet another embodiment, the security system 140 may instantiate a copy of the VM 119-2 and/or a subset of applications of the VM 119-2 on the server 115 or a separate server and monitor all activity performed by the instance of the VM. The execution of the instance of the VM is an

6:62-66

PO's District Court Brief

or data about the snapshot). But a snapshot can be analyzed indirectly, and nothing in the intrinsic

evidence supports reading such indirect analysis out of the claims as Wiz asks the Court to do. *See*,

snapshots even if the VMs are instantiated from snapshots. Further, on its face, the language Wiz

selectively quotes—“limitations reciting ‘analyzing the at least one snapshot’ *encompass*

analyzing the snapshot itself’—is not limiting. *See Thorner v. Sony Comput. Ent. Am. LLC*, 669

EX1117, 64, 67-68

Claim Construction – “taking at least one snapshot”

[1.5] 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, wherein the at least one snapshot represents a copy of the virtual disks of the virtual machine at a point in time

Prosecution History

generate perform at least one of taking a snapshot of the virtual machine at rest, or requesting the snapshot of the virtual disks of the virtual machine at rest, wherein the snapshot represents a copy of the virtual disks of the virtual machine at a point in time;

EX1004, 123

United States v. Teletronics, Inc., 857 F.2d 778, 782–83 (Fed. Cir. 1988)

to remove this recitation. The arguments emphasizing the use of a skin electrode, which were made at the time the application claims explicitly contained such a limitation, cannot furnish a basis for restricting issued claim 1, which lacks any such limitation. *See Smith v. Snow*, 294 U.S. 1, 16, 55 S.Ct. 279, 285, 79 L.Ed. 721 (1935) (“It is of no moment that in the course of the proceedings in the Patent Office the rejection of narrow claims was followed by the allowance of the broader Claim 1.”); *Kistler Instrumente AG v. United States*, 628 F.2d 1303, 1308, 224 Ct.Cl. 370, 211 USPQ 920 (1980) (*aff’g* and adopting 203 USPQ 511, 516) (courts are not permitted to read “back into the claims limitations which were originally there and were removed during prosecution of the application through the Patent Office.”)

Reply, 1

PO contends the “at rest” requirement of option (ii) applies to option (i).

While the petition also proves obviousness under PO’s flawed interpretation, the intrinsic record refutes PO.

Specification/Provisional

machine at rest. That is, VM 119-2 is mostly in an inactive state (e.g., being execute a day in a month, a month in a year, and remains inactive otherwise.

may generate a query to create a new VM 119-2. The view may be taken when the VM 119-2 is about to transition into an inactive state or when the same VM 119-2 is at rest. It

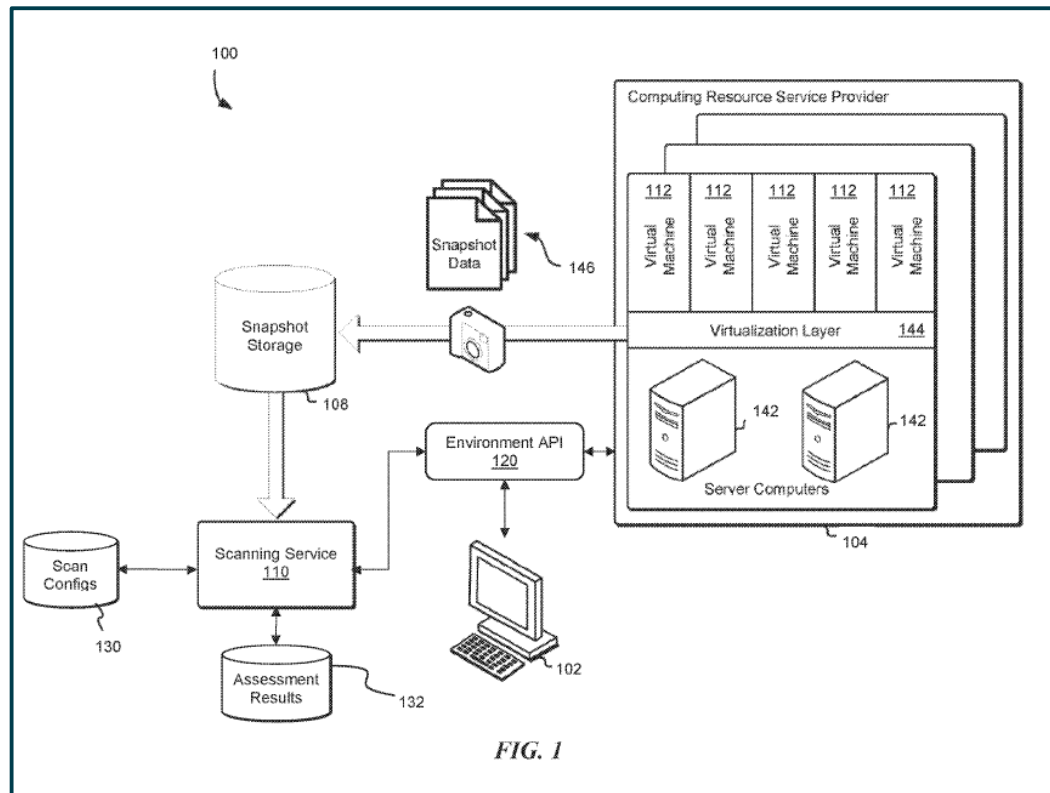
’031 Patent 4:5-19, 6: 1-9

machines. The present disclosure contemplates implementation of any suitable security assessment, including security assessments that are defined by rules packages, such as Common Vulnerabilities and Exposures (CVEs), Center for Internet Security (CIS) benchmarks, “best practices” packages, static or runtime behavior analysis, host configuration assessments, and the like.

is/are the target of the security assessment. In various embodiments, the scanning system obtains, or obtains access to, a state of the resource at a point in time (e.g., a “snapshot”) prior to, or in conjunction with, initiating the security assessment. The snapshot may include all of the data needed to recreate the state of the computing resource within a duplicate, virtual computing resource. For example, the target computing resource may be an instance of a virtual machine implemented within block-level storage device resources allocated to a logical volume. The snapshot may be a copy of the state of memory, the state of any devices (virtual or physical) allocated to the resource, block-level image of the entire logical volume; or, the snapshot may be an image of only a portion of the logical volume containing

In any case, the snapshot is 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.

2:35-41, 3:23-40, 4:1-4



computing resource service provider 104. The 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 scanning service 110 as described further below.

5:1-8

Price

Price

[0027] With the boom in virtual infrastructure, a problem has emerged in vulnerability assessment services, as virtual infrastructure, in many instances, is programmatically powered down and only in-use, in some cases, for short periods of time, which in many instances do not overlap with the scan window periods. For instance, a biotech firm that is using

tion. The security assessment module can collect, for each virtual machine in the set determined to be offline, a machine image of the virtual machine via the particular virtual machine manager, and assess security of the offline virtual machines from the collected machine images.

respective virtual machines (i.e., 315, 325). Security server 305 can then read the collected images 405, 410 to perform various security assessments on data included in the images describing attributes and conditions of the offline virtual machines 315, 325. Further, in some instances, security

¶¶27, 14, 41

Dr. Stavrou

achieved predictable results in this combination. For example, as I explain below, the type of snapshot-based analysis described in Veselov does not depend on the state of the original VM during the analysis step, and Price's offline analysis is performed on VM images using techniques analogous to those in Veselov. *See, e.g.*, EX1007 (Veselov), 3:20-67, 4:1-4, 4:49-54, 16:45-53, 18:32-53, Figs. 3A-B (analyzing assessment VM), 5A-B (directly analyzing snapshot data); EX1048, ¶¶32, 42. Indeed, a POSA would have expected many VMs to already be at rest and inactive when the assessment begins. EX1066 (Zhou), 1-2 ("A significant portion

EX1002, ¶104

'031 Ground 1 Combination – Motivation

Well-known security benefits

Dr. Stavrou

VM is inactive. For example, a POSA would have arrived at the combination in order to provide the well-known security benefit of preventing further damage that might otherwise be caused if a potentially compromised VM actively runs (e.g., a vulnerability could have been exploited, or suspected malware could have contaminated other resources). EX1081 (Budko), ¶79. That is, a vulnerability in EX1002, ¶105

EX1081

may be fixed and enable non-tampering. A detective control may require vulnerability scanning of the VMs at rest to detect vulnerabilities that were recently exposed in the operating system or applications in the VM. A corrective control may require that the VO be quarantined and remediated before being allowed to be activated in a particular environment.

¶79

EX1067

of dormant images. Further, powering on VMs that haven't been used for several months could expose the datacenter to security threats. The VMware vCenter Protect tool (formerly

399

Well-known cost/efficiency benefits

Dr. Stavrou

92 (most VMs dormant for more than 1.5 months). A POSA would have been motivated to leave the target VM in its inactive/rest state in some circumstances to avoid well-known delays and costs associated with activating the VM. EX1066 (Zhou), 1-2 (starting offline VM “will be an extra cost to the image owners”); EX1002, ¶106

EX1066

A significant portion of the VM images are mostly offline and infrequently booted. Thus, any attempt to start these VMs and install patches will be an extra cost to the image owners.

1-2

EX1067

ImageElves allows updates to be applied offline on a dormant VM instead of a running instance. This allows us to update dormant VMs without any need to bring them online. Dormant VMs often do not have compute or memory resources assigned and bringing them online for updation is not feasible. Further, dormant images often lie in backup application instances or in libraries, which are brought up only when the primary instance has failed. In such cases, updating the dormant instances may delay recovery, leading to extensive downtimes. Offline updates can also lead to faster updation in

lazy fashion. The patch installer and data are injected into the image so that the patch will be applied the next time the image is booted up. This allows the downtime needed for patching to be reduced, but there could be a significant delay during image startup. This could be of the order of tens of minutes even for images that have been dormant for one month, which can be prohibitive.

