

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

WIZ, INC.,
Petitioner,

v.

ORCA SECURITY LTD,
Patent Owner.

Case IPR2025-00442
Patent No. 11,627,154

DECLARATION OF DR. ANGELOS STAVROU

TABLE OF CONTENTS

I.	ENGAGEMENT.....	1
II.	QUALIFICATIONS.....	1
III.	COMPENSATION.....	3
IV.	INFORMATION CONSIDERED.....	3
V.	LEGAL PRINCIPLES.....	4
VI.	THE RELEVANT TIMEFRAME FOR ANALYSIS.....	7
VII.	PERSON OF ORDINARY SKILL IN THE ART.....	8
VIII.	OVERVIEW OF THE '154 PATENT.....	9
	A. Challenged Claims.....	13
	B. Prosecution History of the '154 Patent.....	15
IX.	CLAIM CONSTRUCTION.....	16
X.	STATE OF THE ART.....	16
XI.	OVERVIEW OF THE PRIOR ART.....	27
	A. Keren (U.S. Patent No. 11,374,982, EX1005).....	27
	B. Morgan (U.S. Patent No. 12,015,631, EX1006).....	31
XII.	GROUND 1: CLAIMS 1-20 WERE OBVIOUS OVER KEREN AND MORGAN.....	37
	A. Reasons to Combine Keren and Morgan.....	39
	B. Independent Claims 1, 14, and 20.....	52
	1. Preambles.....	52
	2. Element 1.i.....	62
	3. Elements 1.1, 14.1, and 20.1.....	63

4.	Elements 1.2, 14.2, and 20.2.....	65
5.	Elements 1.3, 14.3, and 20.3.....	70
6.	Elements 1.4, 14.4, and 20.4.....	75
7.	Elements 1.5, 14.5, and 20.5.....	80
8.	Elements 1.6, 14.6, and 20.6.....	89
9.	Elements 1.7, 14.7, and 20.7.....	100
C.	Dependent Claims	107
1.	Claims 2 and 15.....	107
2.	Claims 3 and 16.....	108
3.	Claims 4 and 17.....	109
4.	Claims 5 and 18.....	114
5.	Claims 6 and 19.....	117
6.	Claim 7.....	118
7.	Claim 8.....	120
8.	Claim 9.....	123
9.	Claim 10.....	124
10.	Claim 11.....	124
11.	Claim 12.....	125
12.	Claim 13.....	126
XIII.	CONCLUDING STATEMENTS.....	128
XIV.	APPENDIX A – MATERIALS CITED.....	129

I, Angelos Stavrou, declare as follows:

I. ENGAGEMENT

1. I have been retained by counsel for Wiz, Inc. as an expert witness in the above-captioned proceeding. I have been asked to provide my opinion about the state of the art of the technology described in U.S. Patent No. 11,627,154 (the “’154 patent”) and on the patentability of claims 1-20 of this patent. The following is my written testimony on these topics.

II. QUALIFICATIONS

2. I received my M.Sc. in Electrical Engineering, M.Phil., and Ph.D. (with distinction) in Computer Science all from Columbia University. I also hold an M.Sc. in theoretical Computer Science from the University of Athens and a B.Sc. in Physics with distinction from the University of Patras, Greece.

3. I am a Virginia Tech Innovation Campus founding Professor, and the Entrepreneurship activities lead. I am also a member of the Bradley Department of Electrical & Computer Engineering at Virginia Tech. From 2017 to 2020, I was a Professor in the Computer Science Department at George Mason University (“GMU”), teaching courses including Operating Systems Security and Cyber Security Laboratory. From 2012 to 2017, I was an Associate Professor in GMU’s Computer Science Department, teaching courses including Operating Systems Security, Enterprise Security Practices, and Enterprise Security Technology. From

2014 to 2017, I was an Academic Director in GMU's School of Management for the M.S. in Management of Secure Information Systems Program. From 2013 to 2015, I was an Academic Director in GMU's Computer Science Department for the M.S. in Information Security and Assurance Program. From 2011 to 2020, I was an Associate Researcher in the Computer Security Division at the National Institute of Standards and Technology ("NIST"). I am a co-author of numerous publications involving virtualization and cyber security, among other topics. EX1003 (CV), 3-19.

4. I am also the founder of Quokka, Kryptowire Labs, Aether Argus, and Impedyme Inc. I have served as a principal investigator on research awards from NSF, DARPA, IARPA, DHS, AFOSR, ARO, ARL, and ONR. I have written more than 140 peer-reviewed conference and journal articles. Furthermore, I am an inventor of nineteen issued patents and several pending patent applications. I am an Associate Editor of IEEE Transactions on Computers, IEEE Security & Privacy, and IEEE Internet Computing magazine and a co-chair of the IEEE Blockchain initiative. I am a senior member of the ACM, USENIX, and IEEE. My current research interests include security and reliability for distributed systems, security principles for virtualization, and anonymity, focusing on building and deploying large-scale systems.

5. I received the GMU Department of Computer Science Outstanding Research Award in 2010, 2016, and 2018. Also, I was awarded the 2012 George Mason Emerging Researcher, Scholar, Creator Award, a university-wide award. In 2013, I received the IEEE Reliability Society Engineer of the Year award. In 2024, I received the IEEE Reliability Society Lifetime Award. My team at Kryptowire was awarded the DHS Cyber Security Division's "Significant Government Impact Award" in 2017 and the "Bang for the Buck Award" in 2019. Currently, I am the primary Principal Investigator (PI) in two DARPA awards focusing on Cyber Security for cloud and wireless systems, namely the DARPA OPS-5G and CASTLE efforts.

III. COMPENSATION

6. I am being compensated for my time at my standard consulting rate. My compensation is not contingent upon the results of my study and analysis, the substance of my opinions, or the specifics of my testimony. I have no financial interest in the outcome of this matter or in any litigation involving the '154 patent.

IV. INFORMATION CONSIDERED

7. My opinions are based on my years of education, research and experience, as well as my investigation and study of relevant materials. In preparing this declaration, I have reviewed the '154 patent and the '154 patent's file history

(EX1004). I have also reviewed the materials cited in this declaration, a list of which is provided in Appendix A.

V. LEGAL PRINCIPLES

8. I understand that a claim is not patentable under 35 U.S.C. §102 as anticipated if the claim as a whole is described, either expressly or inherently, in a single prior art reference. I understand that this also requires that the prior art reference disclose that each claim element is arranged as recited in the claims.

9. I have been advised that a claimed invention is not patentable under 35 U.S.C. §103 if it is obvious. A patent claim is unpatentable if the claimed invention would have been obvious to a person of ordinary skill in the field at the time the claimed invention was made. This means that even if all of the requirements of the claim cannot be found in a single prior art reference that would anticipate the claim, a person of ordinary skill in the relevant field who knew about all this prior art would have come up with the claimed invention.

10. I have further been advised that the ultimate conclusion of whether a claim is obvious should be based upon several factual determinations. That is, a determination of obviousness requires inquiries into: (1) the level of ordinary skill in the field; (2) the scope and content of the prior art; (3) what difference, if any, existed between the claimed invention and the prior art; and (4) any secondary evidence bearing on obviousness.

11. I have been advised that, in determining the level of ordinary skill in the field that someone would have had at the time the claimed invention was made, I should consider: (1) the levels of education and experience of persons working in the field; (2) the types of problems encountered in the field; and (3) the sophistication of the technology.

12. I have also been advised that, in determining the scope and content of the prior art, in order to be considered as prior art, a reference must be reasonably related to the claimed invention of the patent. A reference is reasonably related if it is in the same field as the claimed invention or is from another field to which a person of ordinary skill in the field would look to solve a known problem.

13. I have been advised that a patent claim composed of several elements is not proved obvious merely by demonstrating that each of its elements was independently known in the prior art. In evaluating whether such a claim would have been obvious, I may consider whether there are reasons that would have prompted a person of ordinary skill in the field to combine the elements or concepts from the prior art in the same way as in the claimed invention.

14. I have been further advised that there is no single way to define the line between true inventiveness on the one hand (which is patentable) and the application of common sense and ordinary skill to solve a problem on the other hand (which is

not patentable). For example, market forces or other design incentives may be what produced a change, rather than true inventiveness. I may consider whether the change was merely the predictable result of using prior art elements according to their known functions, or whether it was the result of true inventiveness. I may also consider whether there is some teaching or suggestion in the prior art to make the modification or combination of elements claimed in the patent. I may consider whether the innovation applies a known technique that had been used to improve a similar device or method in a similar way. I may also consider whether the claimed invention would have been obvious to try, meaning that the claimed innovation was one of a relatively small number of possible approaches to the problem with a reasonable expectation of success by those skilled in the art.

15. I have also been advised, however, that I must be careful not to determine obviousness using the benefit of hindsight; many true inventions might seem obvious after the fact. I should put myself in the position of a person of ordinary skill in the field at the time the claimed invention was made and I should not consider what is known today or what is learned from the teaching of the patent.

16. Finally, I have been advised that any obviousness rationale for modifying or combining prior art must include a showing that a person of ordinary skill would have had a reasonable expectation of success.

17. With regard to secondary considerations of nonobviousness, I have been advised that any objective evidence may be considered as an indication that the claimed invention would not have been obvious at the time the claimed invention was made. I understand that the purpose of secondary considerations is to prevent a hindsight analysis of the obviousness of the claims.

18. I have been advised that there are several factors that may be considered as a secondary consideration. These factors include the commercial success of the invention, industry praise for the invention, skepticism of the invention, licensing of the invention, copying of the invention, any long-felt need that the invention solved, failure of others, and unexpected results of the invention.

19. I have been further advised that in order for evidence of secondary considerations to be significant, there must be a sufficient nexus between the claimed invention and the evidence of secondary considerations. I understand that this nexus serves to provide a link between the merits of the claimed invention and the evidence of secondary considerations provided.

VI. THE RELEVANT TIMEFRAME FOR ANALYSIS

20. I understand that the '154 patent issued from U.S. Application No. 17/657,981 ("the '981 application"), filed April 5, 2022. EX1001, Face. The '981 application claims priority to Provisional Application No. 63/180,048, filed April

26, 2021. I have been asked to treat April 26, 2021 as the effective filing date for purposes of my analysis.

VII. PERSON OF ORDINARY SKILL IN THE ART

21. As I noted earlier in §V, I have been advised that, in determining the level of ordinary skill in the field that someone would have had at the time the claimed invention was made, I should consider: (1) the levels of education and experience of persons working in the field; (2) the types of problems encountered in the field; and (3) the sophistication of the technology.

22. In my opinion, a person of ordinary skill in the art (“POSA”) at the relevant time would have had (1) at least a bachelor’s degree in computer science, computer engineering, electrical engineering, or a related degree, and (2) would also have 2-3 years of professional experience with cyber security analysis and virtualization. Such experience includes, for example, malware analysis, security analysis of cloud computing systems, and security analysis of virtual machines. I satisfied these requirements before April 26, 2021. *See supra*, §II. Additional experience can compensate for less education and vice versa. The State-of-the-Art section below summarizes some basic background knowledge of the type that a POSA would have had. *See infra*, §X.

VIII. OVERVIEW OF THE '154 PATENT

23. The '154 patent is entitled “Forward and Rearward Facing Attack Vector Visualization.” EX1001, Face. The '154 patent discloses techniques “for securing cloud infrastructure to help meet compliance mandates, without disrupting business operations in live environments.” *Id.*, 1:15-18. As part of these techniques, the '154 patent describes “a system for performing visualization of forward and backward facing threats,” as illustrated in Fig. 8, reproduced below. *Id.*, 45:38-40.

