

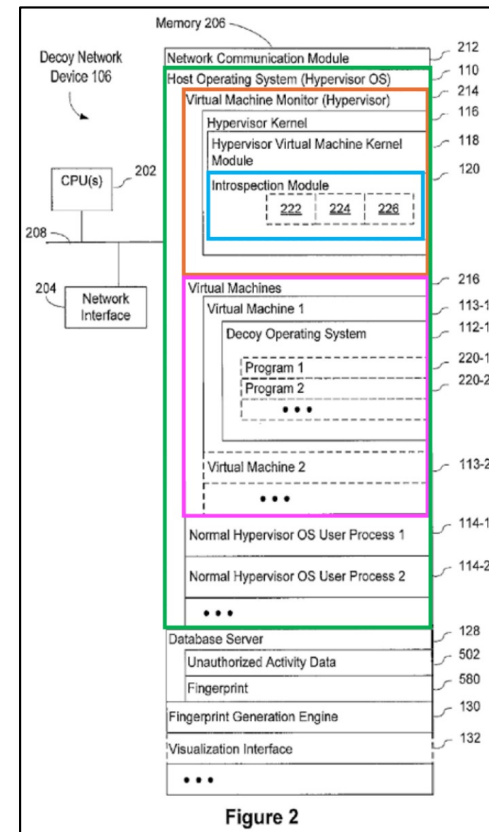
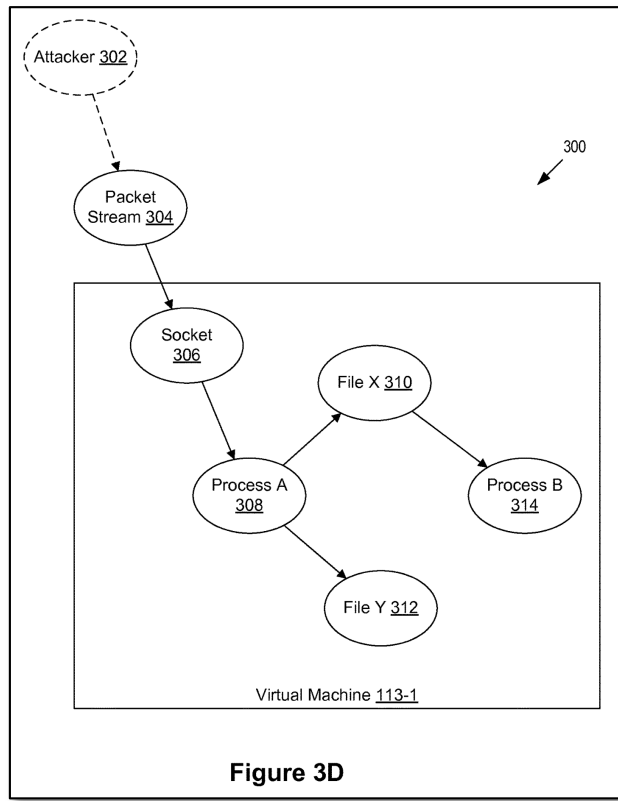
Inter Partes Review
U.S. Patent No. 9,954,872

Oral Argument, June 10, 2026

CrowdStrike, Inc. v. GoSecure, Inc.
Case No. IPR2025-00070

'872 Patent's Disclosure

- ▶ “[M]ethods for analyzing malicious activities on a computer system in order to better protect the computer system from future malicious activity.” ’872 Patent, 1:44-48
- ▶ Virtual machine (VM) on decoy device monitors unauthorized activities and a fingerprint of the unauthorized activities is generated. ’872 Patent, 2:27-48
- ▶ Fingerprint used to stop similar attacks on non-decoy devices. ’872 Patent, 2:44-48; 5:3-9



'872 Patent - Illustrative Claims

1. A computer implemented method of identifying unauthorized activities on a first computer system attached to a computer network, wherein the first computer system comprises one or more processors and memory, the method comprising:
 - monitoring activity on the first computer system;
 - identifying a plurality of activities being performed at the first computer system, wherein each of the activities includes an activity source, an activity target, and an association between the activity source and the activity target;
 - storing in the memory a data structure that identifies the activity sources, the activity targets, and the associations for the plurality of activities; and
 - transmitting to one or more computer systems other than the first computer system information identifying one or more of the activity sources, the activity targets, and the associations for preventing future attacks, to the one or more computer systems, associated with the one or more of the activity sources, the activity targets, and the associations.
2. The method of claim 1, wherein the monitoring further comprises monitoring all activity on the first computer system.
3. The method of claim 1, further comprising:
 - creating, from the stored information, a fingerprint indicative of the activity on the first computer system.

Capalik-Based Grounds Prevent Improper Claim Scope Recapture

- ▶ *Capalik & '872 Patent* have common inventor and both disclose same core concepts:
 - ▶ Decoy network w/ decoy devices running VMs
 - ▶ Attract and monitor attacks
 - ▶ Capture forensic data of attack
 - ▶ Generate signatures of attack from forensic data
 - ▶ Transmit attack info to protected network
 - ▶ Protected network
 - ▶ IDS/IPS
 - ▶ Protected end devices

Capalik-Based Grounds Prevent Improper Claim Scope Recapture (cont.)