393, 399

'031 Ground 1 Combination – No Teaching Away

Dr. Stavrou

106. Furthermore, a POSA also would have expected many VMs subject to security assessments such as those taught by Veselov to already be at rest/inactive when the assessment began. *E.g.*, EX1066 (Zhou), 1-2 (“A significant portion of the VM images are mostly offline and infrequently booted.”); EX1067 (Jeswani), 390-92 (most VMs dormant for more than 1.5 months). A POSA would have been EX1002, ¶106

Kaeli does not appear to dispute this. EX1002, ¶¶104, 107. Moreover, Veselov’s straightforward, readily apparent applicability to offline assets does not interfere with benefits in circumstances when the target happens to be actively running. Its state-independent snapshot-based assessment conveys *both* benefits: it does not interfere with active VMs and it allows the scanning system to assess inactive VMs. Veselov also discusses the benefit of reducing data-center costs, which is

Price

EX1400, ¶22

[0027] With the boom in virtual infrastructure, a problem has emerged in vulnerability assessment services, as virtual infrastructure, in many instances, is programmatically powered down and only in-use, in some cases, for short periods of time, which in many instances do not overlap with the scan window periods. For instance, a biotech firm that is using virtual infrastructure to perform DNA sequencing or a graphic design firm doing intensive 3D rendering/modeling may only temporarily access and “turn on” virtual computing infrastructure during tasks and periods where peak processing capacity is desired or needed. As virtual infrastructure can be priced based on the amount of time virtual infrastructure is used, excess computing resources provided by virtual infrastructure is typically “turned off” when it is not needed, among other reasons for periodically powering-on and -off virtual infrastructure.

Dr. Kaeli

Q. So you don't have an understanding or an opinion as to roughly what percentage of virtual machines in a cloud environment are powered off at any given time; right?

MR. BAJAJ: Objection to form.

A. I don't believe I've opined on that particular point. I believe that if I was -- I could go off and do that analysis, but that would take some -- quite a bit of time and effort.

EX1401, 112:11-19

'031 Ground 1 Combination – Offline Protection Routine, Predictable

Dr. Stavrou

33. It was well known that snapshots could be taken of VMs while they were in a variety of different states (e.g., powered on, suspended, powered off, etc.).

62. Security scans were routinely performed on VMs while target resources were in various states, including while they were inactive (e.g., powered off or suspended) or actively running. EX1065 (Thakar), 2:29-33 (“Scanning the echo for

EX1002

A snapshot is reproduction of the virtual machine just as it was when you took the snapshot. The snapshot includes the state of the data on all virtual machine disks and the virtual machine power state (on, off, or suspended). You can take a snapshot when a virtual machine is powered on, powered off, or suspended.

EX1015, 164

Price

virtual machine is online. For at least the virtual machines determined to be offline, a machine image is collected for each offline virtual machine. Security of the offline virtual machines is assessed from the collected images. For virtual

the offline virtual machines. The security server 305 can thereby collect registry, file, and policy setting values from the images and files obtained over the virtual machine manager 330 API(s) and then analyze these values for vulnerabilities, missing patches, policy compliance violations, and other issues. For instance, machine images (including registry files)

Abstract, ¶42

mining vulnerabilities of the machine. Scanning the echo for vulnerabilities rather than scanning the actual machine may free up resources at the machine and may allow scanning even if the machine is currently offline, e.g. not connected to a network.

EX1065, 2:29-33

like image browsing, access control and deployment, Mirage provides offline image content introspection, and manipulation capabilities. This is enabled by indexing the

EX1073, 1

'031 Ground 1 Combination – Strawmen

Dr. Kaeli

(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.

EX2073, ¶107

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.

EX2073, ¶114

Petition, 23

Wiz

(citing EX1066, 1-2; EX1067, 390-92; EX1081, ¶¶29-31, 37, 79). The combination merely involves using these well-known techniques to achieve predictable results. EX1002, ¶104. For example, Veselov’s assessment is based on the VM’s snapshot and therefore does not depend on the target resource’s operational state, and Price’s offline analysis is performed on VM images using techniques analogous to those in Veselov. EX1007, 3:20-67, 4:1-4, 4:49-54, 16:45-53, 18:32-53, Figs. 3A-B (analyzing assessment VM), 5A-B (directly analyzing snapshot data); EX1048, ¶¶32, 42; EX1002, ¶104.

Petition, 24

¶14; EX1002, ¶105 (citing EX1067, 399; EX1081, ¶79). A POSA also would have expected many VMs subject to security assessments such as those taught by Veselov to already be at rest/inactive. EX1002, ¶106 (citing EX1066, 1-2 (“A significant portion of the VM images are mostly offline and infrequently booted.”); EX1067, 390-92 (most VMs dormant for more than 1.5 months)). A POSA therefore would have applied Price’s teachings to Veselov to provide cost and efficiency benefits, since it would allow such VMs to remain in that state when obtaining/analyzing the snapshot. EX1002, ¶106 (citing EX1066, 1-2 (starting

'031 Ground 1 Combination – Snapshots and Images

Veselov

Non-limiting examples of such properties include: properties of logical volume(s) containing the target resource; properties of physical storage devices allocated to the logical volumes; file system and file information; operating system information; system and software patches applied; configurations of network protocols and services; properties of connections to other computing resources; permission settings; software packages installed, and configurations thereof; system log properties; and the like.

9:59-67

service. Examples of generating the scannable volume described below include launching a duplicate virtual machine instance in an allocated logical volume, mounting the snapshot data in an allocated logical volume like a storage device and reading the target resource state like a data file, and incorporating data elements of the snapshot data into a partial framework of the target resource to produce the duplicate target resource.

10:9-10:16

Price

managed by the virtual machine manager 330). In addition, or alternatively, an offline registry API of the virtual machine manager can be used to read and/or collect registry files from the offline virtual machines. The security server 305 can thereby collect registry, file, and policy setting values from the images and files obtained over the virtual machine manager 330 API(s) and then analyze these values for vulnerabilities, missing patches, policy compliance violations, and other issues. For instance, machine images (including registry files)

¶42

[0042] In one particular illustrative example, security server 305 can use a virtual disk API of the virtual machine manager 330 to mount a virtual image of an offline virtual machine and read files of the offline virtual machine (e.g., virtual machines 2 (315) and 4 (325) or other virtual machines managed by the virtual machine manager 330). In addition, or

Dr. Stavrou

snapshot/volume-image analysis. For example, in AWS, the machine-image resource (referred to as an “Amazon Machine Image” or “AMI”) is essentially just a wrapper for one or more snapshots. A snapshot in AWS encompasses a single volume/disk, which may or may not be a boot volume (EX1121), whereas an AMI essentially just includes one or more snapshots (one of which is a boot volume) along with metadata such as a launch parameters. EX1131.³ In fact, an AMI can simply be created from a snapshot. EX1132.⁴ This type of close relationship between machine images and snapshots was common in cloud/virtualization platforms.

EX1400, ¶27

'031 Ground 1 Combination – Veselov's Common Disclosures

Veselov

embodiment of claimed subject matter. Thus, appearances of the phrase “in one embodiment” or “an embodiment” in various places throughout this specification is not necessarily intended to refer to the same embodiment or to any one particular embodiment described. Furthermore, it is to be

24:58-62

Referring to FIG. 1, embodiments of the present disclosure may operate within or upon a computing environment 100 in which users, e.g., developers, customers, administrators, and other “users” that may hold a “user account” with a computing resource service provider 104, may use user

4:19-23

ment. The 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. At step 202, the