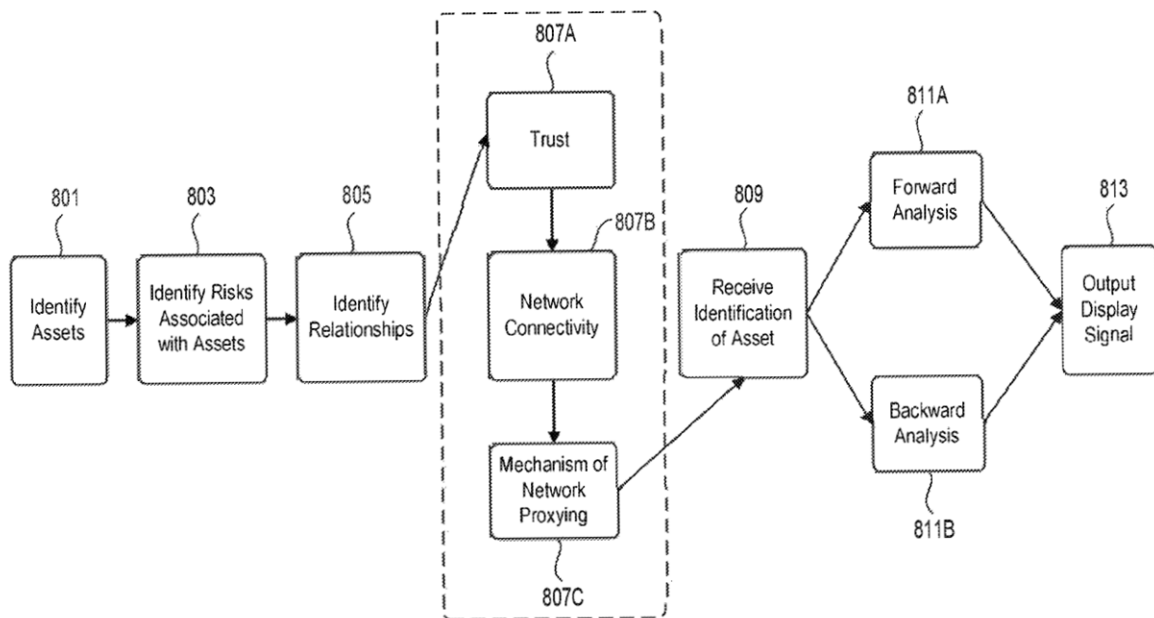


FIG. 8

Id., Fig. 8.

24. As shown above, the '154 patent explains that the system first identifies assets in a cloud environment. *Id.*, 45:55-46:3. According to the '154 patent, the

assets may include “at least one of a virtual machine, a network appliance, a storage appliance, a compute instance, or an engine instance.” *Id.*, 47:25-42; *see also id.*, 10:26-30 (“an asset (e.g., a system or device)”), 13:8-18, 45:55-46:3, 49:3-17 (assets in a cloud environment may include virtual machines, databases, storage, keystores, and load balancers), 65:11-21 (asset categories may include categories for accounts, containers, databases, images, managed services, messaging services, monitoring services, networks, storage, users, access, virtual machines, or serverless systems), Fig. 1, claim 3. The assets may be identified “based on at least one of an identity and access management policy, an organization policy, or an access policy.” *Id.*, 47:43-54. Furthermore, the cloud environment may be a service provided by a third party, such as “Amazon Web Services, Microsoft Azure, Google Cloud, Alibaba Cloud, IBM Cloud, Oracle, Salesforce, SAP, or similar service.” *Id.*, 45:66-46:3.

25. The system of the ’154 patent is also configured to “identify risks 803 associated with each of the identified assets.” *Id.*, 46:4-15. The system may “detect a potential risk associated with the specific asset based on a network activity of the specific asset” and the potential risk may be “based on historical threats to the specific asset or a list of possible threats to the specific asset based on its current network activities.” *Id.*, 48:28-34. Additionally, the system may further detect asset risk based on the network activity of upstream or downstream assets such as by

determining if the downstream asset poses a risk to the system based on entry points of risk. *Id.*, 48:35-55.

26. The system is also configured to “identify relationships 805 between at least some of the identified assets,” which can include “at least one of a trust 807A” (e.g., “cryptography, digital signatures, electronic certificates, or similar method of establishing a trusting relationship”), “a network connectivity 807B” (e.g., “wireless or wired connections”), or “a mechanism of network proxying 807C” (e.g., “established through any number of connected servers to accomplish proxying”). *Id.*, 46:16-29; *see also id.*, 49:30-43.

27. After identifying asset relationships, the system is further configured “to receive an identification 809 of a specific asset under investigation” and perform “a forward analysis 811A” and “a backward analysis 811B” of the specific asset. *Id.*, 46:30-47:4; *see also id.*, 49:44-52 (“The specific asset under investigation may be a physical or virtual machine that has software operating a specific version that is vulnerable to any number of associated risks 803.”). The ’154 patent explains that the forward analysis “identif[ies] at least one possible attack vector reaching the specific asset via a network outside the cloud environment.” *Id.*, 46:39-53; *see also id.*, 48:63-65, 49:53-50:3. For example, a forward analysis may involve “external entities that may access an analyzed asset” and “may identify new or existing threats

and attack vectors currently threatening the associated system and workload” from a network outside of the cloud environment (e.g., the internet). *Id.*, 46:49-53; *see also id.*, 47:19-21 (network outside cloud environment “includes the Internet”).

28. Similarly, the backward analysis of an asset includes “the possible impacts to an analyzed asset and those impacts’ possible effect” and may identify “at least one exposure risk” to a downstream asset, such as “an identification of an exposed asset, an entry point to the exposed asset, and lateral movement risks associated with the exposed asset.” *Id.*, 46:54-67, 48:65-67; *see also id.*, 50:4-19. An entry point may be used to “access the exposed asset and any related machines potentially vulnerable to attack from the analyzed and exposed asset.” *Id.*, 46:67-47:4; *see also id.*, 48:17-19 (“An entry point of risk may involve a port of access to the workload....”). A lateral movement risk identifies “potential lateral movement between assets,” for example, where “[a]n attacker who establishes a network foothold usually attempts to move laterally from one resource to another in search of rich targets such as valuable data.” *Id.*, 61:19-39; *see also id.*, 15:14-15, 69:26-28 (“A lateral movement may include any pathway between assets where a cybersecurity risk can travel from one affected asset to another affected asset.”).

29. Finally, the ’154 patent describes that the system is configured to display “a presentation of forward and backward paths associated with the specific

asset, thereby enabling visualization of a plurality of entry points and lateral movement risks associated with the plurality of entry points.” *Id.*, 47:5-10. The visualization may include “graphical, sequential, or multi-dimensional displays of attack vectors.” *Id.*, 45:41-54; *see also id.*, 47:10-13 (display of paths may be “one or several visual displays such as linear, graphical, or through computer-generated images”). Moreover, the “visualization of a plurality of entry points and lateral movement risks with the plurality of entry points may include visualization of the plurality of paths [and] may sequence the paths based on likelihood of access via that path or the severity of potential threat in a path of access to an asset.” *Id.*, 47:13-18.

A. Challenged Claims

30. The ’154 patent has 20 claims, with claims 1, 14, and 20 being independent. I understand that each of claims 1-20 is being challenged in the *inter partes* review proceeding related to this declaration. Below, I reproduce the text of independent claim 1 with numbering added to certain claim elements for ease of reference.

A graphical user interface system for providing forward and backward facing attack vector visualizations, the system comprising:

[1.i] at least one processor configured to:

[1.1] identify assets in a cloud environment;

[1.2] identify risks associated with each of the identified asset;

[1.3] identify relationships between at least some of the identified assets, the relationships including at least one of a trust, a network connectivity, or a mechanism of network proxying;

[1.4] receive an identification of a specific asset under investigation;

[1.5] perform a forward analysis of the specific asset under investigation to identify at least one possible attack vector reaching the specific asset via a network outside the cloud environment;

[1.6] perform a backward analysis of the specific asset to identify at least one exposure risk to one or more assets that is in a downstream of the specific asset, wherein the at least one exposure risk includes an identification of an exposed asset, an entry point to the exposed asset, and a lateral movement risk associated with the exposed asset; and

[1.7] output a signal to cause on a display to present a presentation of forward and backward paths associated with the specific asset, thereby enabling visualization of a plurality of entry points and lateral movement risks associated with the plurality of entry points.

31. Furthermore, independent claim 14 is directed to a method, and independent claim 20 is directed to a computer-readable medium, but these claims essentially mirror claim 1. The dependent claims add other conventional aspects of cybersecurity and cloud computing, including identifying attack vectors reaching the asset via the Internet (claims 2 and 15), using a virtual machine, a network appliance, a storage appliance, a compute instance, or an engine instance as assets in a cloud

environment (claims 3 and 16), identifying assets based on an identity and access management policy, an organization policy, or an access policy (claims 4 and 17), presenting the forward and backward paths to indicate alternative paths connecting the asset and an upstream or downstream asset (claims 5 and 18), presenting the alternative paths in the visualization (claims 6 and 19), detecting a potential risk associated with the asset based on monitored network activities (claim 7), indicating in the presentation of the forward and backward paths port numbers for each pathway (claim 8), indicating an entry point at risk as part of an entry point visualization (claim 9), monitoring network activities of the cloud assets (claim 10), and detecting a potential risk associated with the asset based on a network activity of the asset, an upstream asset, or a downstream asset (claims 11-13).

B. Prosecution History of the '154 Patent

32. I understand that the '981 application (i.e., the patent application that led to the '154 patent) never received a rejection under §102 or §103. I understand that, in the first office action, the Examiner indicated that the claims were allowable without making a rejection. EX1004, 630-36. I also understand that, in allowing the claims, the Examiner identified a reference (Wang (CN 110708315)) as the closest prior art, but only broadly indicated the supposedly lacking teachings by copying and pasting most of the independent claim language without explanation. *Id.*, 322-23, 634-35.

IX. CLAIM CONSTRUCTION

33. I understand that, in this proceeding, the claims are construed according to their ordinary and customary meaning, in view of the specification and prosecution history, to a POSA as of the effective filing date of the underlying patent application, which I apply in my analysis below. I understand that this is the same claim construction approach used in district court litigation.

X. STATE OF THE ART

34. By April 26, 2021, cybersecurity systems that calculated and provided visualizations of the network paths that an attacker could take through assets within a computing environment were well-known and predictable. These attack vector visualizations were commonly known in the art as “attack graphs” or “attack trees.” *See, e.g.*, EX1007 (Engelberg), ¶¶5 (“In an effort to defend against cyber-attacks, so-called analytical attack graphs (AAGs) can be generated, which represent potential lateral movements of adversaries within computer networks.”), 26-29, Fig. 3 (showing example process-aware AAG); EX1009 (NIST Attack Graphs),¹ iv

¹ EX1009 is a copy of an August 2011 NIST publication obtained from <https://doi.org/10.6028/NIST.IR.7788>.

(“Attack likelihoods are propagated through the attack graph, yielding a novel way to measure the security risk of enterprise systems.”); EX1010 (Zeng), 1-2 (“[T]he attack graph plays a crucial role in network security analysis.”); EX1011 (Lallie), 2 (noting “attack graphs and attack trees are the most popular method of representing cyber-attacks”). Attack vector visualizations were aimed at showing network administrators how an attacker could leverage vulnerabilities associated with assets in a network environment in order to gain entry into the network and move among assets to achieve the attacker’s goal (e.g., obtaining access to sensitive information or compromising network services). *See, e.g.*, EX1007 (Engelberg), ¶¶5 (“An AAG can be used to understand how a computer network can be hacked and undesirable consequences that can result.”), 26-29; EX1008 (Bruttomesso), 1:25-29, 1:44-58 (stating “an attack graph ... enumerates the set of all possible sequences in which atomic-level vulnerabilities can be exploited to compromise system security”); EX1009 (NIST Attack Graphs), iv (“To more accurately assess the security of enterprise systems, one must understand how vulnerabilities can be combined and exploited to stage an attack.”), 6 (noting attacks launch complex multistep cyberattacks); EX1011 (Lallie), 2. Because POSAs understood that malicious attackers often targeted operationally sensitive and/or critical assets, it was appreciated that attack paths were a useful tool to understanding how important

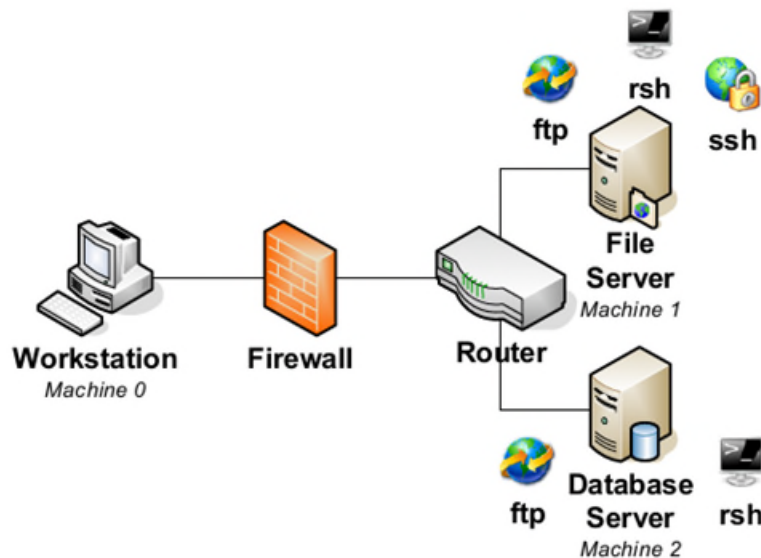
assets could be at risk. *See, e.g.*, EX1007 (Engelberg), ¶¶4-5 (recognizing that, “[d]ue to the potential severe damage and cost experienced by a victim,” critical infrastructure networks “have been intentionally targeted and have suffered from significant losses when successfully exploited”); EX1009 (NIST Attack Graphs), 6 (“Cyber attackers can launch multistep and multi-host attacks that can incrementally penetrate the network with the goal of eventually compromising critical systems.”); EX1010 (Zeng), Fig. 3 (illustrating multi-object attack path aimed at accessing protected internal servers).