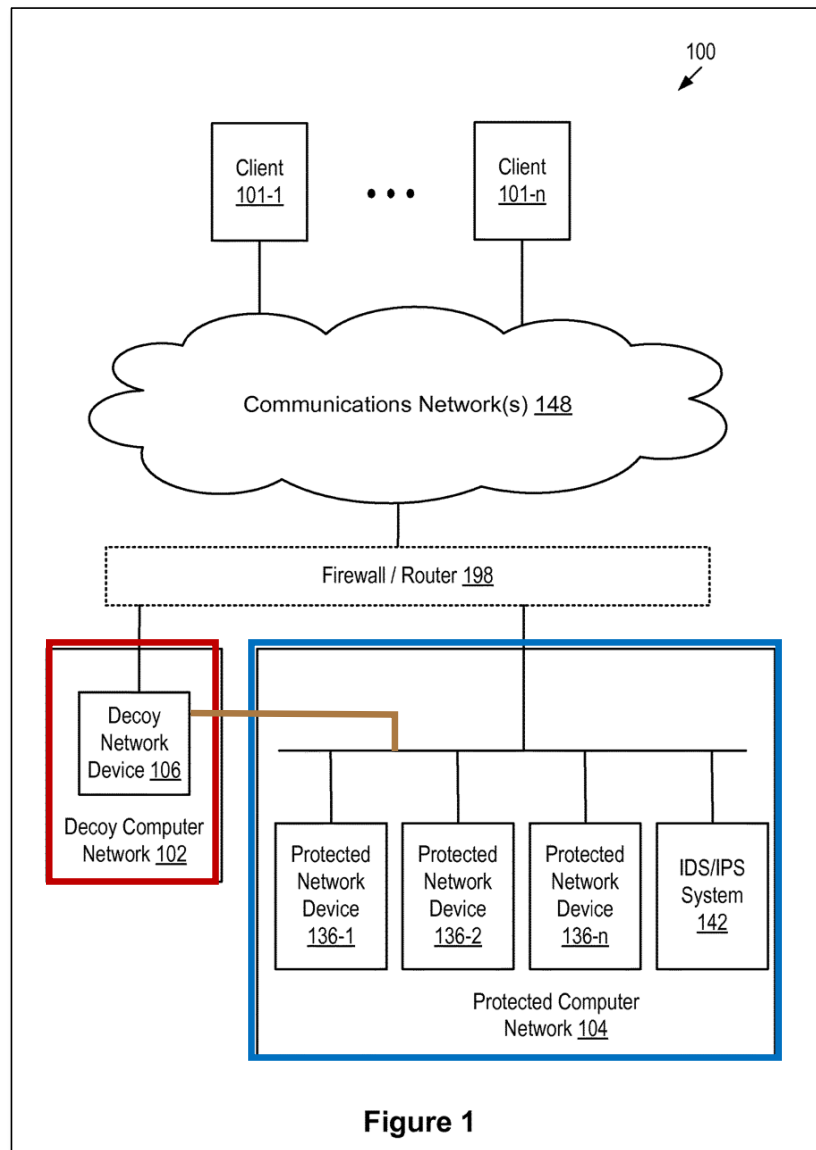


Figure 1

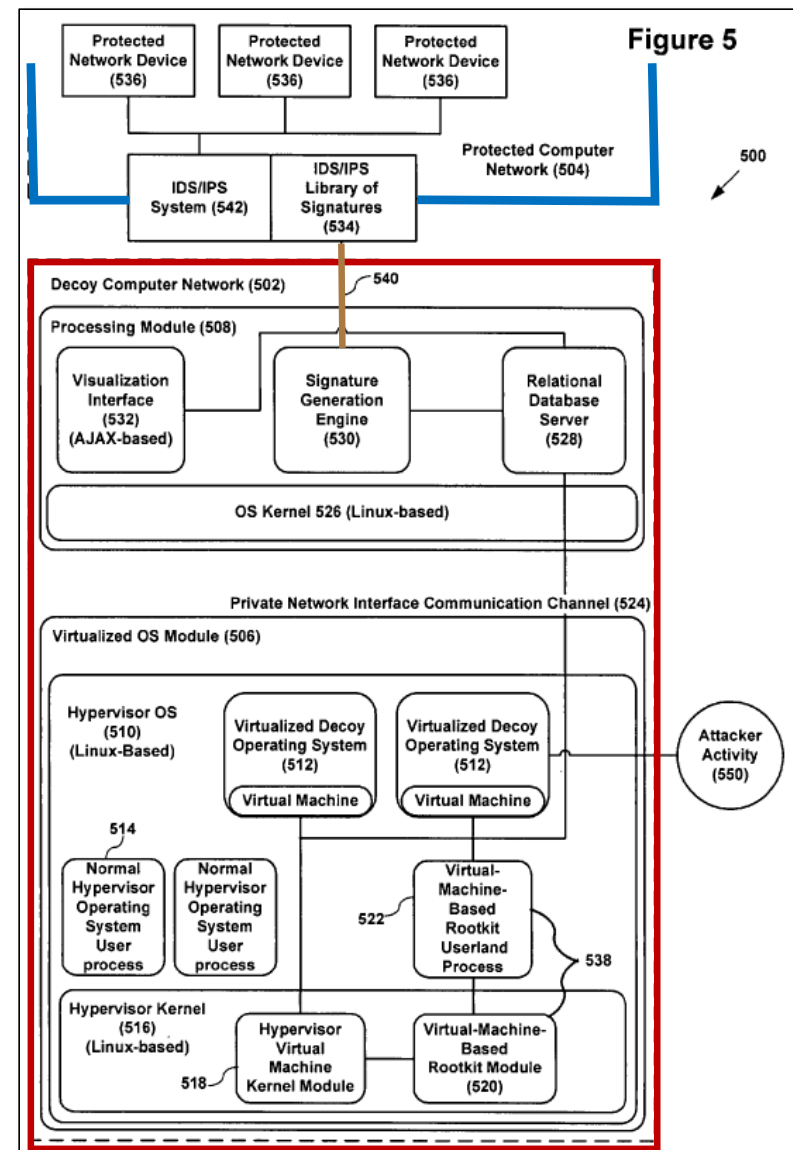


Figure 5

Petitioner's Grounds

Proposed Grounds	Basis	Disputed Claims
Ground 1: <i>Capalik</i>	§ 103	1-6, 12, 15, 20-21
Ground 2: <i>Capalik/Goel</i>	§ 103	1-6, 12, 14-15, 20-21
Ground 3: <i>Capalik/King</i> or <i>Capalik/Goel/King</i>	§ 103	7-11, 14, 16, 18-19
Ground 4: <i>Capalik/Pike</i> or <i>Capalik/Goel/Pike</i>	§ 103	13
Ground 5: <i>Capalik/Farley</i> or <i>Capalik/Goel/Farley</i>	§ 103	17

- ▶ Grounds 2-5: If PO disputes that *Capalik* identifies and stores a source, target, and association for each attack activity, *Goel* provides additional detailed disclosure.

Remaining Issues – Independent Claim I

- ▶ Computer System is not limited to a single endpoint device
 - ▶ *Capalik* sends attack information to “one or more computer systems other than the first computer system”
- ▶ *Capalik*’s forensic data is a data structure
 - ▶ The same structured “forensic data” is stored and used to generate fingerprints in both *Capalik* and the ’872 Patent
- ▶ The combination of *Capalik* and *Goel* is technically feasible
 - ▶ *Goel*’s process is not bodily incorporated into *Capalik*
 - ▶ The combination simply supplements which datapoints are stored
- ▶ PO’s secondary consideration arguments are unsupported
 - ▶ PO fails to show that CrowdStrike copied the ’872 Patent or that its success has a nexus to the ’872 Patent

Remaining Issues – Dependent Claims

▶ Ground 1

- ▶ Claim 2 - *Capalik* monitors all attack activity

▶ Ground 3

- ▶ Claim 8 – *King* “visually distinguish[es]” subset of associations by displaying only malignant associations
- ▶ Claim 16 – *King* “determin[es] an activity status” as “complete” before creating graph
- ▶ Claim 19 - Malignant association “types” are contained in *Capalik*’s forensic data structure, pursuant to *King*

▶ Ground 4

- ▶ Claim 13 - *Pike*’s DEP detects a request to access memory in the same manner as the ’872 *Patent*’s DEP

Computer System

Claim 1 “monitoring activity on the first computer system;...

transmitting [attack activity information] to one or more computer systems other than the first computer system ... for preventing future attacks, to the one or more computer systems”