8:52-62

Dr. Stavrou

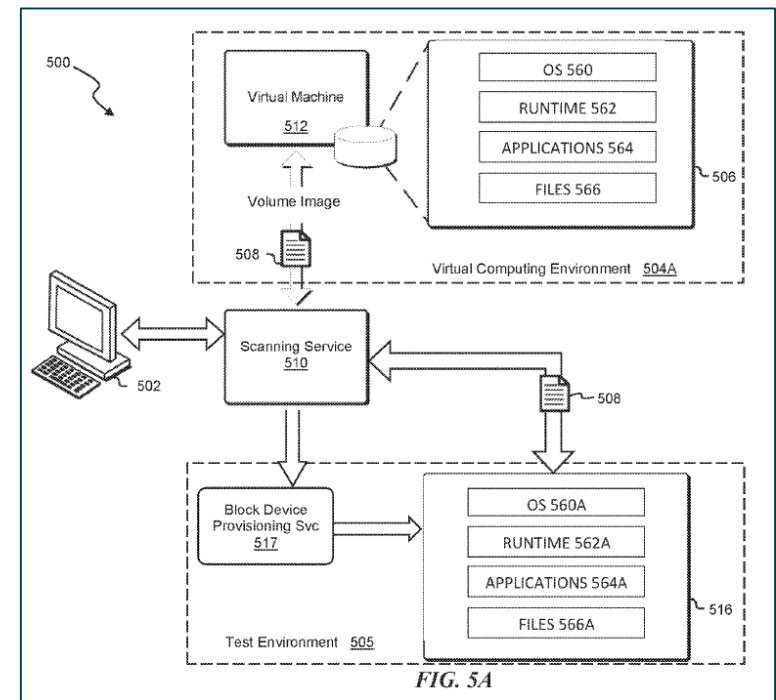
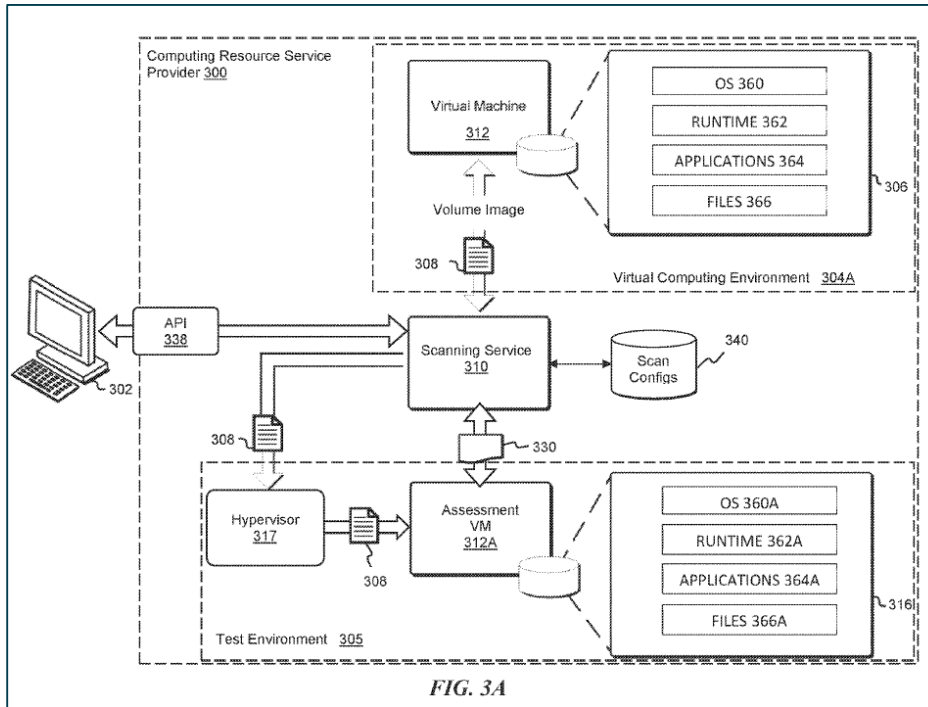
embodiments.”). In other words, Veselov makes it clear that when a portion of the disclosure associated with a particular figure/embodiment describes certain techniques or components, those teachings are broadly applicable to the rest of Veselov's disclosure. *See id.*, 24:54-25:2 (noting that phrases such as “in an understood Veselov as teaching/suggesting for each of these embodiments. In

EX1002, ¶91

other words, a POSA would have recognized that, aside from certain aspects of the analysis steps themselves, Veselov is describing features common to all embodiments and then simply not repeating itself later on to minimize unnecessary redundancy. Veselov is explicit about this, as I previously explained (EX1002,

EX1400, ¶16

'031 Ground 1 Combination – Veselov's Common Disclosures



Veselov

FIGS. 5A-B illustrate example computing environments 500, 550 providing additional embodiments of the present security assessment systems. The examples again use a virtual machine 512 as the example target computing resource of the security assessment; the user device 502, virtual computing environment 504A, logical volume 506 (with virtual machine 512 operating system 560, runtime environment 562, software applications 564, and files/file system 566), virtual machine 512, and volume image 508, as well as their implementation and interactions, are the same as described above for the associated components 302, 304A, 306, 308, 312, 360-366 of FIGS. 3A-B. In the

16:56-67

POR, 15

Initially, Veselov discusses “APIs” only in its Figure 1 and Figures 3A/3B embodiments, where computing environments are implemented by a “computing resource service provider.” Veselov, Figs. 1, 3A-3B, 4:59-5:8, 10:37-16:14;

Veselov

to the encryption key to perform the analysis. The encryption key may be maintained in a security module or security service (not shown) of the computing resource service provider. In these embodiments, the scanning service 510

18:8-11

'031 Ground 1 Combination – Veselov's Common Disclosures

assessment on the duplicate instance. In one embodiment, the scanning system may set up and launch an active virtual machine instance in the allocated computing resources, the active virtual machine instance having the same configuration as the original virtual machine instance had at the time the snapshot was captured; the system may then perform the security assessment on the active virtual machine instance in a static or runtime environment. In another embodiments, the scanning system may mount the logical volume (or transform the snapshot into a data file by identifying the file system of the original logical volume and then mount the snapshot in the duplicate logical volume) as a hard drive with data stored thereon; the system may then perform the security assessment on the data at the file system level.

In any case, the snapshot is 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. The scanning system may store the assessment results, or provide them to an owner or administrator of the virtual machine in order to evaluate the vulnerabilities of the target resources. The scanning system may then delete the snap-

3:54-4:8

'031 Ground 1 – Veselov Teaches Identifying a Virtual Disk

[1.2] using the interface, utilize cloud computing platform APIs to identify virtual disks of a virtual machine in the client environment

Veselov

resources allocated to a logical volume. The snapshot may be a copy of the state of memory, the state of any devices (virtual or physical) allocated to the resource, **block-level image of the entire logical volume**; or, the snapshot may be an image of only a portion of the logical volume containing the data required to embody an exact copy of the virtual machine instance; or the snapshot may simply be a copy of certain files of the target computing resource, such as a database of software packages installed in a virtual machine.

scan data store 130. The scanning service 110 may, in some embodiments, use the scan data to determine the parameters of the security assessment, and further to **identify the data elements of the virtual machine 112** that are needed to perform the security assessment. The scanning service 110

3:32-40, 8:25-45

installation and monitoring of the assessment system. A security assessment may involve sending requests and/or other information to one or more targets, such as a virtual machine, a group of interconnected virtual machines, a container of a virtual machine, a software application, a service, a server, a hard disk drive or logical volume, etc.,

more APIs 338, web interfaces, and the like. Various aspects of the security assessment(s) may be configured by user input, non-limiting examples of such aspects including: **selection of the assessment target(s)**; parameters of the security assessment(s), such as tests or tasks to perform,

2:46-54, 14:41-53

EX1121

Request Parameters

For information about the common parameters

Description

A description for the snapshot.

Type: String

Required: No

VolumeId

The ID of the Amazon EBS volume.

Type: String

Required: Yes

Dr. Stavrou

identify the user. EX1007, 6:25-38. Regardless, Veselov does not say that the scanning service generates a snapshot simply by giving the cloud provider a VM identifier, and a POSA would have known that this is not how snapshot generation typically worked in cloud platforms. For example, in AWS, creating a snapshot via the standard API call requires an identifier of the volume (not the VM).

EX1121 (CreateSnapshot). In any event, Dr. Kaeli's treatment of VM

EX1400, ¶41

'031 Ground 1 – Utilizing Cloud APIs to Identify the Virtual Disk

[1.2] using the interface, utilize cloud computing platform APIs to identify virtual disks of a virtual machine in the client environment

Veselov

This problem may become more acute when the target of the security assessment is a service or distributed application being provided using a distributed computing environment (e.g., a “cloud” computing environment). In such a comput-

2:64-67

computing resource service provider 104. The 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

5:1-5

Dr. Stavrou

services. *Supra*, §VIII.C. Veselov also expressly teaches, or at least suggests, using its techniques in cloud computing environments. For example, Veselov explains that its techniques address concerns that are particularly important for target resources that are “provided using a distributed computing environment (e.g., a ‘cloud’ computing environment).” EX1007 (Veselov), 2:42-67; *see also id.*, 2:67-3:19 (discussing security issues in cloud environments operated by service provider running “distributed application”). Veselov also describes its embodiments as using environments that a POSA would have recognized as typical environments providing cloud computing services. *E.g., id.*, 19:30-37 (data stores may include

EX1002, ¶126

'031 Ground 1 – Querying/Receiving a Location of the Identified Virtual Disk

[1.3] use the computing platform APIs to query a location of at least one of the identified virtual disks

[1.4] receive an identification of the location of the virtual disks of the virtual machine

Veselov

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. When executing or requesting the snapshot, the scanning service may include instructions identifying the data to include in the snapshot. For example, the scanning system may determine from the scan data that the security assessment will analyze the file system of a virtual machine instance, and may indicate to the snapshot service to include the file system details in the snapshot data. Additionally or alterna-

9:17-32

Dr. Stavrou

scanning service, to determine disk locations using such queries. These types of routine queries are described, for example, in Sancheti, which tracks virtual disk identifiers and locations “using APIs to query the virtual machine host” as a precursor to the snapshot-generation process. EX1020 (Sancheti), 21:9-22:18, Fig. 7 (depicting tracked disk identifiers and locations). A POSA would have understood Veselov’s comparable process to use these types of well-known, routinely used cloud APIs to query the location of the virtual disks that the scanning service will include in the snapshot. Accordingly, in my opinion, Veselov teaches the recited

EX1002, ¶135

Petition, 33

generation. *Supra*, §XI.B.4. A POSA would have understood that Veselov’s scanning service—which can generate the snapshot itself (EX1007, 9:9-25)—queries the location of the identified virtual disks so that it can locate them when generating a snapshot that includes those disks. EX1002, ¶134 (citing EX1020,

'031 Ground 1 – Querying/Receiving a Location of the Identified Virtual Disk

[1.3] use the computing platform APIs to query a location of at least one of the identified virtual disks

[1.4] receive an identification of the location of the virtual disks of the virtual machine

Dr. Stavrou

VM's data elements, which I discussed above. EX1007, 8:16-45. Dr. Kaeli further argues that virtual disks and VMs can have different locations. Kaeli-Dec, ¶133. With respect to their general storage location, however, that is not typically the case in a cloud environment. For example, AWS, like other major platforms, requires VMs and their associated volumes to be in the same "region" and "availability zone" (which generally meant being the same data center or group of data centers). EX1133⁵ ("The volume and instance must be in the same

EX1400, ¶41

You can attach multiple EBS volumes to a single instance. The volume and instance must be in the same Availability Zone. Depending on the