35. A POSA understood that as a starting point to analyzing how an attacker could gain access into a network and move among its assets was knowledge of the overall network topology of the computing environment. *See, e.g.*, EX1006 (Morgan), 2:1-5, 7:9-19 (risk management system collects “knowledge of network architecture (e.g., device data, topology data, inter-device connection data)”), Fig. 2; EX1007 (Engelberg), ¶35 (security platform determines “network topology” of enterprise); EX1008 (Bruttomesso), Abstract (generation of attack paths includes “defining the topology of the network”), 2:4-30; EX1010 (Zeng), 2 (teaching collection of “information about network topology” for attack graph analysis, which is “designed to represent abstracted network topology with a directed acyclic graph”). This included, for example, knowledge of the assets contained within the

environment and the network paths by which they communicated or were reachable with one another. *See, e.g.*, EX1006 (Morgan), 7:9-19 (risk management system collects “knowledge of network architecture (e.g., device data, topology data, inter-device connection data)”); EX1009 (NIST Attack Graphs), 8 (computing network reachability); EX1010 (Zeng), 2 (collection of network topology, network configuration, and network connectivity), Fig. 2 (illustrating network topology of environment from which an attack graph is generated). A POSA understood that this information was relevant to informing an asset’s exposure to outside sources, such as sources originating from the internet, which could be informative of risk, especially for cloud-based assets. *See* EX1005 (Keren), 12:62-13:5 (using network topology and connectivity information to assess whether sensitive assets “are currently exposed to external networks”), 19:34-20:12 (analyzing database’s reachability with respect to internet-originating sources), Fig. 5; EX1017 (Lichtenstein), 6:7-23 (stating cloud-based “workloads run a risk of being exposed to the external networks, such as the Internet,” and “[t]his risk is inherent, for example, to being provisioned part of a cloud infrastructure, such as GCP, as a network connection is needed, by definition, to access the resources which are provisioned to any particular organization”).

36. However, the POSA was also aware that assessing the risk to assets contained in a network environment could be further improved by exploring the exact network paths that a malicious attacker could use to gain access to or control over assets in the network. That is, POSAs understood that the existence of a network path by which assets could communicate did not necessarily mean that an outside attacker could gain entry into an asset along the given network path. *See* EX1006 (Morgan), 3:4-19 (“selectively identifying network pathways” that “correspond to associated likelihoods that a given edge will be used by an attacker to potentially gain access to a sensitive asset”); EX1007 (Engelberg), ¶28 (noting “incomplete solutions for addressing real attack scenarios,” such as those that “only consider the topological connection between stepping stones to measure the difficulty of reaching a target”). Instead, POSAs understood that attackers would also need to exploit vulnerabilities present in some or all of the assets along the network path to accomplish this goal. *See, e.g.,* EX1006 (Morgan), 2:1-7 (additionally collecting “a plurality of vulnerability instances for the computing network”), 3:4-19, 7:20-27; EX1007 (Engelberg), ¶36 (identifying “specific vulnerabilities and improper configurations” for attack path analysis); EX1008 (Bruttomesso), Abstract (“identifying vulnerabilities of the topology”); EX1010 (Zeng), 2 (attack graph framework includes collection of vulnerability information).

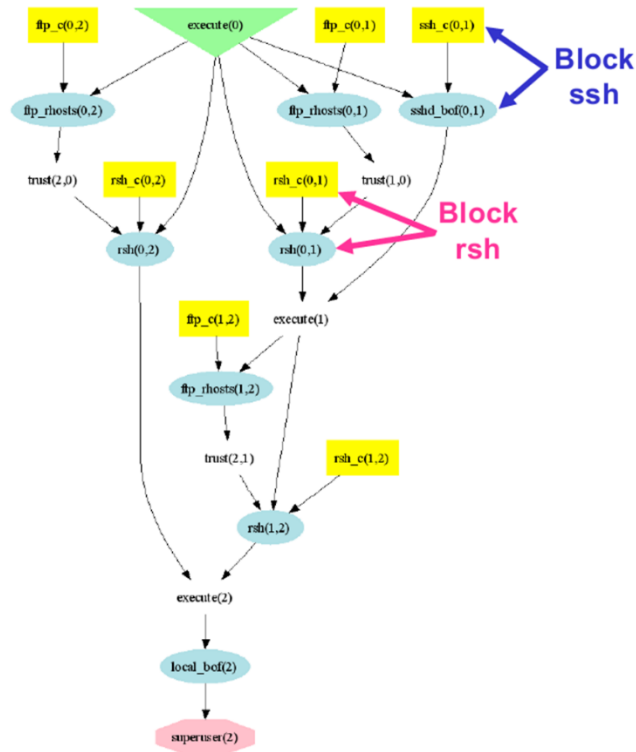
37. To illustrate this, consider the example network topology below in which traffic flows between a workstation (Machine 0), a file server (Machine 1, which services ftp, rsh, and ssh traffic), and a database server (Machine 2, which services ftp and rsh traffic), making the objects communicable with each other along the indicated paths:



EX1009 (NIST Attack Graphs), 7, Fig. 1.

38. To determine the likelihood that the network paths between the workstation, the file server, and the database server could be traversed by an attacker, POSAs knew to also analyze the vulnerabilities present on these devices. See EX1010 (Zeng), 2 (attack analysis requires collection of “information about network topology, vulnerability, network configuration, and network connectivity”). For instance, in one possible path of attack, an attacker could use the workstation to

execute a buffer overflow exploit against the file server's ssh service, allowing the attacker to execute arbitrary code. EX1009 (NIST Attack Graphs), 8. This would enable the attacker to exploit a particular ftp vulnerability that allows the attacker to anonymously upload a list of trusted hosts from the file server to the database server, which gives the attacker the required trust needed to remotely execute shell commands on the database server. *Id.* Using this acquired capability, the attacker can then gain control over the database server as a user with regular privileges, making a local buffer overflow exploit possible, which elevates the attacker to full privileges. *Id.* The end result is that the attacker now has full control and can execute code on the database server for malicious purposes. *Id.* This path can be visualized, shown below as the rightmost vector (labeled "Block ssh"), along with other alternative paths that an attacker could exploit to achieve the same goal (i.e., control of the database server):



EX1009 (NIST Attack Graphs), Fig. 1; *see also id.*, 13-20 (describing additional attack vector visualizations for various network topologies).

39. POSAs appreciated that an attack path analysis could be performed relative to any point within or outside the computing environment. For example, while the above example considered possible attack vectors relative to a malicious attacker operating from a workstation, it was well-known that attack path analysis also considered attacks originating from outside the computing environment, such as the internet. This is illustrated by the network topology shown in the top figure below and its corresponding attack graph (bottom figure). In this example, an

internet-based attacker is able to leverage vulnerabilities (as indicated by a Common Vulnerabilities and Exposures (“CVE”) identifier) existing in the network’s internet-facing assets (e.g., a web server located in the DMZ subnetwork) to gain access into internally protected assets (e.g., a SQL database server located in the trusted subnetwork). EX1010 (Zeng), 2-3.

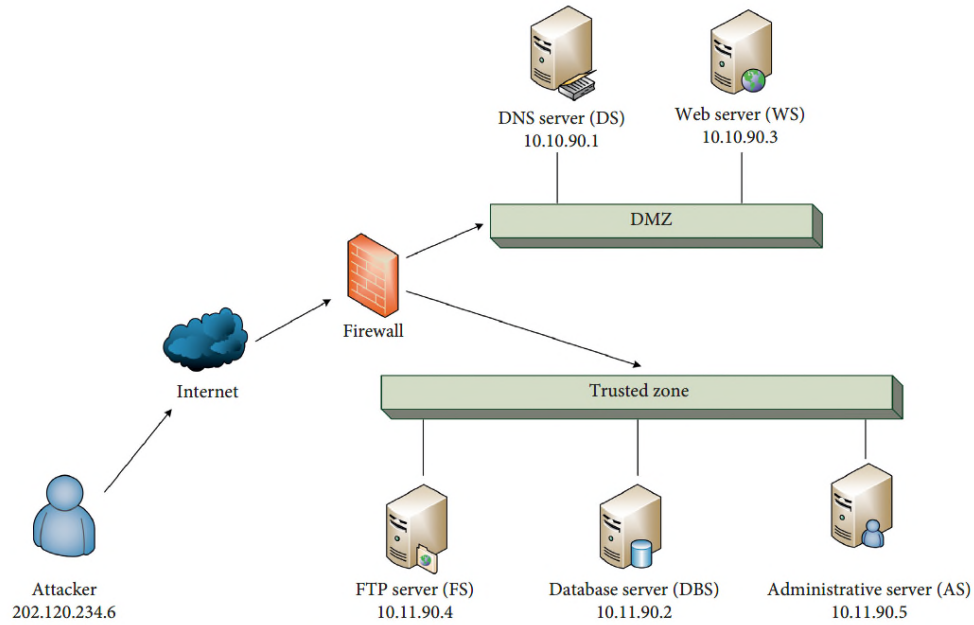


FIGURE 2: Example of network topology.

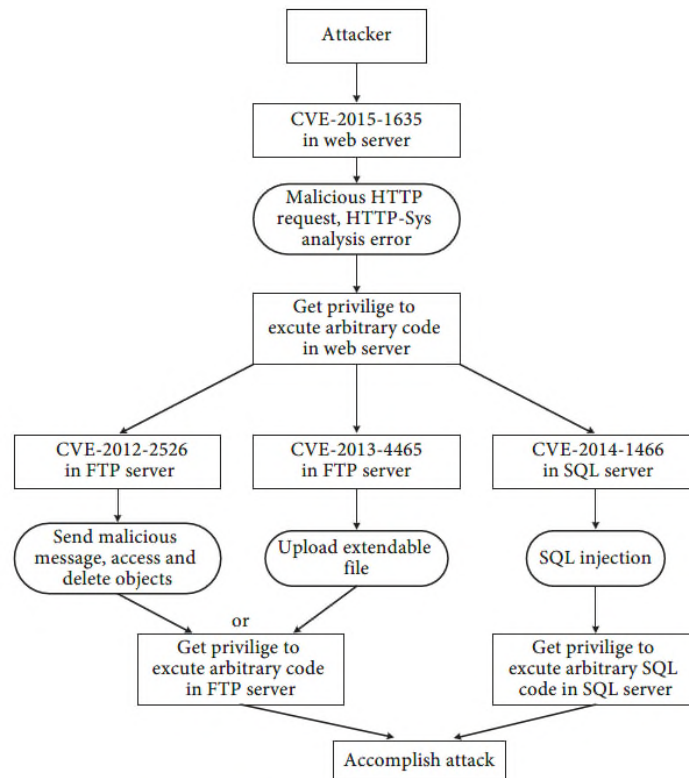


FIGURE 3: The corresponding attack graph.

EX1010 (Zeng), Figs. 2-3.

40. Thus, by also identifying and analyzing the vulnerabilities present along the reachable pathways between assets in a network, a POSA understood that assessments regarding the risk of attack could be beneficially improved. It was appreciated that this, in turn, would aid network administrators in the decision-making process for remediation efforts. *See, e.g.*, EX1006 (Morgan), 14:46-15:2, 15:50-57 (attack graphs may be “used for evaluation, e.g., determine the risk or analyze a threat and generally prioritize vulnerabilities and remediations”), 19:4-9; EX1007 (Engelberg), ¶29 (attack graphs “can be used to identify the most vulnerable components within a computer network, and can be used to evaluate fixes of vulnerabilities that the AAG reveals (e.g., by fixing a limited number of issues, any adversary attack on the computer, or on certain components in the computer network can be stopped)”); EX1010 (Zeng), 2 (stating “[o]ne of the main application scenarios” of attack graphs “is for network vulnerability analysis”). For example, by visualizing the possible attack paths to and from assets, possible impacts to the network could be better ascertained and the network administrator can gain greater insight into which fixes will result in a greater reduction of overall risk to the network. *See* EX1009 (NIST Attack Graphs), 20-22 (discussing example remediation actions, such as patching software vulnerabilities or removing network access, to reduce or eliminate risk of attack); EX1007 (Engelberg), ¶29; EX1017

(Lichtenstein), 6:20-23 (“It is therefore beneficial to be able to detect vulnerabilities and understand impacts of exposure as quickly as possible in order to address them.”).