Computer System: No New Mapping

Petition Mapped Protected Computer Network as Claimed Second/Other Computer System

A

POSITA would have further understood that *Capalik* transmits this information to a computer system other than computer decoy 502 since the signature is sent “through a standard network connection” to IDS/IPS 534 of protected network 504. *Capalik*, [0059], Fig. 5; *Dec.* ¶¶137-144.

Petition, 34

The parties do not agree about where information must be transmitted in order to perform the claimed step. *E.g.*, Prelim. Reply 1–3; Prelim. Sur-Reply 1–3. Petitioner contends that transmitting information to a “protected network, including protected endpoint devices and the IDS/IPS” falls within the scope of the claimed step. *E.g.*, Prelim. Reply 2. Patent Owner contends that the claimed step requires transmitting the information “to ‘protected network device’ that are actually the subject of future attacks.” *E.g.*, Prelim. Sur-Reply 3. According to Patent Owner, this does not encompass transmitting the information to an IDS/IPS. *Id.* at 1.

ID, 12

Capalik similarly teaches that once the attack signature is generated, it can be sent “to the intrusion detection system (IDS) or intrusion prevention system (IPS) for the protected network 504.” *Capalik*, [0059], Fig. 5; *see also id.*, [0052] (teaching that the entire process may occur automatically, without human intervention). The attack signature is then applied to the IDS/IPS library of signatures, updating the library, to prevent an attacker from accessing protected network 504 in a future similar attack. *Capalik*, [0059], *see also id.*, [0053] (teaching that “protected computer network 504 includes an IDS/IPS library of signatures 534 and an IDS/IPS system 542 coupled to multiple protected network devices 536”).

Dec., ¶143

Computer System: The '872 Patent's Varied Uses of "Computer System"

- ▶ '872 Patent uses "computer system" to refer to a subcomponent, a device, or a collection of devices

FIG. 1 is a high-level block diagram illustrating an exemplary distributed computer system in accordance with some embodiments.