EX1133, 1

Dr. Stavrou

136. A POSA also would have understood Veselov as teaching or at least suggesting using such APIs to query the virtual disks' more general location, such as the endpoint/virtual network address of the VM to which the disks are allocated. As I discuss in more detail with respect to the "virtual address" aspect of dependent claims 4 and 12, Veselov teaches that the target VM may have a virtual network address and may communicate with other systems that are hosted by the same resource provider or elsewhere. EX1007 (Veselov), 10:48-11:4. Veselov's

Petition, 34

EX1002, ¶136

another type of "location of at least one of the identified virtual disks"). The target VM may communicate with the scanning service via the VM's virtual network address in order to perform functions such as generating a snapshot or monitoring the VM's activity/connections. See EX1007, 4:59-5:5 (scanning service may be

Veselov

300, 350. Subject to such limitations, the virtual machine 312 may use, exchange data, and otherwise interact with other physical and virtual computing resources of the computing resource service provider 300, 350, and also may communicate with devices external to the computing resource service provider 300, 350, such as a user device 302 and end user computing devices. For example, the computing resource service provider 300, 350 may implement a virtual network including public and/or private subnets, and the virtual machine 312 may be associated with a virtual network address; the computing resource service provider

10:57-67

REDACTED

'031 Ground 1 – Veselov Teaches Reporting the Vulnerabilities as Alerts

[1.7] report the detected vulnerabilities as alerts

Veselov

in a results data store; providing the assessment results in a user-readable format to a user interface to be displayed on the user device; if vulnerabilities are identified in the assessment results, comparing the assessment results to a remediation framework to identify one or more actions the user can take to address the vulnerabilities, and providing the identified actions to the user; and the like.

10:17-36

image; perform a security assessment on the second virtual machine instance to produce a first assessment result describing a security vulnerability of the first virtual machine instance; and, associate the first assessment result with the first virtual machine instance. The computer-ex-

20:57-21:13

Dr. Stavrou

(alerts “notify stakeholders of situations that require attention”). Dr. Kaeli’s assertion that Veselov does not teach reporting alerts also fails to consider the perspective of the POSA in reading these descriptions in Veselov. As I previously explained, reporting vulnerabilities as alerts was commonplace in security platforms (EX1002, ¶¶64-68), and a POSA would have recognized that Veselov’s descriptions that I noted above are consistent with this well-known, commonplace technique.

EX1400, ¶50

Dr. Kaeli

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

EX2073

'031 Ground 2 – Claims 2/10 Obvious over Veselov, Price, and Hufsmith

Claim 2: ...wherein reporting the detected vulnerabilities as alerts includes indicating priority levels associated with the detected vulnerabilities.

Hufsmith

[0006] Moreover, even in instances where potential vulnerabilities are detected, a degree to which they pose 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. As a result, resources are often used without despite vulnerabilities (even if an administrator or developer is capable of performing a thorough analysis) due to the computational and cognitive load associated with appropriately processing surfaced vulnerabilities.

various criteria. In some cases, security vulnerabilities may be classified as high, medium, or low; scored on a scale of 1 to 10; assigned some other ordinal or cardinal classification based on attributes of vulnerabilities; labeled in a taxonomy or ontology of security vulnerabilities; or otherwise associated with classifications that make the security vulnerability faster for a developer to assess than if only provided its identifier.

¶¶6,180

Dr. Stavrou

assessments. For instance, Veselov provides a high-level description of detecting security problems and reporting alerts, while Hufsmith elaborates on the detection and reporting of vulnerabilities using similar techniques (e.g., configuration assessments, static and dynamic analysis, and the use of repositories defining known security problems), that includes further prioritizing the vulnerabilities and their alerts. EX1007 (Veselov), 2:35-41, 10:30-36, 18:53-56; EX1078 (Hufsmith), ¶¶8,

205. Second, a POSA would have been motivated to combine the teachings of Veselov and Hufsmith to obtain known benefits of prioritized alerts. For example, it was well known that prioritizing alerts allowed for rapid identification of which vulnerabilities are the most dangerous to a virtual asset. EX1078 (Hufsmith), ¶138

EX1002, ¶¶204-05

'031 Ground 2 – PO's Arguments Fail

Hufsmith

[0149] The present techniques, under the above-heading and elsewhere are, in many cases, described with reference to containers, but it should be emphasized that the present techniques are applicable to other forms of encapsulated functionality, including virtual machine images, machine images, unikernels, and AWS Lambda™ functions.

[0154] The following techniques may be uses in conjunction with the approaches above or independently, which is not to suggest that any other description is limiting.

vulnerabilities. This approach leverages IDE abilities for issues specific to containers, images, and files, in some embodiments (though similar approaches are contemplated for virtual machines, orchestration configuration files, serverless configuration files, and the like). The information

¶159

images or build upon existing container images. In turn, the combined threat scores may be considered during application or container development and deployment. For

¶108

Dr. Stavrou

28, 84, 104, 113-22; *supra*, §§XI.A-C (summary of references). A POSA would have understood that reporting prioritized alerts was broadly applicable to security assessments that identify multiple vulnerabilities in target computer systems, including the type of assessments taught by Veselov. §VIII.D.3 (discussing prioritized alerts).

EX1078 (Hufsmith), ¶155. These benefits noted by Hufsmith would apply equally to security assessments such as those taught by the combination of Veselov and Price, since those assessments also address multiple vulnerabilities and similarly complex environments.

EX1002, ¶¶204, 206

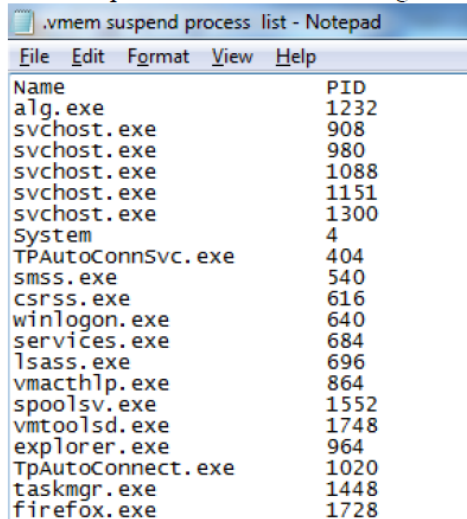
'031 Ground 3 – Claims 6/14 Obvious over Veselov, Price, Hufsmith, Huseinović

Claim 6: ...wherein the snapshot includes a page file of memory associated with the virtual machine, the page file being configured to allow deduction of one or more applications running on the virtual machine.

Huseinović

is to take a virtual machine snapshot. File vmem is virtual machine paging file. The snapshot consists of multiple

Processing the first .vmem files in Volatility framework gave the list of the processes shown on the Fig 2.



Name	PID
alg.exe	1232
svchost.exe	908
svchost.exe	980
svchost.exe	1088
svchost.exe	1151
svchost.exe	1300
System	4
TPAutoConnSvc.exe	404
smss.exe	540
csrss.exe	616
winlogon.exe	640
services.exe	684
lsass.exe	696
vmacthlp.exe	864
spoolsv.exe	1552
vmtoolsd.exe	1748
explorer.exe	964
TpAutoConnect.exe	1020
taskmgr.exe	1448
firefox.exe	1728

Fig. 2. List of the processes running on the VMware virtual machine obtained from .vmem suspend state file

The same list was obtained from the second .vmem file created by taking a snapshot.

940-41

Dr. Stavrou

Huseinović, was well-known and predictable. For example, Huseinović shows that generating a snapshot containing such a page file can be done using “standard and built-in options” of common virtualization platforms such as VMware®. EX1049 (Huseinović), 940. Deducing the running processes/applications from the page was similarly known and predictable. For example, Huseinović shows that this was

EX1002, ¶217

223. Second, with respect to clause (ii) of elements 1.5/9.6/16.6, the references teach or at least suggest that the snapshot includes the same type of page file discussed above if the VM is at rest when Veselov’s scanning service requests the snapshot. As I previously explained, many VMs would already be powered off

225. Thus, while a POSA would have recognized that starting up a VM before generating a snapshot might cause certain delays/costs, a POSA would have appreciated the tradeoffs between those concerns and the benefits of being able to analyze the VM’s memory state, and a POSA therefore would have reasonably chosen to bring an already powered off VM into a running or suspended state in some situations. For example, if the user wanted the snapshot to include information reflecting the VM’s activity and network connections as disclosed in Veselov.

EX1002

'032 Patent

Topic	Issues
Claim Construction	• “location” — not disputed • “analyzing the snapshot” — previously discussed
Ground 1	• Veselov-Hufsmith Combination • Element 1.5 • Elements 1.7 • Claims 2-5 • Claims 16/20
Ground 2	• Veselov-Hufsmith-Hutchins combination

Hufsmith

Hufsmith

ability scanners. The scanner properties determined by each vulnerability scanner are adjusted responsive to properties of the context and normalized to determine component threat scores for the container image. Then the component threat

the file. The information about the file may include image-specific information determined about the file such as a file name, file type, file location, file size, hash of the file, hash of a malicious portion of the file, one or more virus/malware signatures, whether the file or code is executed, etc. deter-

Execution environment records may further indicate which portions of a container image are dormant and which are active in a given use case. For instance, a container image may include a subroutine that registered with a networks socket, but a composition file or other code may indicate that the subroutine is never invoked. Or an container image may include a library with module having a sys.exec command based on a passed value, but a call graph of a the larger distributed application may indicate that the sys.exec command module is never called. Environmental context may be hand-coded by a developer or inferred from the composition file or other configuration data of the distributed application. Vulnerabilities associated with dormant code may be down-weighted in some cases.