XI. OVERVIEW OF THE PRIOR ART

A. Keren (U.S. Patent No. 11,374,982, EX1005)

41. Keren is a patent that relates to cybersecurity methods and systems “for automated generation of network analysis insights” in a cloud environment. EX1005 (Keren), 1:16-18; *see also id.*, Abstract, 3:6-39, 4:31-55, Figs. 1A-1B. The cloud environment includes “multiple security objects,” which are “objects, systems, devices, components, applications, entities, and the like, configured to operate within the cloud platform 104 and provide various functionalities therein.” *Id.*, 4:35-43, 5:20-44. The security objects may include, for example, “network objects” that are “configured to provide one or more traffic-management functionalities,” such as “proxies, firewalls, routing tables, and the like.” *Id.*, 5:32-39; *see also id.*, 6:13-19, 9:6-15. The security objects may also be “computing objects,” which are “configured to provide one or more data management or processing functionalities,” such as “dedicated processing systems, file servers, databases, and the like.” *Id.*, 5:32-44; *see also id.*, 6:13-19 (security objects may further include other computing resources such as virtual machines and containers).

42. The cloud environment is configured to connect with a cybersecurity system 150. *Id.*, 4:43-47, Figs. 1A-1B. “The cyber-security system 150 is a system, device, or component, configured to provide one or more network analysis functionalities including, without limitation, network analysis, traffic analysis, object querying, graph generation, and the like, as well as any combination thereof.” *Id.*, 6:47-51. Keren teaches that an aim of its system is “to provide network information which describes the role of an individual entity within such a network or a cloud environment, as well as the vulnerabilities presented by the entities’ placements within the network configuration.” *Id.*, 2:43-53.

43. To perform its analysis, the cybersecurity system of Keren is configured to identify the security objects contained within the cloud environment and collect security object data. *Id.*, 9:26-55; *see also id.*, 9:56-66 (“Security object data may include data describing, as examples and without limitation, object types, object names or unique identifiers (IDs), object network addresses, object port configurations, object status, such as online, offline, busy, and available, object input signal configurations, object output signal configurations, object security or access configurations, object processing rules and the like, as well as any combination thereof.”), 10:11-14 (“collection of security object data at S210 may include collection of the identities of all objects included in a cloud platform by generation

and transmission of an API query”), Fig. 2. This collected data allows the system to identify “the various objects included in, and adjacent to,” the environment and determine the various network connections, or relationships, between the security objects. *Id.*, 10:27-33, 10:49-57.

44. By identifying the security objects and their relationships, Keren teaches that the system constructs a security graph of the cloud environment, which describes the cloud environment’s overall network topology. *Id.*, 7:18-24, 10:33-48, Figs. 4A (visual node and link graph), 4B (list of objects illustrating network route). Based on the system’s analysis of the network, the security graph can be populated with properties describing features of the cloud environment. For example, Keren teaches that the network paths in the cloud environment can be analyzed and a “reachability property” of each security object can be calculated, which “defines if and how an object on the generated security graph can be reached from an external or internal network,” such as the internet. *Id.*, Abstract, 11:61-12:10, 13:26-34, 15:15-24, Figs. 3A-3B. Additionally, “security insights may be generated,” which describe aspects of the network, including descriptions of risky network configurations, such as “description of network management or anomaly detection data,” “descriptions of network configurations or events which are rare or

novel,” “descriptions of connections which are unauthorized,” and “descriptions of connections which display anomalous behavior.” *Id.*, 12:11-14, 12:45-13:5.

45. Keren further teaches that the system may include a security graph utility that is configured to provide a visualization of the cloud environment and its features to a user. *Id.*, 16:41-53. The security graph utility can include a visual representation of the security graph generated by the system (shown below) where security objects may be represented by vertex visualizations and relationships between objects may be represented by edge visualizations. *Id.*, 16:54-67.

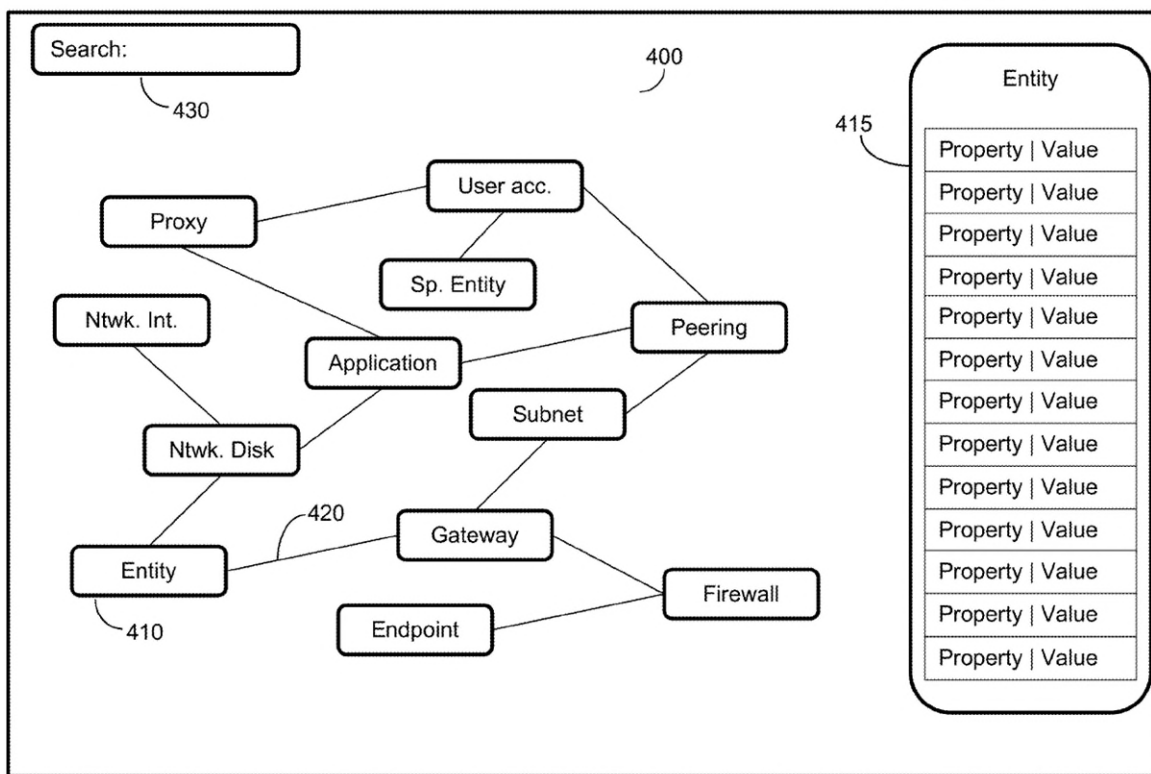


FIG. 4A

Id., Fig. 4A.

46. The security graph utility can also include interactive functionalities. *Id.*, 17:1-3. For example, the graphical interface may allow a user to click on a vertex visualization, which allows the user to view information specific to the security object represented by the vertex visualization. *Id.*, 17:3-18. The graphical user interface may also provide a “security graph object list,” which can list selected security objects in one or more orders, including based on “mode-specific factors.” *Id.*, 17:52-18:1. Keren explains that “[m]odes are selected list-organization profiles” that provide for the population of a security graph object list in one or more configurations, such as “routing paths, object utilization or availability descriptions, other, like, configurations, and any combination thereof.” *Id.*, 18:8-15. As one example, “a ‘route’ mode may be selected” in which a security graph object list “is populated to provide information describing a route between the specified objects.” *Id.*, 18:22-39, Fig. 4B.

B. Morgan (U.S. Patent No. 12,015,631, EX1006)

47. Morgan is a patent describing systems and methods for “identifying instances of vulnerabilities on a computing network and generating a graph representing pathways that an attacking entity may take with respect to accessing one or more sensitive assets.” EX1006 (Morgan), Abstract; *see also id.*, 1:54-67, 5:54-63.

48. Morgan explains that “[r]ecent years have seen significant growth in the use of computer networks” and “[t]hese computer networks may be used for a variety of purposes including communication between users, data storage, and any number of applications.” *Id.*, 1:6-12. However, “[a]s the size and complexity of computer networks continues to grow, increasing quantities of information, including sensitive information, can be at risk” of unauthorized entities gaining access to such information. *Id.*, 1:12-18.

49. To protect against this risk, Morgan describes a risk management system that “receive[s] network information associated with a computing network including asset data, network architecture data, access rights data, and application configuration data.” *Id.*, 2:1-5; *see also id.*, 7:9-19 (system collects “sensitive asset information (e.g., customer indications of sensitive assets), knowledge of network architecture (e.g., device data, topology data, inter-device connection data), knowledge of system access rights (e.g., node data, permissions), and knowledge of application configurations (e.g., policies, application configurations, node configurations, device configurations)”), 11:48-56. The risk management system also “receive[s] or otherwise identif[ies] a plurality of vulnerability instances for the computing network.” *Id.*, 2:5-7; *see also id.*, 7:20-27. Morgan explains these vulnerability instances include “any problem or issue that has been called out with

respect to a piece of software or a configuration (e.g., a device configuration, an account configuration, and application configuration),” such as “a common vulnerabilities and exposures record instance” or “CVE.” *Id.*, 3:56-4:8; *see also id.*, 11:66-12:12 (risk management system “receive[s] knowledge of discovered vulnerabilities,” such as “a listing of CVEs that are applicable to the computing network” and can include “out of date software, vulnerable application policies, and any other network information that represents a known vulnerability”).

50. After gathering relevant asset data, determining network architecture, and identifying applicable vulnerabilities, Morgan teaches that the risk management system can build a “threat graph” that includes “a plurality of pathways between an attacking entity (or multiple attacking entities) and one or more sensitive assets.” *Id.*, 2:7-15; *see also id.*, 14:46-60 (“The system uses a combination of knowledge of sensitive assets in a particular network, knowledge of the greater network security architecture, as well as knowledge of system access rights and application configurations, to generate a comprehensive threat model.”), 15:3-15, 15:50-57. The threat graph includes a plurality of nodes (representing, for example, “a level of access or privilege” to an asset, such as user account, group account, or administrative account) that are connected by edges, which represent “escalations” that can be used by an attacker and “may refer to a specific instance of a

vulnerability, the access granted explicitly by a system configuration at a specific location in the network, or the implicit access an entity would have due to the inherent design of a system at a specific location in the network.” *Id.*, 4:9-27, 4:36-49; *see also id.*, Abstract, 4:50-67, 6:11-21, 15:39-43, Figs. 3A-3C.

51. Morgan explains that risk management system identifies attack paths “based on identified vulnerabilities that correspond to associated likelihoods that a given edge will be used by an attacker to potentially gain access to a sensitive asset.” *Id.*, 3:7-12; *see also id.*, 7:28-44 (“Accordingly, the threat model or resulting graph may include a partial representation of nodes and edges that make up the total topology of the network architecture based on those nodes and edges being associated with corresponding vulnerabilities.”). For example, as one simplified example (reproduced below), the threat graph can indicate particular paths that an attacker (annotated by the red box) can traverse (e.g., the path annotated in red) within the network to gain access to a particular sensitive asset (e.g., first asset 308a, annotated by the yellow box, which can be, for example, a database containing sensitive information or a service that has a low tolerance for failure). *Id.*, 5:14-32, 12:44-13:43.