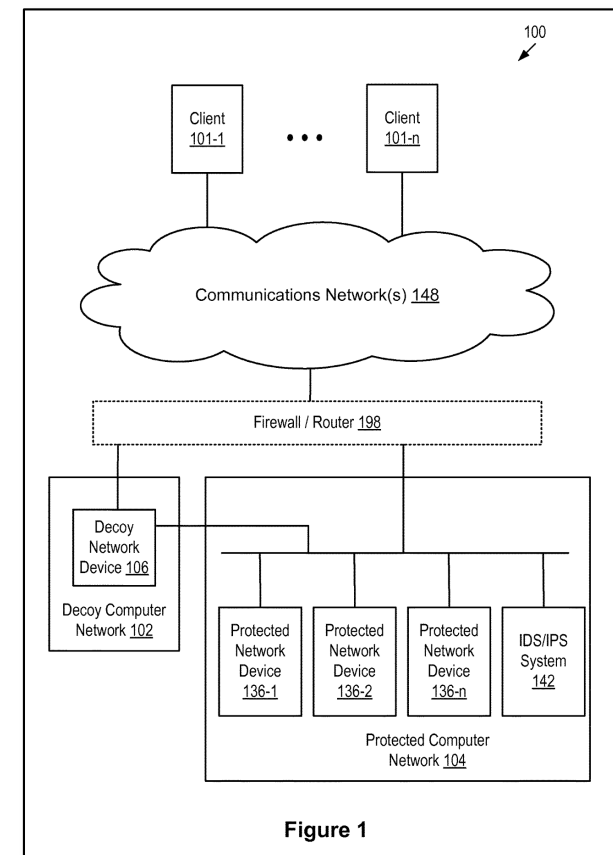
'872 Patent, 3:26-28

FIGS. 3A-3D and 4A-4E illustrate exemplary network diagrams representing unauthorized activity on a computer system (e.g., the virtual machine 113-1), in accordance with various embodiments.

'872 Patent, 11:15-18

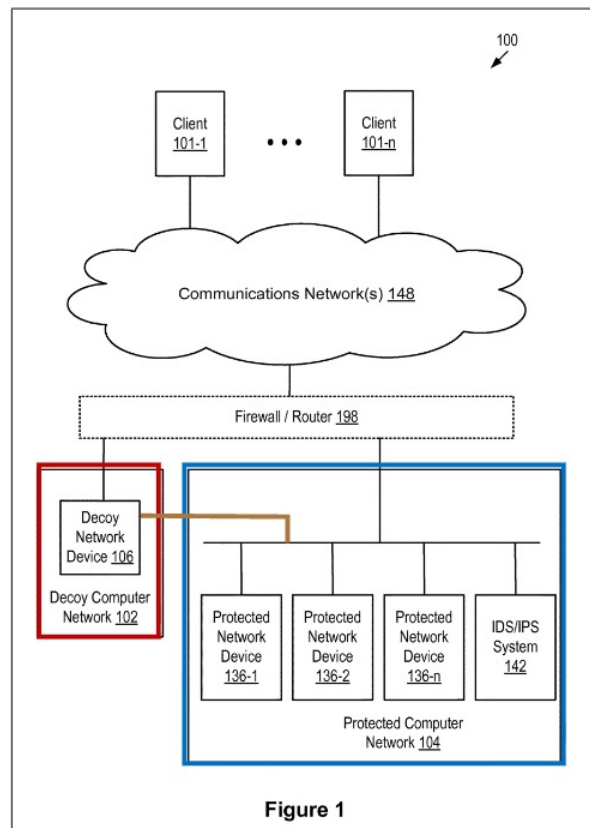
The virtual machines 216 are one or more software applications emulating one or more physical machines (e.g., computer systems).

'872 Patent, 7:59-61

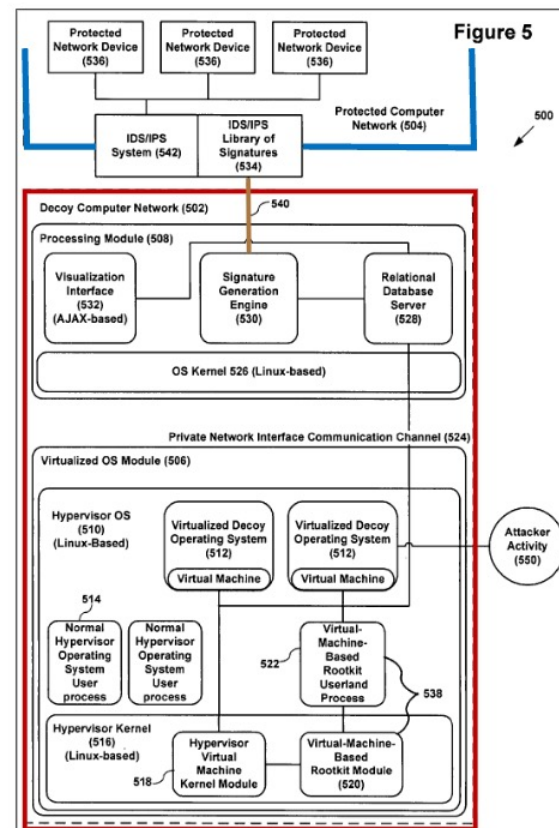


Computer System: *Capalik* and the '872 Patent Describe Nearly Identical Architectures

- Both transmit attack info from **decoy computer network** to **protected computer network**



'872 Patent, Fig. 1 (annotated)



Capalik, Fig. 5 (annotated)

Computer System: Extrinsic Evidence Cannot Supplant Unambiguous Intrinsic Record

As the *en banc* Federal Circuit held decades ago, extrinsic evidence⁴ may be considered in claim construction so long as it is “not used to contradict claim meaning that is unambiguous in light of the intrinsic evidence.” *Phillips v. AWH Corp.*, 415 F.3d 1303, 1324 (Fed. Cir. 2005) (en banc). Here, there is no ambiguity. The ’872 Patent ascribes a spectrum of meanings to “computer system,” and the claim language must be interpreted accordingly. The law does not permit supplanting these express uses with the meaning ascribed by *other references* describing *other systems*.

Reply, 15

Data Structure

Claim 1(c) “storing in the memory a data structure that identifies the activity sources, the activity targets, and the associations for the plurality of activities”

Claim 3 “creating, from the stored information, a fingerprint indicative of the activity on the first computer system”

Data Structure:

GoSecure's Secondary Indicia Copying Allegations Point to Attack Nodes and Connections as Evidence of Claimed Data Structure

CrowdStrike argues that “the claims do not impose any specific structure of the stored data.” Pet. 29. Not so. The limitation specifically requires storing a data structure “that identifies the activity sources, the activity targets, and the associations.” CrowdStrike has failed to show disclosure or obviousness of this requirement.