Abstract, ¶¶121, 140

Hufsmith

[0182] In some embodiments, the annotation includes information pertaining to several security vulnerabilities, examples including classifications, rankings, scores, or other metrics based on attributes of the vulnerabilities, such as classifying the line of code as unsecure based upon a number of security vulnerabilities having risk scores above some value exceeding an aggregate threshold or based on presence of a particular type of vulnerability. In some embodiments, the annotation includes discrete entries for each of the security vulnerabilities, like a listing with any permutation of the above-described types of information relevant to security vulnerabilities. In some embodiments, the annota-

Dr. Stavrou

103. First, Hufsmith's matching and prioritization techniques were a natural fit and directly applicable to Veselov's teachings regarding snapshot-based security assessments. Veselov provides a high-level description of CVE analysis and alert reporting, while Hufsmith presents a more detailed discussion of common techniques for detecting and prioritizing vulnerabilities/alerts, including detecting vulnerabilities by comparing installed applications to repositories listing known vulnerable applications and usage-based prioritization of such vulnerabilities.

EX1002, ¶103

'032 Ground 1 Combination– Motivation

Dr. Stavrou

Hufsmith

various criteria. In some cases, security vulnerabilities may be classified as high, medium, or low; scored on a scale of 1 to 10; assigned some other ordinal or cardinal classification based on attributes of vulnerabilities; labeled in a taxonomy or ontology of security vulnerabilities; or otherwise associated with classifications that make the security vulnerability faster for a developer to assess than if only provided its identifier.

¶180

EX1063 (Hibbert)

another browser like FireFox or Chrome? Yes, the system is technically vulnerable but the offending application is not used and therefore a lower risk even if the user logs in as an administrator. Finally to understand the true meaning of this risk, this vulnerability has been observed in the wild exploiting targets. Users running as administrators are highly susceptible to drive by attacks versus the standard user. A traditional vulnerability report does not know the difference.

¶80

[0113] Various embodiments allow organizations currently performing process, application and transaction monitoring to centrally integrate with a virtual vulnerability scanning system (i.e., vulnerability scanning without network scanning as described in the prior art) to provide useful information. Types of information include, but are not limited to:

[0114] Indication of which applications are being run within the environment that have known vulnerabilities that can put the organization at risk;

[0115] Identification of the most frequently run vulnerable applications to help prioritize remediation of these applications through path deployment or other means; and

[0116] Identification of when critical servers or sensitive accounts are utilizing vulnerable applications to help prioritize remediation of these applications through path deployment or other means.

usage-based prioritization. It was well known that prioritizing alerts was beneficial for security assessments in several ways, including rapid identification of which assets are most at risk and which threats and vulnerabilities are the most dangerous.

EX1078 (Hufsmith), ¶138; EX1045 (Canzanese), ¶7; EX1063 (Hibbert), ¶¶187-89, 207-11, 215-16; *supra*, §VIII.D.2 (known benefits of prioritized alerts). Hufsmith

EX1002, ¶104

vulnerabilities....”). The benefits of use-based prioritization, as taught by Hufsmith, were similarly well known and readily achieved in combination with the type of security assessment taught by Veselov. *Id.*, ¶¶121, 140; *infra*, §XII.B.8 (discussing Hufsmith’s teachings of use-based prioritization); EX1063 (Hibbert), ¶¶80 (“[T]he

EX1002, ¶104

'032 Ground 1 Combination – The References Use Similar Data and Techniques

Veselov

Non-limiting examples of such properties include: properties of logical volume(s) containing the target resource; properties of physical storage devices allocated to the logical volumes; file system and file information; operating system information; system and software patches applied; configurations of network protocols and services; properties of connections to other computing resources; permission settings; software packages installed, and configurations thereof; system log properties; and the like.

9:59-67

Veselov

machines. The present disclosure contemplates implementation of any suitable security assessment, including security assessments that are defined by rules packages, such as Common Vulnerabilities and Exposures (CVEs), Center for Internet Security (CIS) benchmarks, “best practices” packages, static or runtime behavior analysis, host configuration assessments, and the like.

2:35-41

Hufsmith

files and packages (and other resources) in each layer. Then based on the file/package information, some embodiments submit those files/packages and other resources to various vulnerability scanning engines (including Veracode™ and others enumerated below) for vulnerability assessment.

a result engine 54. In some embodiments, these components may cooperate to arbitrate which layers and which portions of layers are scanned by which scanner application 16 among a heterogeneous set of scanner applications configured to apply different types of scans to different types of bodies of code and other computing resources (e.g., configuration files, images, audio files, video files, and other non-executed content).

[0105] As an example, configuration properties 106 may correspond to a set of configuration properties determined by a configuration scanner, CVE properties 111 may correspond to a set of identified CVE properties determined by a static analysis scanner or a dynamic analysis scanner, malware properties may correspond to a set of malware properties determined by a malware analysis scanner or an antivirus scanner, and CWE properties 121 may correspond to a set of CWE properties determined by a static analysis scanner or dynamic analysis scanner. The above should not be taken to suggest that other scanner properties or logical groupings of properties determined by scanning applications cannot be stored in the container score database 102. For example, one or more scanner applications may determine different or additional configuration, CVE, malware, or CWE properties which may be logically grouped and stored in ways other than those illustrated and described herein.

¶¶31, 68, 105

'032 Ground 1 Combination – Broad Applicability of Teachings in Container/IDE Examples

Hufsmith

[0149] The present techniques, under the above-heading and elsewhere are, in many cases, described with reference to containers, but it should be emphasized that the present techniques are applicable to other forms of encapsulated functionality, including virtual machine images, machine images, unikernels, and AWS Lambda™ functions.

[0153] Independent Development Environment Configured to Annotate Source Code of Container Images with Notifications of Security Vulnerabilities

[0154] The following techniques may be used in conjunction with the approaches above or independently, which is not to suggest that any other description is limiting.

vulnerabilities. This approach leverages IDE abilities for issues specific to containers, images, and files, in some embodiments (though similar approaches are contemplated for virtual machines, orchestration configuration files, serverless configuration files, and the like). The information

¶¶149, 153-54, 159

REDACTED

'032 Ground 1 Combination – The Combination Teaches the Recited Use Determination

[1.5] determining whether the matching installed applications are used by the protected virtual cloud asset

Hufsmith

the file. The information about the file may include image-specific information determined about the file such as a file name, file type, file location, file size, hash of the file, hash of a malicious portion of the file, one or more virus/malware signatures, whether the file or code is executed, etc. deter-

mand module is never called. Environmental context may be hand-coded by a developer or inferred from the composition file or other configuration data of the distributed application. Vulnerabilities associated with dormant code may be down-weighted in some cases.

¶¶121, 140

[0003] Distributed applications are computer applications implemented across multiple network hosts. The group of

selves be composed of other services. Examples of a service include an application component (e.g., one or more executing bodies of code) that communicates via a network (or loopback network address) with another application component, often by monitoring network socket of a port at a network address of the computer upon which the service executes.

For example, a CVE vulnerability may be present in a first version of an application package in a container image and a higher layer may modify that lower layer to correspond to a subsequent version in which the potential vulnerability is removed.

¶¶3-4, 115

images or build upon existing container images. In turn, the combined threat scores may be considered during application or container development and deployment. For

¶108

Dr. Stavrou

by Veselov. Indeed, Veselov already teaches the same general techniques for identifying vulnerabilities (e.g., CVEs and other rule packages) used in Hufsmith, making use-based prioritizing of such vulnerabilities equally applicable and beneficial in the type of security assessments taught by Veselov. EX1007 (Veselov), 2:35-41; EX1078 (Hufsmith), ¶¶8, 28, 84, 104, 113-22, 140; *supra*, §VIII.D.2 (prioritizing threats).

EX1002, ¶103

As I previously explained, however, a POSA would have understood that Hufsmith's container image for a "distributed application" encompasses a high-level service with multiple installed applications. EX1002, ¶¶92, 152; EX1078 (Hufsmith), ¶¶3-8. For example, a container operating as a web server (*see id.*, ¶¶3, 40) would typically include a web server application (e.g., Apache) and other supporting applications like a backend service, a web-application firewall, a process manager, etc. This is confirmed by Hufsmith's expansive assessment for things like CVEs and malware, which were typically related to installed applications. *Id.*, ¶¶33-34 (evaluating "packages, binaries, and other resources"),

EX1400, ¶26

'032 Ground 1 Combination – The Combination Teaches Prioritized Alerts

[1.7] reporting the determined potential cyber vulnerabilities, as prioritized alerts according to the use determinations

Petition

user interface, like a dashboard a report.” *Id.*, ¶96. Such reports can be “comprehensive” and convey an overall score for the resource and information about individual vulnerabilities. *Id.*, ¶¶31-33 (“comprehensive” report of image’s

42

Hufsmith

Additionally, the result engine may remove commonalities, provide scoring information and mask out at least some (e.g., all) previously identified false positives. Further, in

[0153] Independent Development Environment Configured to Annotate Source Code of Container Images with Notifications of Security Vulnerabilities

[0154] The following techniques may be uses in conjunction with the approaches above or independently, which is not to suggest that any other description is limiting.

various criteria. In some cases, security vulnerabilities may be classified as high, medium, or low; scored on a scale of 1 to 10; assigned some other ordinal or cardinal classification based on attributes of vulnerabilities; labeled in a taxonomy or ontology of security vulnerabilities; or otherwise associated with classifications that make the security vulnerability faster for a developer to assess than if only provided its identifier.