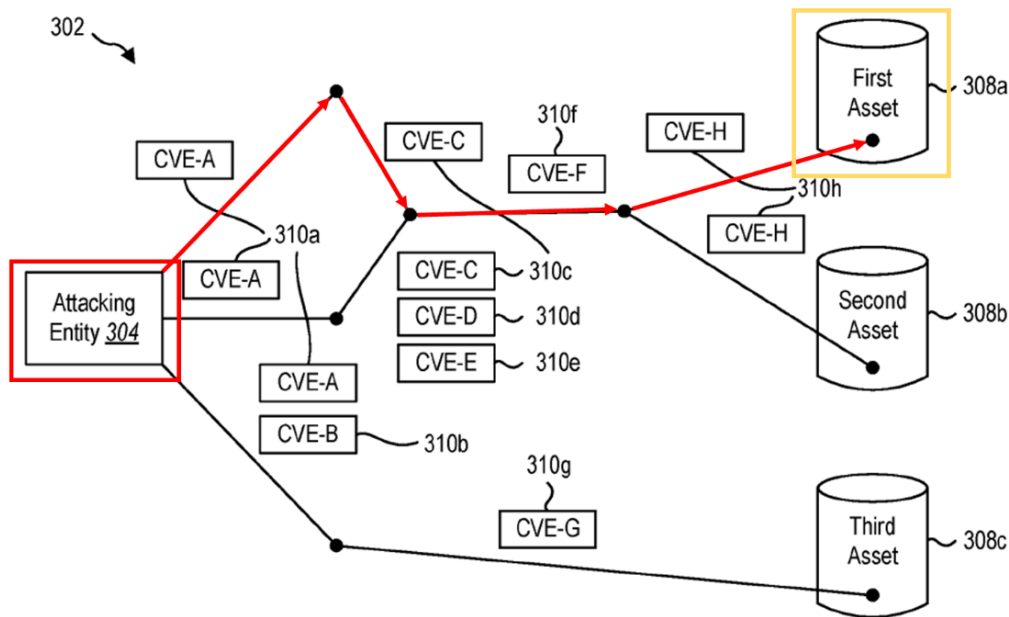


FIG. 3B

Id., Fig. 3B (annotated).

52. By successively exploiting a set of vulnerabilities present in hosts on the network (e.g., CVE-A to CVE-C to CVE-F to CVE-H, each represented by an edge), the attacker can gain increasing access (e.g., by increasing privileges or permissions, such as moving from non-privileged access to administrative privileges or acquiring read/write permissions, as represented by nodes) in an effort to gain entry into the target asset. *Id.*, 4:36-49, 13:8-31; *see also id.*, 13:66-14:17, 14:58-16:38, 16:57-66 (explaining vulnerability information may be “associated with any level of access (represented as a node) within th[e] host that could be compromised by exploiting” vulnerable software contained on the host), Fig. 4A.

53. Morgan teaches that the threat graph is presented to a user in a graphical user interface, which allows the user to view the paths that an attacker can use to move within the computing network. *Id.*, 14:18-35, Fig. 4B (reproduced below). As Morgan explains, this sort of visualization makes it easier for the user to understand where weaknesses exist in the current network configuration and, as a result, where to efficiently focus remediation efforts. *Id.*, 15:50-57, 19:15-34, 19:55-20:7; *see also id.*, 3:12-15 (“In this way, the risk management system significantly reduces the amount of information that a security professional would need to navigate in determining potential security risks to the computing network.”), 10:63-11:29 (contrasting conventional systems that only “provide a report listing known vulnerabilities” and are “limited to diagnosing vulnerabilities to a network based on likelihood of infiltration of the particular vulnerabilities”).

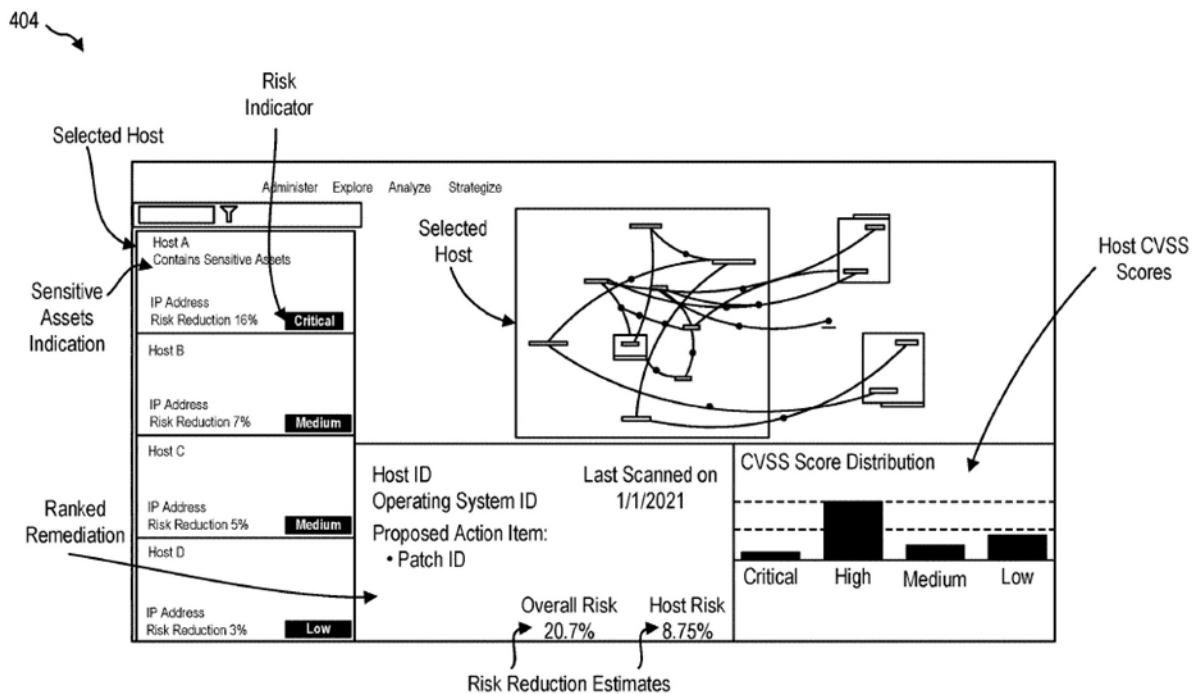


FIG. 4B

Id., Fig. 4B.

XII. GROUND 1: CLAIMS 1-20 WERE OBVIOUS OVER KEREN AND MORGAN

54. As I explain below, it is my opinion that claims 1-20 were obvious over Keren and Morgan. Keren describes a cybersecurity system for determining and analyzing the network topology of a cloud computing environment. *Supra*, §XI.A (Keren summary). Keren's system identifies security objects contained within the environment, such as virtual machines, network devices, databases, and the like, and identifies the network relationships between the objects. *Id.* Based on this information, Keren teaches that the system can analyze the network paths of the

environment to calculate the reachability of the security objects from outside the cloud environment (e.g., the internet) and generate security insights describing network configurations that may present risk to the environment. *Id.* This analysis is then presented to a user in a graphical interface that provides a graph illustrating the overall network topology of the cloud environment, which visually shows the network paths to and from security objects within the network. *Id.* The interface further includes interactive features where the user can select a security object to view object-specific properties of the selected security object, including visualizations that illustrate the network path for the selected object. *Id.*

55. Morgan discusses that analysis of risk present in the network topology of a computing environment can be further improved by analyzing the environment's network paths to identify the specific paths that an attacker can use by exploiting vulnerabilities present on hosts along the paths. *Supra*, §XI.B (Morgan summary). Morgan also teaches providing the resulting attack path analysis as a visualization to the user, which aids the user in prioritizing risk and remediation efforts for network administration. *Id.*

56. As I detail more below, in view of Morgan's teachings, a POSA would have had reason to include attack path visualizations with Keren's graphical user interface system such that, when a user selected a security object using interactive

features taught by Keren to view object-specific information, the system would analyze the network paths to and from the selected security object to identify the particular paths that could be exploited by an attacker and output those paths as attack vector visualizations. *Infra*, §XII.A (reasons to combine).

A. Reasons to Combine Keren and Morgan

57. In my opinion, a POSA would have had reason to combine the teachings of Keren and Morgan to arrive at claims 1-20. As detailed more below, a POSA would have appreciated that adding functionality to a system like Keren's that would output visualizations of the specific network paths for a security object that could be compromised by an attacker outside of the cloud environment would enhance the system's ability to identify security risks in the environment by providing greater insight into how threats from outside sources can enter a network and the extent to which those threats could impact assets within in the network. The POSA would have also recognized that such attack vector visualizations would improve network administration because, by visualizing the likely attack paths to and from a security object, the network administrator can gain a better understanding as to where remediation efforts can be efficiently focused. The POSA would have appreciated that such functionality could be readily integrated into Keren's interactive graphical user interface system by allowing the user to select a security

object to view the network paths to and from the selected security object that the system identifies as possible attack vectors. *See also infra*, §§XII.B.7-9.

58. As an initial matter, Keren and Morgan are analogous to one another and to the '154 patent. For example, Keren describes cybersecurity systems and methods for generating and outputting network graphs and corresponding path analysis that aid in the administration of security objects (i.e., assets) in a cloud environment. *See, e.g.*, EX1005 (Keren), Abstract, 1:16-18, 2:34-53 (describing systems and methods that provide network administrators “network information which describes the role of an individual entity within such a network or cloud environment, as well as the vulnerabilities presented by the entities’ placements within the network configuration”), 6:47-51 (cybersecurity system provides “one or more network analysis functionalities including, without limitation, network analysis, traffic analysis, object querying, graph generation, and the like”). Morgan teaches risk management systems and methods for generating graphs that provide visualizations of vulnerable network pathways that an attacking entity may exploit to reach assets contained within a computing environment, such as a cloud environment. EX1006 (Morgan), Abstract, 1:54-67 (“The present disclosure generally relates to a risk management system for identifying network vulnerabilities ... on a computing network (e.g., a ... cloud computing network) and determining

levels of risk associated with respective instances of the vulnerabilities....”). Keren and Morgan are therefore in the same field as the ’154 patent, and their teachings are pertinent to the issues addressed by the ’154 patent since those teachings involve cybersecurity systems and methods for security cloud infrastructure, such as performing visualization of forward and backward facing threats of assets contained in a cloud environment. *See, e.g.*, EX1001, 1:16-19, 45:36-57.

59. Moreover, a POSA would have had reason to include attack path visualizations, as taught by Morgan, in a graphical user interface system like Keren’s to help improve the security posture of the cloud environment managed by Keren’s system by more accurately identifying the risks of an outside attacker entering into a network and moving along assets within a network path. As a general matter, Keren recognizes that “[a]s businesses, governments, and other organizations expand and increase their digital presence through various computer, network, and web technologies, the same parties may be increasingly vulnerable to developing cyber-threats.” EX1005 (Keren), 1:22-29. Keren thus teaches that there is a need for tools that allow for effective management of assets contained within an organization’s network and further teaches a desirability that such tools “provide granular, comprehensible representations of network information.” *Id.*, 1:36-49, 2:34-42. To meet this need, Keren describes a cybersecurity system that “provide[s]

network information which describes the role of an individual entity within such a network or a cloud environment, as well as the vulnerabilities presented by the entities' placements within the network configuration.” *Id.*, 2:42-55.

60. The system taught by Keren identifies the assets (or “security objects”) contained within the cloud environment and their relationships and generates security graphs describing the overall network topology of the cloud environment. *Id.*, 9:22-27, 10:27-30, 10:49-57, Figs. 2, 4A. Using this information, the system analyzes the network paths to and from security objects to calculate certain object-specific properties. *See, e.g., id.*, 11:61-12:12, 13:47-53, 14:31-33, 15:15-24, Figs. 3A-3B, 5. For example, one such property is an asset’s “reachability property,” which describes the communication paths by which the asset can be reached from a given source, such as the internet. *Id.*, 12:4-10, 13:26-61, 19:66-20:12, Figs. 3A-3B. From these calculated properties, Keren teaches that the system may generate security insights regarding the current state of the cloud environment to aid in effective network administration. *Id.*, 2:34-55 (system provides “granular, comprehensive representation of security object properties and insights”), 12:11-13:5.

61. As I discussed above in §X, a POSA would have appreciated that a security object’s reachability along a network path can inform a network

administrator the extent to which the security object was exposed to outside sources, such as sources originating from the internet, which could be indicative of risk. See, e.g., EX1005 (Keren), 12:62-13:5 (generating security insights notifying user of assets that “are currently exposed to external networks”), 19:34-20:12 (analyzing database’s reachability with respect to internet-originating sources), Fig. 5. However, the POSA would have also appreciated that the existence of a reachable path did not necessarily mean that an outside attacker could actually use the path to gain entry into security objects along that path. *Supra*, §X; see also EX1006 (Morgan), 3:4-19 (recognizing not all network pathways can be used by an attacker); EX1007 (Engelberg), ¶28 (noting “topological connection between stepping stones to measure the difficulty of reaching a target” provides limited analysis as to attack risk).