POR, 16

24 Q How did you interpret the phrase "data
25 structure" in limitation 1C for purposes of
1 assessing infringement?
2 MR. PETERS: Object to form.
3 A I looked at the graph database and I
4 compared -- and I asked the question, is the graph
5 database a data structure? And the answer was
6 yes.
7 Q (By Mr. Hart) So the graph database
8 is the data structure that you have mapped in
9 your analysis. Is that right?
10 A Yes, because it says storing -- I
11 clarify that further, storing the data in a graph
12 database. Identifies the sources, targets,
13 associations, relationships. And the graph
14 database can be considered a data structure.

Ex. 1034, 46:24-47:14
(Deposition of GoSecure's Expert, Dr. Stavrou)



Ex. 2017, (YouTube video at minute 2:11)

Data Structure:

PO's Expert Agrees "Forensic Data" Captures Attack Nodes and Connections

6 Q Does the patent provide an exemplar of
7 forensic data produced by the introspection
8 module?

9 A I believe it does. It does. It
10 illustrates activities. I believe absolutely if
11 you go -- one second. If you go to Figure 3A,
12 please, in the patent.
13 And this is an exemplar, right? It shows
14 the attacker. It shows the source here. It
15 shows the association and the target. So someone
16 who has observed that from introspection module
17 would be able to convert that to the desired data
18 structure we discussed in 1C.

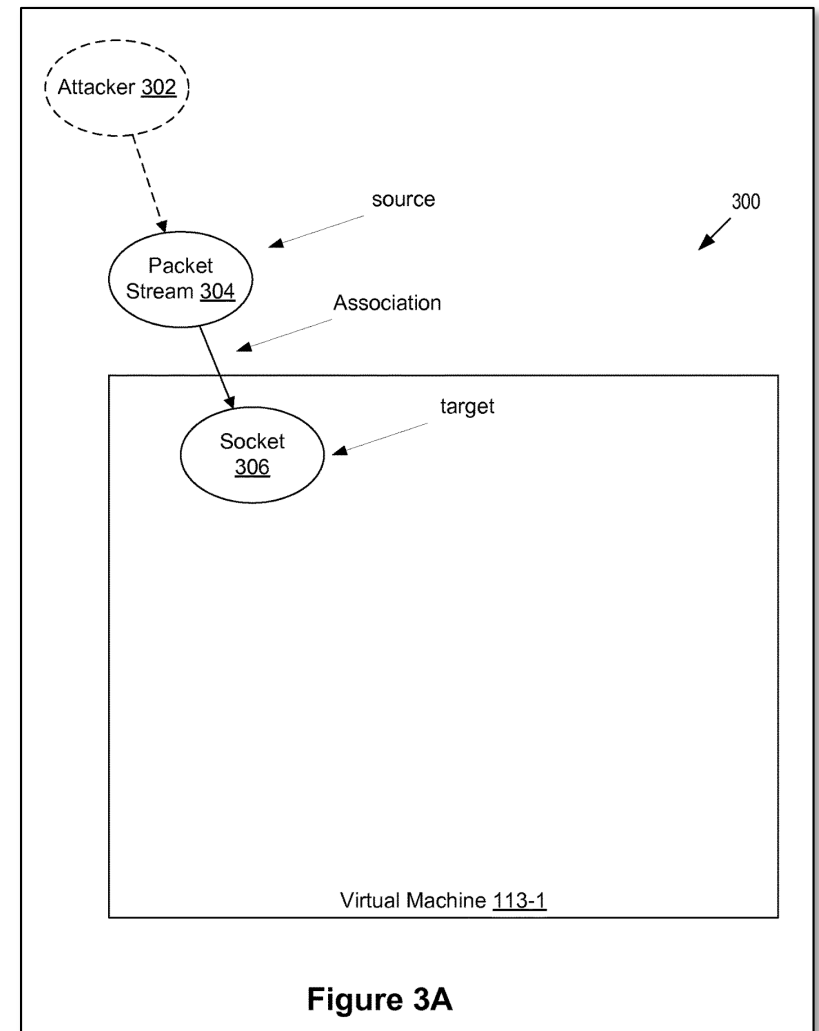


Figure 3A

'872 Patent, Fig. 3A

Ex. 1034, 93:6-18
(Deposition of GoSecure's Expert, Dr. Stavrou)

Data Structure:

Verbatim Forensic Data Disclosures in *Capalik* and '872 Patent

'872 Patent	Capalik
“The <i>introspection module</i> is configured to <i>examine the memory assigned to virtualized decoy operating systems 112</i> in order to <i>acquire low-level data about the</i> interaction between the decoy operating systems 112 and <i>attack activity</i>. . . [and] <i>generates and retrieves forensic data (or fingerprint)</i>.” Ex. 1001, 6:53-7:18.	“The <i>introspection module 538 . . . examines the memory assigned to virtualized decoy operating systems 512</i> in order to <i>acquire low-level data about the</i> interaction between the decoy operating systems and <i>attack activity 500</i>. . . [and] <i>generates and retrieves forensic data</i> for analysis by the processing module 508.” Capalik, [0048].
“[T]he introspection module captures (through the introspection) raw attack information. <i>The raw attack information is . . . stored</i> on the database server 128 <i>as forensic data 236</i> for later analysis. <i>The fingerprint generation engine 130</i> or the trace analyzer 226 <i>then uses this raw forensic information 236 to generate a signature of the attack</i>.” <i>Id.</i>, 10:30-36	“[T]he introspection module 538 captures (through introspection) attack information. <i>The attack information is . . . stored</i> on the relational database server 528 <i>as forensic data</i> for later analysis. <i>The signature generation engine 530 then uses this forensic data to generate a signature of the attack</i>.” <i>Id.</i>, [0052].