¶¶36, 153-54, 180

Patent Owner

Kaeli¶¶111. A POSA also would not have a reasonable expectation of success in applying source code annotations to Veselov, as the targets of Veselov’s security assessments that involve analyzing snapshots as data files are expressly designed to be “read-only.” Veselov, 18:1-3 (“read only to avoid inadvertently modifying the

POR, 24

In re Mouttet, 686 F.3d 1322, 1332 (Fed. Cir. 2012)

crossbar arrays. It is well-established that a determination of obviousness based on teachings from multiple references does not require an actual, physical substitution of elements. *In re Etter*, 756 F.2d 852, 859 (Fed.Cir.1985) (en banc)

'032 Ground 1 – Claims 2-5

... wherein determining whether the matching installed applications are used by the protected virtual cloud asset includes...

- [claim 3] checking configuration files of the matching installed applications...
- [claim 4] verifying access times to files by an operating system...
- [claim 5] analyzing application logs or system logs...

Hufsmith

Execution environment records may further indicate which portions of a container image are dormant and which are active in a given use case. For instance, a container image

...

mand module is never called. Environmental context may be hand-coded by a developer or inferred from the composition file or other configuration data of the distributed application. Vulnerabilities associated with dormant code may be down-weighted in some cases.

¶140

Veselov

Non-limiting examples of such properties include: properties of logical volume(s) containing the target resource; properties of physical storage devices allocated to the logical volumes; file system and file information; operating system information; system and software patches applied; configurations of network protocols and services; properties of connections to other computing resources; permission settings; software packages installed, and configurations thereof; system log properties; and the like.

relatively minimal. In one example of reproducing a virtual machine for a software application-level CVEs assessment, the snapshot may only need to contain a template, such as a virtual machine image, from which the virtual machine is instantiated, and a copy of the virtual machine package manager database, which lists all software packages installed on the virtual machine and their configurations.

9:59-57, 7:16-22

Dr. Stavrou

or active. EX1078 (Hufsmith), ¶140. Although Hufsmith does not expressly state that these records include access times to files by an operating system, such access times were a common part of such records, and application-use determinations such as those described in Hufsmith were routinely and predictably performed by verifying such access times. For example, well-known systems routinely tracked

EX1002, ¶175

Most Recently Used (MRU) Registry Keys – As previously mentioned, the Windows Registry is a series of massive databases that track system configuration and user activity. There are several registry keys that track most recently used items. An analysis of these registry keys can help an analyst quickly identify files accessed. Every application developer has the option of creating registry keys specific to his

Applications Specific Most Recently Used (MRU) — With every Windows application, developers have the ability to create their own set of registry keys to track specific configuration and user activity for their application. If a specific application is used to commit or facilitate a crime or is

EX1079, 33-34

executables that generate different records, such as logs. The records may contain information regarding an instance of an application, including configuration, process calls, exception handling, execution time, calling processes, names of files needed for execution, file types, file versions, application types, application versions, and/or the like. Records may also

that may indicate safe or vulnerable behavior. For example, information contained within the records (e.g., segments indicating file timestamp, execution time, and/or calling process) may be identified as suspicious behavior based on the behavioral rules of the vulnerability database 522. The rules may be

EX1063, ¶¶126, 184

'032 Ground 1 – Claims 16/20

[Claim 16] The method of claim 1, wherein determining the location of the snapshot of at least one virtual disk further includes querying a cloud management console of the cloud computing environment for the location of the snapshot and the location of the virtual disk

Veselov

lar virtual machine 112. The scanning service 110 may determine the corresponding virtualization layer 144 for the virtual machine 112 identified by the request and transmit a command to the virtualization layer 144 to provide snapshot data 146 (or to provide access to snapshot data 146) corresponding to the virtual machine 112. The virtualization layer

6:38-43

Veselov

104. For example, the API 120 may be a web-based interface implemented on a web server of the computing resource service provider 104 as described further below; one or more user interfaces may be transmitted to the user device 102 and displayed thereon, enabling the user of the user device 102 to provide settings, commands, software packages, and other user input, to the computing resource service provider 104.

4:35-41

Veselov

scanning service may cause the snapshot data to include metadata, such as a tag, that identifies the target resource embodied in the snapshot data. In some embodiments, the

the user device; if vulnerabilities are identified in the assessment results, comparing the assessment results to a remediation framework to identify one or more actions the user can take to address the vulnerabilities, and providing the identified actions to the user; and the like.

9:9-41, 10:32-36

Dr. Stavrou

query would also determine the virtual disk's location. For example, a POSA would have understood that the snapshot of a virtual disk and the underlying virtual disk itself would sometimes have the same general location (e.g., being located in the same computing environment). This is further supported by Figs. 3A-B and 5A-B

EX1002, ¶232

EX1133, 1

You can attach multiple EBS volumes to a single instance. The volume and instance must be in the same Availability Zone. Depending on the

more APIs 338, web interfaces, and the like. Various aspects of the security assessment(s) may be configured by user input, non-limiting examples of such aspects including: selection of the assessment target(s); parameters of the security assessment(s), such as tests or tasks to perform,

14:37-63

Dr. Stavrou

235. Querying a cloud management console to determine the virtual disk's location also would have been obvious as part of the mitigation process that I discussed earlier in relation to claim 11, since these assessments may identify vulnerabilities associated with the virtual disk, and a POSA would have expected the console discussed above to provide query-based determination of the virtual disk's location so that those vulnerabilities can be resolved. *Supra*, §XII.C.8 (claim

EX1002

'032 Ground 2 – Claim 12 Obvious over Veselov, Hufsmith, and Hutchins

[Claim 12] The method of claim 11, wherein mitigating a potential cyber vulnerability includes at least one of:
... **quarantining** the protected virtual cloud asset.

Veselov

the user device; if vulnerabilities are identified in the assessment results, comparing the assessment results to a remediation framework to identify one or more actions the user can take to address the vulnerabilities, and providing the identified actions to the user; and the like.

10:32-36

Hutchins

machine to address the detection of the malicious code. Thus, the action in response to detecting the malicious code may include placing the first virtual machine in quarantine.

Abstract

Dr. Stavrou

function, with predictable results. I discussed the well-known, predictable nature of the Veselov-Hufsmith combination in Ground 1 (*supra*, §XII.A), and the further combination with Hutchins would simply add the well-known, predictable mitigation technique of quarantining a VM without significantly impacting the rest of the security assessment. Indeed, Veselov already contemplates taking mitigating actions in general. EX1007 (Veselov), 10:24-36.

249. A POSA would have also been motivated to obtain the well-known benefits of quarantining a vulnerable/infected virtual cloud asset. It was widely recognized that quarantining a compromised asset, as taught by Hutchins, has several benefits. For example, it was well known that quarantining such an asset (1) would minimize its exposure to attacks that might exploit an identified vulnerability and (2) prevent a potentially compromised asset from contaminating other resources in the system (e.g., an infected asset might spread malware to other systems on the network). See *supra*, §VIII.D.3; EX1048 (Price), ¶14 (“The security assessment of

EX1002, ¶¶247, 249

'685 Patent

Topic	Issues
Claim Construction	• “location” — not disputed • “analyzing the snapshot” — previously discussed
Ground 1	• Veselov’s common disclosures — previously discussed • Veselov-Hufsmith combination • Elements 1.2-1.4 — previously discussed • Element 1.6 • Element 1.8 • Claims 2/14
Ground 2	• Veselov-Hufsmith-Chari combination
Ground 3	• Veselov-Hufsmith-Price-Huseinović combination

'685 Ground 1 Combination – Motivation

Dr. Stavrou

22; *supra*, §§XI.A-B. For example, Hufsmith recognizes the importance of avoiding the potential exposure of sensitive data. EX1078 (Hufsmith), ¶¶117-18 (describing

109. Furthermore, a POSA would have recognized that determining a risk level and filtering/prioritizing alerts accordingly also would have furthered this benefit, since the criticality of potentially exposed sensitive data was known to depend on factors such as the type of data and the type of VM on which it is detected.

EX1002, ¶¶108-09

Hufsmith

[0118] Impact metrics may include one or more metrics and their values such as or similar to a confidentially impact (e.g., (N) no loss of restricted information, (L) some restricted information may be obtained, or (H) total loss of restricted information); integrity impact (e.g., (N) no modi-

[0181] In some cases, the annotation may include an indication of a type of harm associated with the security vulnerability, like indicating that the security vulnerability potentially allows for execution of remotely supplied code from an attacker, indicating that the security vulnerability potentially allows for the exfiltration of confidential information, indicating that the vulnerability leaks information about an encryption key, or indicating the security vulner-

Dr. Stavrou

110. Second, a POSA would also have been motivated to combine the teachings of Veselov and Hufsmith to improve security by facilitating a more efficient, targeted review of critical security issues. It was well known that prioritizing and filtering alerts allowed for rapid identification of which vulnerabilities are the most dangerous to a virtual asset. EX1078 (Hufsmith), ¶138

EX1002, ¶110

Hufsmith

corresponds. In turn, one or more weights may be determined based on revision changes indicating that specific vulnerabilities were addressed throughout the revisions. The version history 127 may also specify whether an updated version of a container is available that address one or more vulnerabilities, in which case, the weights associated with those vulnerabilities may be modified or a global weight modified to indicate that the container image is less secure.

[0192] Thus, in some embodiments, the user's cognitive load may be managed by presenting more granular information about security vulnerabilities pertaining to commands likely to currently have the user's attention, without overloading the user with information about every security vulnerability. Though embodiments are also consistent with

¶¶138, 192

'685 Ground 1 Combination – Prioritization/Filtering Benefits Applicable to Veselov

Dr. Stavrou

prioritization of the threats, which was important for multiple reasons. First, it was understood that prioritization was important as a practical matter given the large number of threats and vulnerabilities that were commonly identified and the different concomitant risk levels. Scans of a given computer system often identified

EX1002, ¶65

Mohanty

Security Operations (also referred to as Sec Ops) teams need solutions that help them distinguish the critical vulnerabilities from the noise or false-positives. For example, a mission critical Internet Banking web server may have multiple known vulnerabilities, but which of those present genuine risk to the organization may be unknown. Various

3:9-33

Hufsmith

vulnerabilities. This approach leverages IDE abilities for issues specific to containers, images, and files, in some embodiments (though similar approaches are contemplated for virtual machines, orchestration configuration files, serverless configuration files, and the like). The information

¶159

Veselov

machines. The present disclosure contemplates implementation of any suitable security assessment, including security assessments that are defined by rules packages, such as Common Vulnerabilities and Exposures (CVEs), Center for Internet Security (CIS) benchmarks, “best practices” packages, static or runtime behavior analysis, host configuration assessments, and the like.