62. Morgan teaches that assessments regarding the risk to a network can be beneficially improved by further identifying and providing as visualizations the specific pathways between assets that can be traversed by an attacker. Like Keren, Morgan teaches that “[a]s the size and complexity of computer networks continue[] to grow, increasing quantities of information, including sensitive information, can be at risk.” EX1006 (Morgan), 1:12-14; see also *id.*, 1:14-18 (“This becomes increasingly apparent as the number of security threats continues to grow with

respect to unauthorized entities gaining access to information maintained on computer networks.”). However, to improve the security of a network to effectively mitigate against these risks, Morgan teaches that “[u]nderstanding where and how to implement these security improvements ... can become complicated, particularly where a computer network may have hundreds of thousands of vulnerable points that need to be addressed.” *Id.*, 2:26-29. Morgan explains that merely identifying vulnerabilities present in a network can be limited because “not all known vulnerabilities are necessarily important to fix or provide a significant boost to security of a network system.” *Id.*, 2:35-53. Thus, to improve risk evaluation, Morgan teaches analyzing the network paths within a computing environment, and further identifying the specific paths that, based on identified vulnerabilities, correspond to pathways that an attacker could use to potentially gain access to assets within the network. *Id.*, 1:60-67, 3:7-12, 12:44-63, Figs. 3A-3B; *see also id.*, 2:1-7, 7:9-27, 15:3-49. A POSA would have recognized that this provides for an improved assessment of risk to a computing environment because the analysis allows the network administrator to view the cumulative effects of vulnerabilities along a network path. *Id.*, 10:63-11:29 (stating identification of attack paths “represent material improvements to conventional network security systems that are often limited to diagnosing vulnerabilities to a network based on likelihood of infiltration

of the particular vulnerabilities”), 15:54-57 (“The comprehensive threat model may in turn be used for evaluation, e.g., determine the risk or analyze a threat and generally prioritize vulnerabilities and remediations.”); *see also* EX1009 (NIST Attack Graphs), iv (attack path analysis “is used to maximize the security of enterprise systems” and “can be used to evaluate and strengthen the overall security of enterprise networks”).

63. More specifically, a POSA would have appreciated that outputting attack vector visualizations in a system like Keren’s would provide network administrators with additional insight into how an attacker could gain access into the cloud environment through a given network path and the overall impacts to security objects, including protected sensitive assets, along the path. Keren expresses a desire to inform administrators of potential risk to assets, including risks that “may not be detectable based on the analysis of individual objects,” such as risky exposure of critical assets to external networks. EX1005 (Keren), 12:62-13:5 (describing example insights as “third-party networks A, B, and C currently have access to the internal network,” “objects E, a database, and F, an administrator interface, are currently exposed to external networks,” and “cross-environment exposure has been detected between the development and production environments”); *see also id.*, 1:22-35 (recognizing “attackers may seek to identify and exploit [vulnerabilities] to

gain access to sensitive systems and data”), 2:49-53. Moreover, Keren teaches that the properties of a security object can affect how reachable a downstream object is from an outside network. *Id.*, 15:25-33 (“[A] reachable property is determined as a minimal set of reachable properties of all other objects in the path.”), 18:59-19:18 (reachable properties are determined based on “the traffic permitted to pass through each object-to-object step of the path”), 19:34-20:12, Fig. 5 (illustrating that network properties of a public proxy limit traffic forwarded to a downstream database in a network path). Thus, Keren teaches a desire to provide analysis and insights that comprehensively consider the properties of objects within a given network path.

64. Morgan teaches that attack vector visualizations enable the network administrator to understand how an attacker can both enter a network and move within it to gain entry into target assets, such as sensitive assets. EX1006 (Morgan), Abstract (“The systems described herein may use graph theory to generate or otherwise identify pathways that an attacker is likely to use in accessing the sensitive assets(s).”), 3:23-25, 5:14-32 (analysis considers impact on sensitive network assets), 7:40-44, 8:40-56, 13:66-14:17 (Fig. 4A illustrating multiple host traversal between “a point of origin (e.g., a point of entry to the network or an attacking entity) and a sensitive asset”), 20:3-7 (contemplating visualization of attack path relative to a selected host), 20:23-62 (process analyzing “an entry point ... of the customer

security architecture, e.g., an adversary, and an endpoint, e.g., a sensitive asset,” to establish “possible attack vectors”), Figs. 4A-4B; *see also* EX1009 (NIST Attack Graphs), 6 (“Cyber attackers can launch multistep and multi-host attacks that can incrementally penetrate the network with the goal of eventually compromising critical systems.”); EX1010 (Zeng), Fig. 3 (illustrating multi-object attack path aimed at accessing protected internal servers). Thus, a POSA would have recognized that providing attack vector visualizations in a graphical interface system like Keren’s would further Keren’s stated goal of providing “granular, comprehensive representations of network information” because such visualizations would improve the identification of specific network paths, and the assets within those paths, that are at risk to outside attack. EX1005 (Keren), 2:34-55 (further stating desire to “describe[] the role of an individual entity within such a network or a cloud environment, as well as the vulnerabilities presented by the entities’ placements within the network configuration”).

65. In addition, a POSA would have recognized that providing attack vector visualizations would enable better network administration by improving the network administrator’s decision-making process regarding remediation activities. Morgan teaches that by visualizing possible attack paths and their overall impacts to assets within the path (e.g., the impact to operationally sensitive and/or critical assets), the

network administrator can use this information to make more informed decisions on where to focus remediation efforts. EX1006 (Morgan), 14:61-15:2 (attack path analysis enables identification of “actual, business-specific threats,” which “facilitates buy-in from decision makers that choose and implement remediations”), 15:54-57 (“The comprehensive threat model may in turn be used for evaluation, e.g., determine the risk or analyze a threat and generally prioritize vulnerabilities and remediations.”), 19:4-14; *see also id.*, 13:55-65 (providing example of remediation that reduces overall risk); EX1007 (Engelberg), ¶29 (attack graphs “can be used to evaluate fixes of vulnerabilities that the AAG reveals (e.g., by fixing a limited number of issues, any adversary attack on the computer network, or on certain components in the computer network can be stopped)”); EX1009 (NIST Attack Graphs), 20-22 (providing examples illustrating how attack graph metrics enable “system administrators to decide among the different risk mitigation options”). For example, POSAs would have appreciated that knowledge of how an attacker could access a given security object and the downstream consequences to other security objects that could occur from such access would provide valuable insight into which fixes would result in better protection for network assets (e.g., by cutting off multiple pathways that could be used for attack). *See, e.g.*, EX1006 (Morgan), 13:55-65, 18:33-48 (teaching remediation of vulnerability based on its association with

multiple pathways), 19:4-19 (deprioritizing vulnerability fix because it “is not located on a viable attack path leading to a sensitive asset”); EX1009 (NIST Attack Graphs), 20-21 (consideration of patches for a given asset and their effect on the risk of compromise for other assets within the same network path).

66. Thus, in view of the above, a POSA would have had reason to configure Keren’s system that would allow the user to select a security object to view object-specific information that would visualize to the user the object’s network paths that the system identifies as attack vectors so that the user can visualize how threats from outside sources can access the security object and the extent to which those threats could impact other objects within its network paths and, in turn, visualize where remediation efforts can be efficiently implemented.

67. Furthermore, as I explained above, calculating and providing visualizations of attack vectors, as taught by Morgan, was well-known and predictable. *Supra*, §X. In addition, POSAs were familiar with the various modeling techniques used to analyze and identify possible attack vectors in a network and understood how to visualize such paths to a network administrator. *See, e.g.*, EX1006 (Morgan), Abstract (systems “use graph theory to generate or otherwise identify pathways that an attacker is likely to use”), 8:27-30; EX1007 (Engelberg), ¶73 (“The data object underlying an AAG can be processed to generate a visual

representation, a graph, of the AAG.”); EX1009 (NIST Attack Graphs), 8-9 (discussing known tools for generating attack graphs, including those that “compute network reachability and attack graphs representing potential paths for adversaries exploiting known vulnerabilities,” allowing for the discovery of “all hosts that can be compromised by an attacker starting from one or more locations”); EX1010 (Zeng) 4-10 (providing overview of known attack graph analysis methods); EX1011 (Lallie), 2-11 (describing “[a]ttack modelling techniques (AMT)” and visualization of attack graphs and trees). Thus, in addition to the reasons I identified above, the combination of Keren and Morgan would have merely involved using the familiar technique of analyzing network paths in a cloud environment to identify attack vectors and presenting the analysis to a user as visualizations in a graphical user interface.

68. Finally, a POSA would have had a reasonable expectation of success in combining the above teachings of Keren and Morgan to arrive at the claimed invention. For instance, a POSA would have recognized that Keren already provides the foundational framework—i.e., identification of the overall network architecture of the cloud environment, analysis of security objects’ network paths and reachability, and an interactive graphical user interface—that would be needed to further analyze network path information to identify and visualize to a user likely

attack paths for a specific security object. EX1006 (Morgan), 14:46-60 (“knowledge of the greater network security architecture” permits attack path analysis), 15:2-15 (teaching attack model “is based on a particular network environment, including the physical and logical network security architecture”), Fig. 2; *see also* EX1007 (Engelberg), ¶35 (first determining network topology to provide attack graph analysis); EX1009 (NIST Attack Graphs), 8 (known methods of attack graph generation rely on computation of “network reachability”); EX1010 (Zeng), 2 (stating “information about network topology, vulnerability, network configuration, and network connectivity should be firstly collected” and explaining attack graph generation generally begins with “reachability analysis”), Fig. 2 (illustrating example network topology used for attack path analysis). As demonstrated by Morgan, POSAs were also aware of additional data (e.g., access rights data, application configuration data, and vulnerability data) that would facilitate attack path analysis and were familiar with the routine methods for collecting such data. EX1006 (Morgan), 7:9-27, 12:3-7 (vulnerabilities “are discovered by applying a vulnerability scanne[r] (e.g., a conventional or off-the-shelf vulnerability scanner”), 14:46-60; *see also* EX1010 (Zeng), 2-3 (teaching collection of “vulnerability” information, such as CVEs). Moreover, as I explained above, POSAs were also familiar with the known methods for analyzing network attack paths and visually

presenting the identified paths to a user. *See, e.g.*, EX1006 (Morgan), Abstract, 8:27-30; EX1007 (Engelberg), ¶73; EX1009 (NIST Attack Graphs), 8-9; EX1010 (Zeng), 4-10; EX1011 (Lallie), 2-11; *supra*, §X. Thus, the combination of Keren and Morgan would have merely involved using the familiar technique of additionally performing attack vector analysis for a selected security object and visualizing the attack vectors to a user in a graphical user interface. Therefore, in my opinion, combining the references’ teachings as discussed above would not have presented meaningful technical challenges or produced unexpected results.

B. Independent Claims 1, 14, and 20

69. Claims 1, 14, and 20 recite a system, method, and computer-readable medium, respectively, while reciting analogous limitations, which are discussed together below.

1. Preambles

70. As I explain below, the combination of Keren and Morgan teaches the preambles of claims 1, 14, and 20. Claim 1 recites a “graphical user interface system for providing forward and backward facing attack vector visualizations, the system comprising.” Claim 14 recites a “method for generating a graphical user interface for providing forward and backward facing attack vector visualizations, the method comprising.” The preamble of claim 20 recites a “non-transitory computer-readable

medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations comprising.”

71. Keren teaches a graphical user interface system and corresponding methods for generating a graphical user interface in the form of a cybersecurity system, which “is a system, device, or component, configured to provide one or more network analysis functionalities including, without limitation, network analysis, traffic analysis, object querying, graph generation, and the like, as well as any combination thereof.” EX1005 (Keren), 6:47-56. As I detail below, one particular component of the cybersecurity system is a “security graph utility” that is configured to provide forward and backward facing vector visualizations of a cloud network environment to a user.

72. More specifically, Keren teaches that the cybersecurity system assesses “security objects” (e.g., virtual machines, databases, firewalls, load balancers, proxy servers, and the like) included within a cloud environment. EX1005 (Keren), Abstract; *see also id.*, 1:15-18 (“The present disclosure relates generally to network administration....”), 3:6-39, 4:31-43 (cloud environment includes cloud computing platforms having “multiple security objects”), Figs. 1A-1B, claims 1, 11. The described system and corresponding methods assess the security objects and their properties, automatically generate security graphs describing the objects contained

in the cloud environment and their relationships, and provide network analysis insights of the cloud environment. *See id.*, Abstract, 1:15-18, 2:24-55 (“network administrators, and the like, may seek a solution which allows for network analysis and includes granular, comprehensive representation of security object properties and insights”), 4:53-55, 6:47-56.