Data Structure:

Claims Track Disclosure re Creating Fingerprint from Forensic Data

1. A computer implemented method of identifying unauthorized activities on a first computer system attached to a computer network, wherein the first computer system comprises one or more processors and memory, the method comprising:

monitoring activity on the first computer system;
identifying a plurality of activities being performed at the first computer system, wherein each of the activities includes an activity source, an activity target, and an association between the activity source and the activity target;

storing in the memory a data structure that identifies the activity sources, the activity targets, and the associations for the plurality of activities; and

transmitting to one or more computer systems other than the first computer system information identifying one or more of the activity sources, the activity targets, and the associations for preventing future attacks, to the one or more computer systems, associated with the one or more of the activity sources, the activity targets, and the associations.

2. The method of claim 1, wherein the monitoring further comprises monitoring all activity on the first computer system.

3. The method of claim 1, further comprising:

creating, from the stored information, a fingerprint indicative of the activity on the first computer system.

Data Structure:

The '872 Patent Does Not Disclose a Process to Convert Forensic Data into a Data Structure

12 Q (By Mr. Hart) So your opinion is that
13 the data structure required Claim 1 uses the
14 forensic data created by the or generated by the
15 introspection module, but then organizes it in a
16 different way. Is that right?

17 A I believe it requires -- it might
18 require additional steps, like processing,
19 inference, a forensic information. All forensic
20 information by nature can be very big and also
21 might require additional processing.

22 I wouldn't know, because again this is
23 something that's not described. But I would say
24 that while the information might be present in a
25 forensic log, how you organize it, present it
1 and, you know, structure data structuring where
2 you store it, here identifying activity sources,
3 activity targets and associations, is what
4 separates it from the forensics data.

Ex. 1034, 71:12-72:4 (Deposition of GoSecure's Expert, Dr. Stavrou)

5 Q Okay. So the claim data structure
6 differs from the forensic data in how the data is
7 organized and presented. Is that right?

8 MR. PETERS: Object to form.

9 A Not exactly true. Again, it's
10 important that we understand that there is no
11 clear description here how you pass from the
12 forensic data to the -- there is no clear
13 explanation on how the data structure -- there is
14 clear requirement that there is a data structure
15 that identifies the activity sources, activity
16 targets and associations, which can be derived on
17 this case by the forensic information.

18 But how the data is analyzed, processed to
19 reach the point of creating that data structure
20 that has that information inside is not something
21 that is -- I would have understood that this is
22 feasible, but it's not clearly delineated as far
23 as I can tell.

Ex. 1034, 72:5-23

Motivation to Combine *Capalik* and *Goel*

A POSITA would have been motivated to supplement the information collected by *Capalik's* system pursuant to *Goel*

Motivation to Combine: Proposed Mapping Modifies Only Data Collected

PO mischaracterizes the combination, arguing that Goel's kernel-hook logging would be incorporated into Capalik's decoy OS. POR, 55-59. PO insists this hook-based monitoring would make Capalik's modified system detectable by attackers, which runs contrary to Capalik's goals. *Id.* Fatal to PO's criticism, the proposed combination does not add Goel's hook-based monitoring.⁶ It simply supplements (if necessary) the information collected by Capalik's monitoring system. Petition, 46 (noting the combination "utilize[s] Capalik's VMM, relational database, and visualization interface which already have monitoring, storing, and graphically displaying capability of unauthorized activities").

Reply, 21

Motivation to Combine:

'872 Patent and Capalik's Forensic Monitoring Already Slows the System—Addition from Goel Immaterially Modifies Detectability

⁶ PO does not argue that Capalik's monitoring *cannot* capture sources, targets, and associations. Nor could it. As set forth herein, Capalik and the '872 Patent describe identical introspection modules that each produce detailed forensic data of the attack. Nor could argue that Capalik's forensic module monitoring is completely undetectable by attackers. The '872 Patent confirms that this forensic monitoring slows the system enough to raise an attacker's suspicions. Ex. 1001, 7:43-53.

Reply, 21 n.6

An attacker may be able to detect that the decoy operating system 112 is a virtualized operating system, based on the time it takes to perform standardized tasks. However, since the introspection module 120 runs completely outside the virtualized decoy operating system 112, it is difficult for an attacker accessing the decoy operating system 112 to determine whether the decoy operating system 112 is being monitored. In other words, the attacker may suspect that the virtual machine monitor 214 exists, but the attacker may not determine whether the virtual machine monitor 214 includes the introspection module 120.

'872 Patent, 7:43-53

Secondary Considerations

PO secondary considerations arguments are unsupported

Secondary Considerations: No Evidence of Copying

- ▶ PO fails to provide evidence the Falcon Platform copies either the claimed data structure or architecture of the '872 Patent

First, although PO makes much of the reference to “Key Claims” in the presentation (POR, 69-71), the document included no claim language. Indeed, the Challenged Claims did not exist until years later. Ex. 1002 at 316–17. Second, the presentation provides the same level of detail about Capalik and the '872 parent application. Ex. 2100, 34, 36. So PO would have to—but cannot—demonstrate that Mr. Alperovitch took with him and implemented in CrowdStrike’s system the '872 Patent’s teachings *in isolation from* Capalik’s.