2:35-41

Janssen Pharms., Inc. v. Teva Pharms. USA, Inc., 97 F.4th 915, 929-30 (Fed. Cir. 2024)

Second, contrary to the court's assumptions, the '548 protocol did not need to hold itself out as flawed for a POSA to alter it.⁵

REDACTED

'685 Ground 1 – Analyzing the Snapshot to Detect Sensitive Data

[1.6] analyze the at least one snapshot to detect vulnerabilities and sensitive data, wherein analyzing the at least one snapshot requires no interaction with the virtual machine

Veselov

machines. The present disclosure contemplates implementation of any suitable security assessment, including security assessments that are defined by rules packages, such as Common Vulnerabilities and Exposures (CVEs), Center for Internet Security (CIS) benchmarks, “best practices” packages, static or runtime behavior analysis, host configuration assessments, and the like.

2:35-41

Hufsmith

[0118] Impact metrics may include one or more metrics and their values such as or similar to a confidentially impact (e.g., (N) no loss of restricted information, (L) some restricted information may be obtained, or (H) total loss of restricted information); integrity impact (e.g., (N) no modi-

[0181] In some cases, the annotation may include an indication of a type of harm associated with the security vulnerability, like indicating that the security vulnerability potentially allows for execution of remotely supplied code from an attacker, indicating that the security vulnerability potentially allows for the exfiltration of confidential information, indicating that the vulnerability leaks information about an encryption key, or indicating the security vulnerability potentially allows an attacker to direct network traffic elsewhere in a denial of service attack. In some embodi-

¶¶118, 181

Dr. Stavrou

noting exposure of “restricted information”). In view of these teachings, a POSA would have understood Hufsmith as teaching, or least suggesting, that the image analysis (e.g., static or dynamic analysis) detects sensitive data. Indeed, as I

EX1002, ¶160

'685 Ground 1 – The Combination Teaches Filtered/Prioritized Alerts

[1.8] report the detected vulnerabilities and sensitive data as alerts, wherein the alerts are filtered and prioritized based on the determined risk level of the virtual machine

Patent Owner

153. A POSA would not have found these teachings applicable to Veselov, where Veselov's post-deployment assessments do not have source code available to annotate. Kaeli¶152. And without the source code available, Hufsmith's stated

POR

In re Mouttet, 686 F.3d 1322, 1332 (Fed. Cir. 2012)

crossbar arrays. It is well-established that a determination of obviousness based on teachings from multiple references does not require an actual, physical substitution of elements. *In re Etter*, 756 F.2d 852, 859 (Fed.Cir.1985) (en banc) ("Etter's assertions that Azure cannot be incorporated in Ambrosio are basically irrelevant, the criterion being not whether the references could be physically combined but whether the claimed inventions are rendered obvious by the teachings of the prior art as a whole."); *In re Sneed*, 710 F.2d 1544, 1550 (Fed.Cir.1983) ("[I]t is not necessary that the inventions of the references be physically combinable to render obvious the invention under review."); *In re Keller*, 642 F.2d 413, 425 (CCPA 1981) ("The test for obviousness is not whether the features of a secondary reference may be bodily incorporated into the structure of the primary reference....").

Patent Owner

⁵ Wiz asserts that Hufsmith's IDE disclosures "broadly apply to other reporting techniques" (Pet.43-44) but never explains how or why POSAs would have been motivated to apply these or any other Hufsmith disclosures allegedly relating to alerts to Veselov's systems and methods. Kaeli¶¶152-153.

POR, ¶153 fn. 5

Petition, 24-25

issues. It was well known that filtering and prioritizing alerts, as taught by Hufsmith, facilitated rapid identification of higher-priority assets/risks and mitigated alert fatigue. EX1002, ¶110 (citing EX1078, ¶138; EX1045, ¶7; EX1046, 11:52-12:14; EX1063, ¶¶187-89, 207-11); *supra*, §IX.B (known benefits of prioritized alerts). Hufsmith itself explains such benefits. EX1078, ¶¶6, 155

sensitive data. EX1002, ¶¶61, 111. A POSA would have been motivated to obtain these benefits in the types of security scans taught by Veselov, since such scans would typically detect numerous security issues of varying importance.

'685 Ground 1 – Claims 2/14

[Claim 2] wherein during the analysis of the at least one snapshot, the virtual machine is inactive wherein the snapshot includes a page file of memory associated with the virtual machine

Dr. Stavrou

separate assessment VM instantiated from the snapshot. *Supra*, §XII.B.8. The analysis therefore does not depend on the state of the original VM during the analysis step. Moreover, a POSA would have expected many target VMs to already be ...
than 1.5 months). Thus, a POSA would have recognized that in many cases the target VM would be inactive during the analysis of the snapshot by Veselov's scanning system.

EX1002, ¶178

EX1066 (Zhou)

A significant portion of the VM images are mostly offline and infrequently booted. Thus, any attempt to start these VMs

2

EX1064 (Lutas)

protected guest VM. In typical conventional security solutions, execution of the protected virtual machine is suspended for the duration of event analysis (memory forensics, etc.). In contrast, in some embodiments of the present

¶79

'685 Ground 2 – Claims 1-22 Obvious over Veselov, Hufsmith, and Chari

Chari

[0061] Virtual machine images may contain substantial amounts of sensitive data belonging to a cloud computing environment, applications installed on the cloud computing environment, and users who have accounts on the cloud computing environment. Depending on the type of cloud computing environment and the applications and services that are installed, this sensitive data can include information at all layers of the cloud computing environment. Some examples are: raw information in virtual disk blocks; sensitive details in system configuration files including network topology information and administrator accounts and passwords; sensitive details in system logs, such as administrator passwords; application data, such as raw data in database files; and application credentials, such as default passwords for applications.

[0065] The sanitization system of illustrative embodiments includes a labeler and a sanitizer. The labeler labels sensitive data with sensitivity level labels within a virtual machine image based on labeling policies. For example, the labeler may label system logs or secure data with a high sensitivity level label. This labeling information is embedded within a virtual machine image and/or within a label file that specifies labels in a specific format. The sanitizer sanitizes the virtual

example, an extensible markup language. As a result, labeling scripts 228 are able to discover installation paths, files, and presence or absence of particular sensitive data items within, for example, data caches or system logs. Further, labeling

¶¶61, 65, 76

Chari

virtual machine image manager labels and sanitizes the virtual machine image at 904. The virtual machine image manager may utilize a labeler module and a sanitizer module, such as labeler 222 and sanitizer 224 in FIG. 2, to perform the labeling and sanitization processes on each of the software components of a software stack installed on the virtual machine image.

rupted. In addition, the virtual machine image manager labels and sanitizes the snapshot of the specific instance of the virtual machine image at 916 and stores the labeled and sanitized snapshot of the specific instance of the virtual machine image at 918.

¶¶92, 94

Dr. Stavrou

risks. The presence of unsecured sensitive data was a common concern in security assessments. Veselov involves cloud systems, and Chari observes that residual sensitive data is of particular concern to cloud systems. EX1007 (Veselov), 2:27-

228. Moreover, reporting filtered/prioritized alerts for the detected sensitive data would have furthered this benefit, since it would bring more critical sensitive data to the user's attention and improve remediation efforts. EX1056 (Chari), ¶¶39-

232. Furthermore, the VM risk levels discussed in Hufsmith also provided a logical and beneficial basis for filtering/prioritizing alerts for detected sensitive data.

EX1002, ¶¶227-28, 232

'685 Ground 3 – Claims 2-3 and 14-15 Obvious over Veselov, Hufsmith, Chari, Price, and Huseinović

[Claim 2-3]

- The system of claim 1, wherein during the analysis of the at least one snapshot, the virtual machine is inactive wherein the snapshot includes a page file of memory associated with the virtual machine.
- The system of claim 2, wherein the page file is configured to allow deduction of one or more applications running on the virtual machine.

Patent Owner

VM images. Kaeli¶¶81, 166. And, Wiz does not provide any evidence, beyond the '685 Patent itself, that POSAs would have understood that snapshots could be utilized for analyses of inactive VMs. Kaeli¶¶166.

POR, 66

EX1015 (VMware vSphere)

(on, off, or suspended). You can take a snapshot when a virtual machine is powered on, powered off, or suspended.

164

EX1062 (Almulla)

private cloud based VM snapshots (e.g. XenServer). Moreover, we examine snapshots using existing digital forensic tools and were able to successfully acquire data without the need to transform the snapshot files.

...

There are several advantages that snapshot technology presents; such as: (1) the ability to be seized offline, (2) minimal disturbance to business continuity during the course of investigation, and (3) the encapsulation of both memory and storage. However, the main drawbacks are: (1) Point in

724

Patent Owner

memory.” Pet.67. But Huseinović is directed to a particular scenario where “VMware workstation” is used, and explains that preserving a “vnm file” in a suspended state is an *alternative* to a “snapshot.” Ex.1049, 940 (describing “two natively supported options”). Wiz does not explain why a POSA would have implemented any of the combined systems in “VMware workstation,” or why it would have combined different alternative approaches. Kaeli¶167. Huseinović also

POR, 66-67

Huseinović

The same list was obtained from the second *.vmem* file created by taking a snapshot.