73. Keren teaches that the analyses performed by the system are visually outputted to a user. For example, Keren teaches that the cybersecurity system is configured to generate a “security graph visualization” that acts “as a visual representation of a network and is presented through a security graph utility.” *Id.*, 16:41-45, Fig. 4A. The “security graph utility may be an application, interface, or other, like, means of providing a visual representation of a security graph, and the like, where the provided security graph visualization ... may include various interactive features.” *Id.*, 16:45-49; *see also id.*, 16:49-53 (“A security graph utility may be configured as, as examples and without limitation, a web interface, an application or executable installed on a user or administrator device, other, like, configurations, and any combination thereof.”), 21:59-62. An example of a security graph utility is reproduced below:

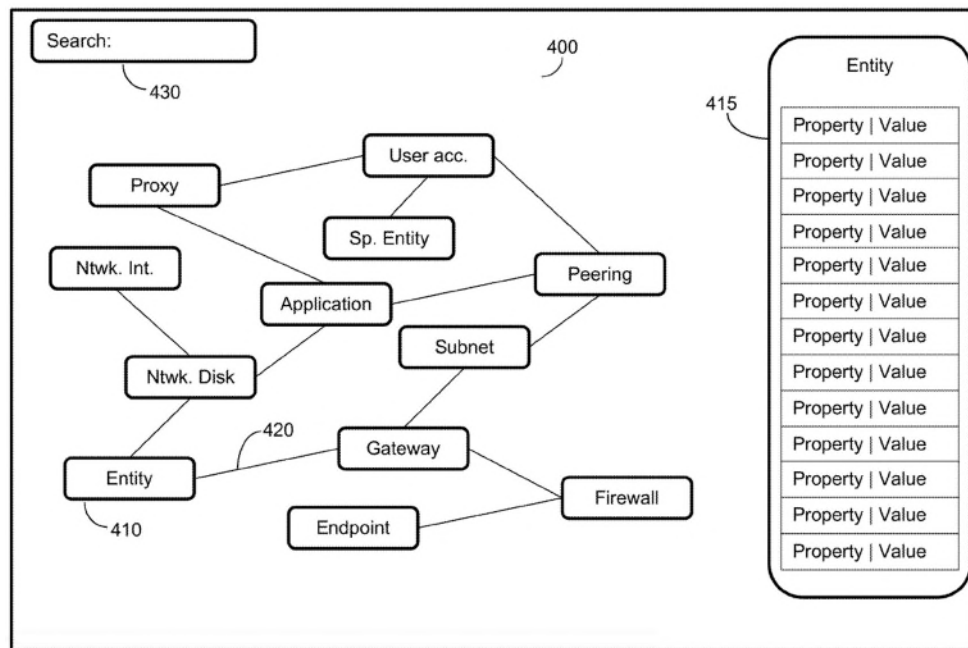


FIG. 4A

EX1005 (Keren), Fig. 4A.

74. As shown above, the security graph utility of the cybersecurity system includes a security graph visualization 400, which “is a security graph visualization representing a network..., wherein the various objects, systems, devices, components, and the like, of the network are represented as vertex visualizations 410,” which, in turn, “are variously interconnected by edge visualizations 420, representing connections between the various objects, systems, devices, components, and the like.” *Id.*, 16:54-62. A POSA would have understood that the network paths between security objects (such as illustrated above in Fig. 4A by the edges connecting the vertexes) were vector visualizations of the possible

communication paths within the cloud environment. *Compare* EX1001, 12:62-13:2 (describing vectors as “connecting the plurality of systems and devices” in a cloud infrastructure “based on the networking configuration”). These vector visualizations would have been understood to include forward and backward facing visualizations because they visualize the network paths going to a given security object and from the object to other downstream security objects within the path. *See infra*, §§XII.B.7-8.

75. Keren further teaches that the security graph utility can provide additional visualizations of specific network properties and network paths in various interactive features. *Id.*, 17:1-3. For example, Keren explains that the security graph utility may allow a user to click on a vertex visualization 410 within the security graph visualization 400 to display properties specific to the security object represented by the vertex visualization (through, for example, a vertex overview pane 415). *Id.*, 17:3-18. Keren also teaches that the security graph utility may include a “security graph object list 450,” which is configured “to list security objects 470 in one or more orders based on factors including, without limitation, object names, object types, object connection latencies, mode-specific factors, other, like factors, and any combination thereof.” *Id.*, 17:39-18:1. When the security graph object list operates based on mode-specific factors, Keren explains that the list

provides information relating to selected security objects according to a specific mode selected by the user, such as “routing paths, object utilization, or availability descriptions.” *Id.*, 18:1-22 (“[W]here a mode is selected, the mode-specific data display 465 may be configured to display information regarding a specific security graph object list 350 populated based on the specified mode or modes.”).

76. One example is shown below, where “a ‘route’ mode” has been selected, which populates the security graph object list 450 “with objects occupying a data path between a first object 470-1” (a container) “and a destination object 470-n” (the internet). *Id.*, 18:23-27. The object list contains a display of “9 hops,” which “indicat[es] that a transmission from a first object 470-1 makes nine ‘hops,’ or transmissions between objects, before reaching the destination object 470-n.” *Id.*, 18:27-34. A POSA would have understood that this example object list also showed forward and backward facing vector visualizations because it visualizes the network path to a specific security object and other downstream objects (e.g., proxies, network interfaces, container) from a destination or source (e.g., the internet). *Compare* EX1001, 12:62-13:2.

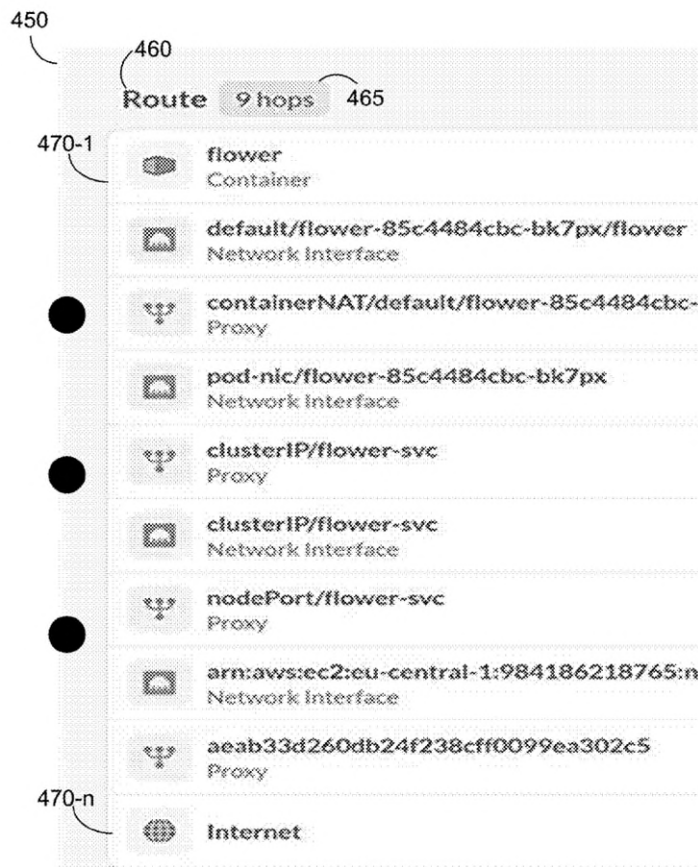


FIG. 4B

EX1005 (Keren), Fig. 4B.

77. While Keren teaches a graphical user interface system that provides visualizations of the current network configuration of a cloud environment, including forward and backward facing vector visualizations showing the network paths to and from specific security objects, Keren does not expressly disclose that the visualizations are “attack” vector visualizations, as recited. However, as I discussed above, a POSA would have had reason to include attack vector

visualizations in a graphical user interface system, as taught by Keren, that would allow the user to view the specific network paths that the system identifies as attack vectors with respect to a specific security object in view of Morgan’s teachings. *See supra*, §XII.A. Specifically, as I explain in more detail below (*infra*, §§XII.B.7-8), Morgan teaches analyzing objects contained within a computing environment to identify the network paths that an outside attacker would likely use to access assets contained in the computing environment. EX1006 (Morgan), Abstract (“The systems described herein may use graph theory to generate or otherwise identify pathways that an attacker is likely to use in accessing the sensitive assets(s).”), 5:14-32, 7:40-44, 8:40-56 (determining “likelihoods associated with each of multiple potential pathways between an origin and a sensitive asset”), 20:56-61 (analyzing “an entry point ... of the customer security architecture, e.g., an adversary, and an endpoint, e.g., a sensitive asset” to establish “possible attack vectors”), Fig. 4B.

78. As shown below, the resulting analysis identifies the network paths an attacker can use to enter a host (e.g., hosts shown on left) and move through the network toward other downstream hosts (e.g., host shown on right). *Id.*, 13:66-14:17 (Fig. 4A illustrating multiple host traversal between “a point of origin (e.g., a point of entry to the network or an attacking entity) and a sensitive asset”), 20:3-7

(contemplating visualization of attack path relative to a selected host of varying granularity); *see also infra*, §§XII.B.7-8.

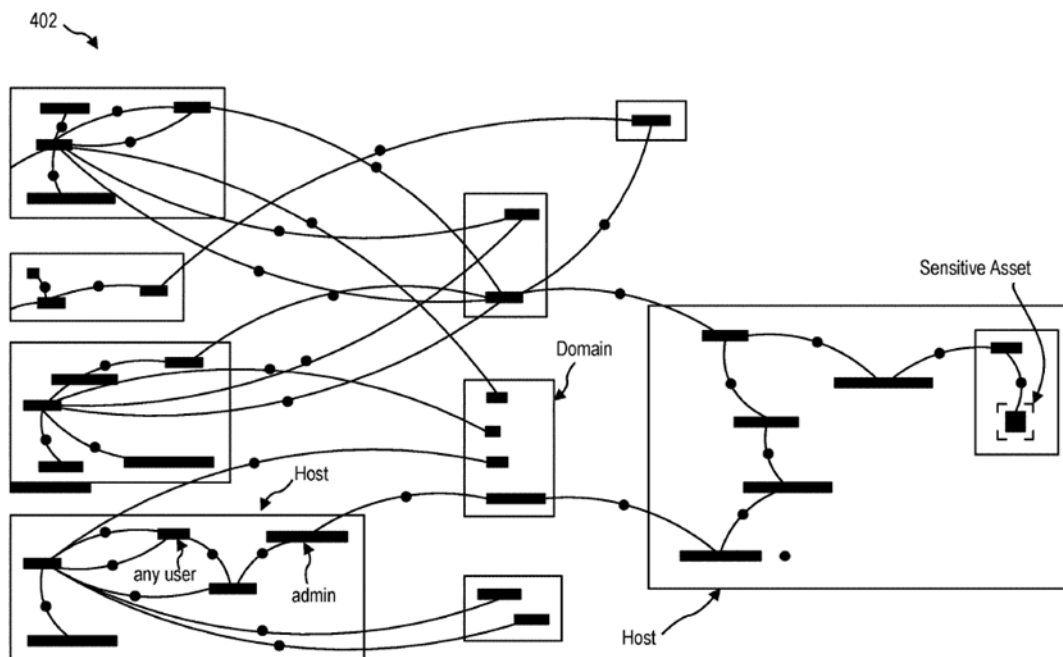


FIG. 4A

EX1006 (Morgan), Fig. 4A.

79. As I also explain more below (*infra*, §XII.B.9), consistent with Keren's teachings of visually presenting network path information to a user of the graphical user interface system, Morgan teaches visually presenting the resulting path analysis. EX1006 (Morgan), 19:55-20:7, Fig. 4B. As I explained above, a POSA would have sought to provide attack path visualizations, as taught by Morgan, for a security object selected by the user in a graphical user interface system like that of Keren so that the user can view the likely network paths that an attacker could take to

compromise the selected object and the downstream consequences to other objects within the selected object's network path. *Supra*, §XII.A. Thus, the combination of Keren and Morgan teaches a “graphical user interface system for providing forward and backward facing attack vector visualizations” and corresponding methods for generating the graphical user interface system, as recited.