Reply, 27

16 Q Are those like-minded trusted
17 defenders not part of the Falcon Platform,
18 according to your analysis?
19 A It doesn't specify, sir. It can be or
20 it cannot be. I don't know. There's no
21 limitation here.

Ex. 1034, 38:16-21
(Deposition of GoSecure’s Expert, Dr. Stavrou)

Claim 2 – Monitoring All Activity

2.The method of claim 1, wherein the monitoring further comprises monitoring all activity on the first computer system.

Capalik monitors all attack activity

Claim 2 – PO’s Interpretation That Claim 2 Requires Monitoring the Monitoring Finds No Support in ’872 Patent

Critically, in both systems, *there is nothing to detect* unless the VM is being attacked. Ex. 1035, ¶¶4-10. The decoy device exists to be attacked, and until it has been attacked, there is no attack activity to monitor. *Id.* Neither Capalik nor the ’872 Patent describe monitoring activities outside a VM. Ex. 1035, ¶¶4-7.

Reply, 17

PO and its expert fatally conflate *monitoring activities* with *identifying activities as unauthorized*. The ’872 Patent’s process can trace activities from File X to Attacker 302 because all activities were monitored after attacker 302 accessed the VM through socket 306. Ex. 1035, ¶¶9-10. Monitored data must exist to trace back the connections depicted in Figs 4A-4D. Absent an attack, no data exists to be monitored. Accordingly, the ’872 Patent, like Capalik, monitors activities *after* an attack is received.

Reply, 19

Claim 2 - Capalik is Identical to '872 Patent—Only Attack Activity on the VM is Monitored

19 Is there any description of any computer
20 system in the '872 patent where activities are
21 monitored outside of virtual machines?
22 A There are no examples that show that.
23 That's correct.

Ex. 1034, 125:19-23
(Deposition of GoSecure's
Expert, Dr. Stavrou)

PO's argument

that Claim 2 requires monitoring activities outside the VM is inconsistent with all disclosed embodiments⁵ and should be rejected. *Vitronics Corp. v. Conceptronic, Inc.*, 90 F.3d 1576, 1583 (Fed. Cir. 1996) (a claim interpretation that excludes a preferred embodiment is “rarely, if ever, correct and would require highly persuasive evidentiary support.”)

Reply, 20

PO's first argument ignores that Capalik and the '872 Patent exclusively teach monitoring attack activity in a VM, which has activity to monitor *only* when attacked. Both describe decoy systems running VMs that are intentionally left susceptible to attacks, so attacks to the VMs can be monitored. Ex. 1001, 5:3-6 (“The purpose of the decoy network device 106 is to allow attackers to attack the decoy network device 106, so that the pattern of the attack can be monitored and analyzed to generate a fingerprint.”); Capalik, [0045] (“The instigator of the attack 550 is thus lured into accessing the decoy 512, which is overseen by the hypervisor operating system[.]”).

Reply, 16-17

Claim 8 – “Visually Distinguishing” a Subset of Associations

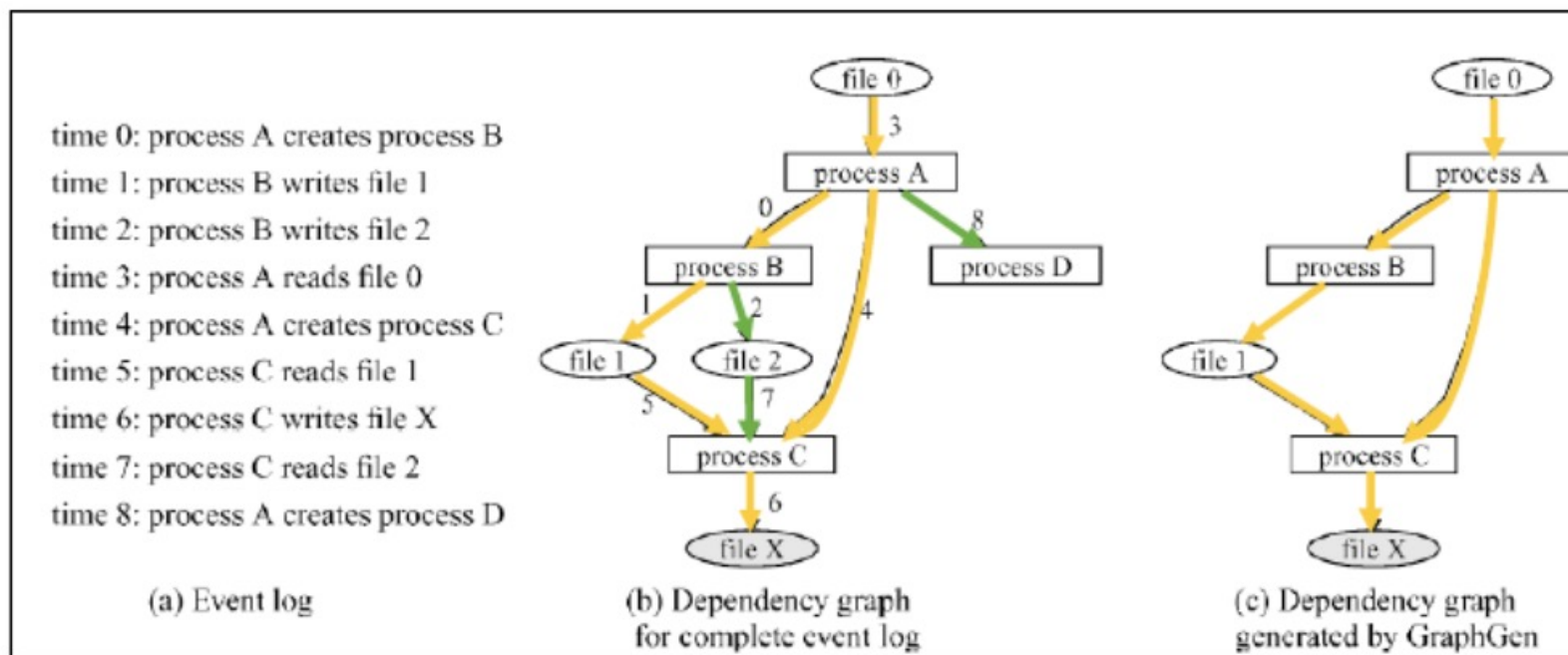
7. The method of claim 1, further comprising: graphically displaying at least a subset of the stored activities.