941

In re Mouttet, 686 F.3d 1322, 1332 (Fed. Cir. 2012)

crossbar arrays. It is well-established that a determination of obviousness based on teachings from multiple references does not require an actual, physical substitution of elements. *In re Etter*; 756 F.2d 852, 859 (Fed.Cir.1985) (en banc)

Secondary Considerations

Topics
PO Fails to Prove SideScanning Evidence Has Nexus
PO Fails to Prove Evidence Results from What is Both Claimed and Novel
PO Fails to Prove Long-Felt Need
PO Fails to Prove Skepticism
PO Fails to Prove Praise
PO Fails to Prove Copying
PO Fails to Prove Commercial Success

Nexus

***Fox Factory, Inc. v. SRAM, LLC*, 944 F.3d 1366, 1373-74 (Fed. Cir. 2019)**

In order to accord substantial weight to secondary considerations in an obviousness analysis, “the evidence of secondary considerations must have a ‘nexus’ to the claims, *i.e.*, there must be ‘a legally and factually sufficient connection’ between the evidence and the patented invention.” *Henny Penny Corp. v. Frymaster LLC*, 938 F.3d 1324, 1332 (Fed. Cir. 2019) (quoting *Demaco Corp. v. F. Von Langsdorff Licensing Ltd.*, 851 F.2d 1387, 1392 (Fed. Cir. 1988)). “The patentee bears the burden of showing that a nexus exists.” *WMS Gaming Inc. v. Int’l Game Tech.*, 184

***Lectrosonics, Inc. v. Zaxcom, Inc.*, IPR2018-01129, Paper 33, 33**

Patent Owner does not provide an analysis demonstrating that its products are coextensive (or nearly coextensive) with the challenged claims. We, therefore, find that a presumption of nexus is inappropriate. *See* Pet.

***SightSound Techs., LLC v. Apple Inc.*, 809 F.3d 1307, 1319 (Fed. Cir. 2015)**

F.3d 135, 140 (Fed.Cir.1996). If a product both “embodies the claimed features” and is “coextensive” with the claims at issue, “a nexus is presumed.” *Brown & Williamson*, 229 F.3d at 1130. In other words, a nexus exists if the commercial success of a product is limited to the features of the claimed invention. But “if the commercial success is due to an unclaimed feature of the device” or “if the feature that creates the commercial success was known in the prior art, the success is not pertinent.” *Ormco Corp. v. Align Technology*,

Nexus

***Fox Factory, Inc. v. SRAM, LLC*, 944 F.3d 1366, 1373-74 (Fed. Cir. 2019)**

re Huang, 100 F.3d 135, 140 (Fed. Cir. 1996). To the contrary, the patent owner is still afforded an opportunity to prove nexus *1374 by showing that the evidence of secondary considerations is the “direct result of the unique characteristics of the claimed invention.” *Id.*

***Janssen Pharms., Inc. v. Teva Pharms. USA, Inc.*, 97 F.4th 915, 933 (Fed. Cir. 2024)**

obviousness analysis. And a proper nexus analysis must “consider the correspondence between the objective evidence and the claim scope.” *Henny Penny Corp. v. Frymaster LLC*, 938 F.3d 1324, 1332 (Fed. Cir. 2019); *see also Merck & Cie v. Gnosis S.P.A.*, 808 F.3d 829, 837 (Fed. Cir. 2015) (“Where objective indicia result from something other than what is both *claimed* and *novel* in the claim, there is no nexus to the merits of the claimed invention.” (cleaned up) (first emphasis added)).

***Yita LLC v. MacNeil IP LLC*, 69 F.4th 1356, 1364 (Fed. Cir. 2023)**

Where prior art teaches a feature and a relevant artisan would have been motivated to use it in combination with other prior-art teachings with a reasonable expectation of success to arrive at the claimed invention—as the Board here found—a secondary consideration related exclusively to that feature does not logically undermine the inference from those premises that the claimed invention would have been obvious from the full body of prior art just because the feature was not *well known*.

REDACTED

REDACTED

REDACTED

REDACTED

REDACTED

REDACTED

REDACTED

PO Fails to Prove Evidence Results from What is Both Claimed and Novel

PO cites:

question is today. There is no doubt, the pivotal moment was the introduction of **agentless** scanning by companies like Orca and Wiz. Agentless scanning was a turning point - but anyone who tells you they knew it at the time is lying. We're old enough to

As discussed earlier, Orca launched the first version of **agentless side scanning**, which allowed deep visibility into cloud workloads without the need to install agents.

EX2076, 8, 11

PASSIVE MONITORING OF VIRTUAL SYSTEMS USING AGENT-LESS, OFFLINE INDEXING

EX1040 (Ammons)

virtual machine (i.e., its offline and not operating). Security tasks may nonetheless be able to be performed on the offline virtual machine by accessing one or more machine images

EX1048 (Price), ¶32

file-system data. Scanning the exposed physical hard disks is similar to, and more effective than, scanning the file-system data from within the virtual machine if an anti-malware agent were to be deployed inside the operating system. Data indica-

EX1129 (Michael), 2:28-48

ically used in IaaS clouds. In addition to classic functions like image browsing, access control and deployment, Mirage provides offline image content introspection, and manipulation capabilities. This is enabled by indexing the

EX1073 (Mirage), 1

REDACTED

REDACTED

PO Fails to Prove Evidence Results from What is Both Claimed and Novel

PO cites:

including on-premise devices, and now increasingly, cloud environments. The task of locating and classifying the entirety of an organization's data across all of the possible locales is no small endeavor and is generally proving unmanageable for legacy solutions such as data loss prevention (DLP) tools.

EX2152, 5

Discover More Data with Content-Aware Detection

Symantec DLP starts with a combination of advanced technologies that can accurately detect all of the confidential data in your organization—whether it's at rest, in motion, or in use. The detection technologies in Symantec DLP include:

EX1134, 1

Forcepoint Data Discovery can be used to learn the location of sensitive data both on-premises and in supported cloud-based applications. It can be used to scan data on file servers, email servers, and databases, as well as in content collaboration

EX1135, 5

[0038] Labeler 222 is a software module that labels sensitive data contained within virtual machines. Sensitive data is

rupted. In addition, the virtual machine image manager labels and sanitizes the snapshot of the specific instance of the virtual machine image at 916 and stores the labeled and sanitized snapshot of the specific instance of the virtual machine image at 918.

EX1056 (Chari), ¶¶38, 94

Detect

Content Analytics

Leverages keywords, pre-defined alphanumeric patterns, regular expressions, file metadata, document fingerprints, and database fingerprints to identify sensitive data.

EX1136, 1

Applies unified policies to all cloud services across data at rest and in transit. Leverage policy templates, import

EX1136, 2

PO Fails to Prove Nexus – No Connection to Recited/Novel Features

Dr. Stavrou

86. In sum, the features described in the secondary-considerations evidence cited by Dr. Kaeli were known, both individually and as a combination. In other words, the techniques that I describe above are not merely isolated features known in the limited context of agent-based scanning. Rather, as Veselov and the other agentless examples I discuss above make clear, these were known aspects of agentless workload scanning. *See, e.g.*, EX1007, 2:35-41, 4:1-4.

'031 EX1400

REDACTED

REDACTED

REDACTED

PO Fails to Prove Praise

Ingenico Inc. v. IOENGINE, LLC, IPR2019-00929, Paper 53, 100-01

We find Judge Bryson’s well-reasoned decision in *Sonos* instructive. *Sonos* draws a distinction between two types of industry praise. On one

...

On the other hand, “that principle does not justify the admission of the full text of every document that contains a favorable comment about Sonos or its products from any source. In order to be admissible, *the evidence must reflect praise in the industry, not just an individual’s favorable view of the product.*” *Sonos*, slip op. at 5 (emphasis added). As a result, “[d]ocuments that reflect simply favorable comments about a company and its products are more problematical” as they are “simply hearsay comments about . . . products.” *Id.* at 5, 7.

In this case, Exhibit 2114 reflects personal opinions of the author, Dan Harkabi. *See* PO Resp. 66. Even if Mr. Harkabi was an industry leader, nothing in Exhibit 2114 indicates that he is speaking on behalf of an industry group or presenting an industry award. *See* Ex. 2114. Rather, the

REDACTED

REDACTED

REDACTED

REDACTED

REDACTED

REDACTED

REDACTED

REDACTED

PO Baselessly Takes Credit for Wiz's Success

Wiz: Wiz pushed agentless scanning to new limits, most importantly by introducing the intuitive graphing of asset relationships to security teams, and building their entire UX around that capability. They also showed publicly

revenue within four years of its founding. Wiz differentiated itself by taking a graph-based approach to visibility, detection, high-fidelity risk prioritization alerts and an obsessive customer-driven approach, setting it apart from the competition. While the

A core factor in Wiz's success is its graph-based architecture, from the actual database to the user experience. Cloud environments are composed of many interconnected

It is widely known and should be emphasized Wiz launched a much more effective agentless scanner that helped teams identify vulnerabilities with incredible quick time-to-value and deployment for the largest enterprises. The underlying premise comes

Wiz's true strength is in their ability to come to market at exactly the right time to meet customers where they're at. Rather than sit on their agentless success, they introduced an agent when people started asking for it. They improved their DSPM capabilities as teams cared about it. Most recently, they announced Wiz Code just as security teams are catching onto the power of ASPM capabilities. Wiz is frightening to the market precisely because they're not just a history lesson stuck in what they invented, but continue to release just the right features at just the right time to appease the majority of the market.

EX2076, 7, 8, 21-22, 23