8. The method of claim 7, wherein the displaying includes visually distinguishing at least a subset of the associations.

King “visually distinguish[es]” subset of associations by displaying only malignant associations

Claim 8 – King’s Fig. 3.2(c) Graph Visually Distinguishes a Subset of Associations

As explained in the Petition, King’s full dependency graph in Fig. 3.2(b) below depicts malignant (annotated **yellow**) and benign (annotated **green**) associations, while its filtered dependency graph in Fig.3.2(c) depicts only the subset of associations that are malignant. Petition, 54-55.



Claim 8 – King’s Fig. 3.2(c) Graph Visually Distinguishes a Subset of Associations (cont.)

7. The method of claim 1, further comprising:
graphically displaying at least a subset of the stored activities.

8. The method of claim 7, wherein the displaying includes visually distinguishing at least a subset of the associations.

When King displays the full Fig. 3.2(b) dependency graph, it displays at least a *subset of the stored activities*, but does not distinguish between malignant and benign associations, satisfying Claim 7, but not Claim 8. And when King displays the filtered Fig. 3.2(c) dependency graph it both displays at least a *subset of the stored activities*, satisfying Claim 7, and visually distinguishes the *subset of associations* that are malignant, satisfying Claim 8.

Claim 16 – Determining an Activity Status

16. [] determining an activity status for the stored events, the activity status comprising ...:
a complete status, representing that a source of unauthorized activities is identified, and each activity target in the stored activities is associated with the source through one or more associations;

King “determin[es] an activity status” as “complete” before creating graph

Claim 16 – *King* Determines a “Complete” Status

16. The method of claim 1, further comprising:
determining an activity status for the stored events, the activity status comprising one of:
a complete status, representing that a source of unauthorized activities is identified, and each activity target in the stored activities is associated with the source through one or more associations; and

The Petition explained that

King’s filtered dependency graph Fig 3.2(c) is created *only after* the system has determined a complete status, identifying (1) a source of unauthorized activities and (2) associations to every activity target. Petition, 63-64; *see also* Ex. 1003, ¶231 (explaining that the process of reaching the filtered graph Fig. 3.2(c) includes determining “a complete status, as claimed, since all associations between each source and target would be identified as malicious or benign before finalizing the graph indicating the path of the invasion.”).

Claim 19 – “Type” in the Identified Information

19. The method of claim 1, wherein the association includes a type that represents whether or not the activity source has modified the activity target to perform the unauthorized activities.

Malignant association “types” are contained in *Capalik’s* forensic data structure, pursuant to *King*

Claim 19 - “Type” in the Identified Information: *King* Teaches “Type” of Association

- ▶ *King* teaches determining whether an association contributes to unauthorized activity (a “type”). *Petition*, 64-66

In other words, *King*’s filtered dependency graph in Fig. 3.2(c) removes any associations that did not contribute (directly or indirectly) to the unauthorized activity identified at time 10 (i.e., suspicious content written into file X). *Dec.*, ¶235. A POSITA would have understood that the associations remaining after this filtering are of a type that represents a source modifying its target to perform unauthorized activities either directly (e.g., writing incorrect content into file X) or indirectly (e.g., causing a subsequent source to write incorrect content into file X). *Id.*

Petition, 65-66

Claim 19 - “Type” in the Identified Information: No Additional Data Structure Requirement

- ▶ *King’s* type is stored in *Capalik’s* structured forensic data
- ▶ Claim 19 modifies the “association” of Claim 1
- ▶ Claim 19 does not impose additional requirements on the data structure recited in Claim 1(c)

19. The method of claim 1, wherein the association includes a type that represents whether or not the activity source has modified the activity target to perform the unauthorized activities.

'872 Petition, Claim 19

Claim 13 – *Pike's* DEP

13. The method of claim 12, wherein the identifying activity sources that are affected by unauthorized activities includes identifying activity sources that request to access a portion of memory that is set as non-executable.

Pike's DEP detects a request to access memory in the same manner as the '872 *Patent's* DEP

Claim 13 - Pike's DEP:

***Pike's* and the '872 Patent's DEP Identify Activity Sources That Request Access to Memory**

In some embodiments, the identifying activity sources that are affected by unauthorized activities includes (616) identifying activity sources that request to access a portion of memory that is set as non-executable. This can be achieved by data execution prevention (DEP) mechanisms.

'872 Patent, 16:10-14

As discussed below, *Pike* teaches one such DEP mechanism. *Pike* discloses scanning “input strings” (e.g. data packets), identifying subsequences in the input strings that could be maliciously exploited, and configuring the target computer system to trigger a fault (or other triggered event) when certain execution conditions suggest use of data or code implicated by the identified subsequences. *Pike*, 3:12-19, 5:13-38, 8:65-67, 2:7-33. *Pike* discloses several techniques to trigger an action when memory of the target system is accessed by the potentially dangerous data in the input string. *Pike*, 8:65-9:5.

Petition, 67