80. Finally, the combination of Keren and Morgan teaches a “non-transitory computer-readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform” the elements of claim 20. Keren teaches that the cybersecurity system “may be configured to execute one or more instructions, methods, processes, and the like.” EX1005 (Keren), 6:51-56, Figs. 2-3B. For example, Keren teaches the cybersecurity system includes a “processing circuitry 610 coupled to ... a storage 430.” *Id.*, 20:61-67; *see also id.*, 4:3-4 (“Fig. 6 is a hardware block diagram depicting a cyber-security system, according to an embodiment.”), Fig. 6 (illustrating processing circuitry 610). Keren further teaches that the processing circuitry “may be realized as one or more hardware logic components and circuits,” such as a processor. *Id.*, 21:1-12 (hardware logic components can be “general-purpose microprocessors, microcontrollers, digital signal processors (DSPs), and the like”). Moreover, the storage may be a non-transitory computer-readable medium (e.g., “magnetic

storage,” “optical storage,” or “any other medium which can be used to store the desired information”), which stores software instructions that “when executed by the processing circuitry 610, cause the processing circuitry 610 to perform the various processes” taught by Keren. *Id.*, 21:16-33. In the context of this ground, a POSA therefore would have understood that execution of the instructions contained on the medium would “cause the at least one processor to perform” each of the operations recited in claim 20. *Infra*, §§XII.B.3-9.

81. It is therefore my opinion that the combination of Keren and Morgan teaches the preambles of claims 1, 14, and 20.

2. Element 1.i

82. Keren teaches this element, which recites “at least one processor configured to....” As I discussed above, regarding claim 20, Keren teaches that the cybersecurity system includes a processor used to execute instructions to perform the described methods. *Id.*, 20:61-67 (cybersecurity system includes “processing circuitry”), 21:1-12 (processing circuitry of cybersecurity system includes a processor), 21:16-33 (processing circuitry executes stored instructions to perform methods of Keren); *supra*, §XII.B.1.

83. It is therefore my opinion that Keren teaches element 1.i.

3. Elements 1.1, 14.1, and 20.1

84. Keren teaches these elements, which recite “[identify/identifying] assets in a cloud environment.”

85. As an initial matter, I note that, in the context of the ’154 patent, an “asset” may generally be “a system or device” and, more specifically, may include “a virtual machine, a network appliance, a storage appliance, a compute instance, or an engine instance.” EX1001, 10:26-30, 47:25-42, claims 3, 16.

86. The cybersecurity system of Keren identifies the same types of assets in the form of “security objects.” Specifically, Keren explains that the cybersecurity system is configured to analyze a “cloud environment,” which “represents an organization’s cloud-based resources, and the various connections between such resources.” EX1005 (Keren), 4:33-36. As shown below, “[t]he cloud environment 13 may include a number of cloud computing platforms, 104-1 through 104-n ..., where a cloud platform may include multiple security objects, 105-1 through 105-n....” *Id.*, 4:36-43; *see also id.*, 4:60-65 (“A cloud platform 104 may be a commercially-available cloud system, provided on a service basis, such as, as examples and without limitation, Amazon AWS®, Microsoft Azure®, and the like.”).

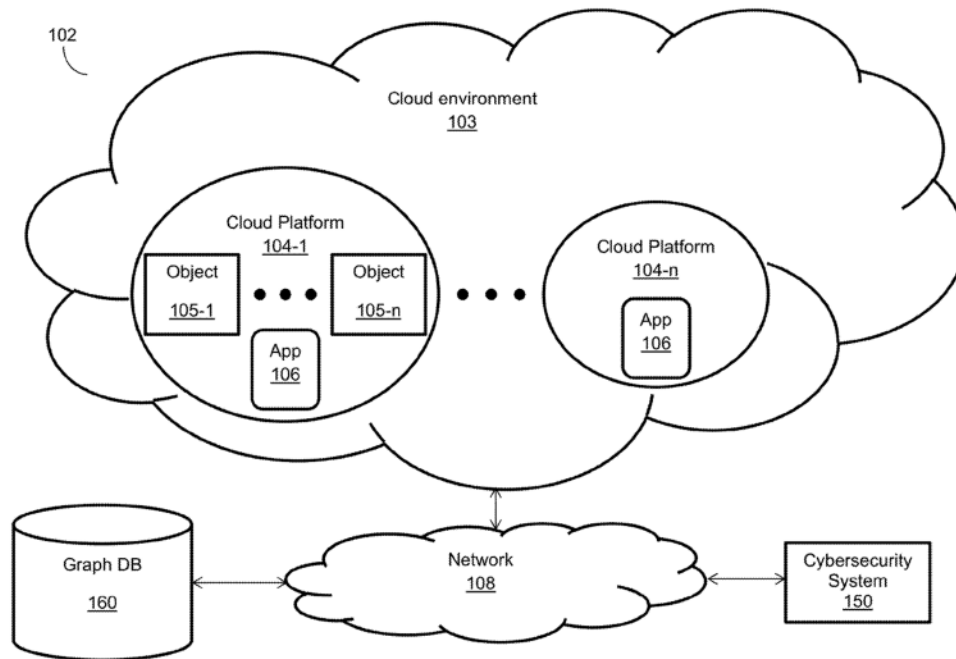


FIG. 1A

EX1005 (Keren), Fig. 1A.

87. Keren explains that the security objects generally are “objects, systems, devices, components, applications, entities, and the like, configured to operate within the cloud platform 104” of the cloud environment. EX1005 (Keren), 5:20-32; *see also id.*, 5:20-6:19. Examples of security objects include virtual machines and other compute and engine instances, such as containers, serverless applications (cloud native), and other cloud processing and management systems. *Id.*, 4:27-51 (security objects may include “third-party containers and objects, container-management systems, such as Kubernetes, as may be virtualized as cloud objects, and the like, as well as any combination thereof”), 5:42-44, 6:13-18. The security objects may also include network appliances, such as proxies, firewalls, gateways,

and load balancers, and storage appliances, such as file servers and databases. EX1005 (Keren), 5:35-39, 5:42-44, 6:13-18; *compare* EX1001, 34:48-49 (explaining a network appliance may be “a load balancer, a firewall, a proxy, or a router”).

88. To perform its analysis, Keren explains that the cybersecurity system identifies the security objects. For instance, Keren teaches that, to “construct[] security graphs” describing the network topology of the cloud environment, “security objects are identified, and security object data is collected” by, for example, “querying a cloud platform” through an API. EX1005 (Keren), 9:21-44, Fig. 2; *see also id.*, 9:56-10:10 (describing collection of security object data, including “object types, object names or unique identifiers (IDs)”), 10:11-14 (“[C]ollection of security object data at S210 may include collection of the identities of all objects included in a cloud platform by generation and transmission of an API query.”).

89. It is therefore my opinion that Keren teaches all aspects of elements 1.1, 14.1, and 20.1.

4. Elements 1.2, 14.2, and 20.2

90. These elements recite “[identify/identifying] risks associated with each of the identified asset [sic].” As I detail below, the combination of Keren and Morgan teaches these elements.

91. Keren teaches that the cybersecurity system identifies risks associated with the identified security objects. Specifically, Keren teaches generating “security insights,” which “are natural-language representations of aspects of the security graph constructed” by the system describing the network topology of the cloud environment. EX1005 (Keren), 12:11-14. In addition to providing insights that “specify a path” between security objects in the network (*id.*, 12:26-44), the security assets may also identify risks associated with the security objects. For example, Keren explains the security insights may describe rare or novel network configurations or events, unauthorized connections to security objects, “such as re-connection of a user’s device to a subnet which the user is not permitted to access,” or anomalous connections, “such as connections displaying spikes of network activity.” *Id.*, 12:45-55. As one specific example relating to a risk identified for a virtual machine, a generated security insight may specify that “VM thirty normally connects to load balancer twenty and firewall eight but is currently only connected to load balancer twenty.” *Id.*, 12:57-61. A POSA would have understood that this insight identified a risk associated with the identified security object (VM thirty) because the virtual machine is identified as no longer being connected to a network security device designed to prevent unauthorized access to the asset. *See, e.g.*, EX1012 (NIST Incident Handling), 17, 37 (explaining firewalls strengthen network

security perimeter);² EX1013 (Li), ¶33 (teaching use of firewall to prevent unauthorized asset exploitation); EX1014 (Basavapatna), ¶¶22-25 (discussing use of passive countermeasures, such as firewalls, to protect against vulnerability exploitation). Another example taught by Keren includes an insight notifying that “objects E, a database, and F, an administrator interface, are currently exposed to external networks,” which a POSA would have understood to identify a risk that assets containing sensitive information or providing privileged access (e.g., a database or administrator interface) were exposed to an external network (e.g., the internet). EX1005 (Keren), 12:61-13:5; *compare* EX1001, 37:25-27 (“In some embodiments, the Internet at large may be considered to be a potentially malicious source.”). Thus, Keren teaches that the cybersecurity system identifies risks associated with each of the identified assets.

92. Morgan also teaches identifying risks, such as vulnerabilities like CVEs, that are associated with each of the devices in a computing environment. *See supra*, §§XI.B, XII.A. Specifically, Morgan teaches that, to identify attack paths

² EX1012 is a copy of an August 2012 NIST publication obtained from <http://dx.doi.org/10.6028/NIST.SP.800-61r2>.

through a network, the system “identif[ies] a plurality of vulnerability instances for the computing network.” EX1006 (Morgan), 2:1-7. The identified vulnerabilities include “any problem or issue that has been called out with respect to a piece of software or a configuration (e.g., a device configuration, an account configuration, an application configuration)” on a host device. *Id.*, 3:56-62 (“As an example, a vulnerability may refer to a known problem that exists with respect to a particular browser or operating system.”); *see also id.*, 19:46-54 (teaching identifying hosts in the network and “associat[ing] the hosts with the vulnerability data”). Morgan explains that the identified vulnerabilities include “a common vulnerabilities and exposures instance (referred to herein as a ‘CVE’).” *Id.*, 3:62-64. “[T]he CVEs are discovered by applying a vulnerability scanne[r] (e.g., a conventional or off-the-shelf vulnerability scanner) that scans the network architecture ... to discover the presence of any number of vulnerabilities thereon,” such as “out of date software, vulnerable application policies, and any other network information that represents a known vulnerability.” *Id.*, 11:66-12:12; *see also id.*, 16:34-38 (“[T]he vulnerability data may be Common Vulnerabilities and Exposure (CVE®) data, i.e., a list of entries that contains an identification number, a description, and a reference for publicly known cybersecurity vulnerabilities.”).

93. As Morgan teaches, and as I described above in §X, this type of vulnerability data facilitates the identification of network paths that could be used as attack paths by an outside attacker. *Id.*, 16:27-38 (“Vulnerability data is added to the customer security architecture to create a comprehensive threat model and facilitate additional analyses, e.g., attack pathfinding and vulnerability prioritization.”), 16:52-66 (“[V]ulnerability data, such as a CVE, is analyzed to create a rule by identifying one or more starting points of the vulnerability (e.g., adversary) and one or more end points for the vulnerability (e.g., user browsers that are affected by the vulnerability). ... That is, a host associated with software impacted by the vulnerability may be a target.”), Fig. 3B (associating attack path edges with the specific vulnerability (e.g., CVE-A) that an attacking entity can exploit to traverse the network toward an asset); *see also* EX1010 (Zeng), 3, Fig. 3, Table 1 (illustrating attack path from an external-facing web server to internal servers based on exploitation of CVEs present on each object); *supra*, §X.

Thus, in view of Morgan, a POSA would have sought to also identify vulnerabilities (e.g., CVEs) associated with each of the security objects identified by Keren’s system in order to identify and output attack path visualizations to the user. *Supra*, §XII.A.

94. It is therefore my opinion that the combination of Keren and Morgan teaches all aspects of elements 1.2, 14.2, and 20.2.

5. Elements 1.3, 14.3, and 20.3

95. These elements recite “[identify/identifying] relationships between at least some of the identified assets, the relationships including at least one of a trust, a network connectivity, or a mechanism of network proxying.” As I explain below, the combination of Keren and Morgan teaches these elements.

96. Keren teaches that, after identifying the security objects and collecting security object data (*supra*, §XII.B.3), the system identifies “relationships between security objects,” which “are descriptions of the various connections between the [identified] security objects.” EX1005 (Keren), 10:49-52, Fig. 2 (step S230). “Network relationships may describe aspects of the connections between objects including, without limitation, connected objects, relevant ports of connected objects, connection bandwidths, connection durations, connection protocols, connection names or IDs, connection statuses, and the like, as well as any combination thereof.” *Id.*, 10:52-57. Thus, Keren teaches that the identified relationships include a relationship of a “network connectivity” between security objects, as recited.

97. Keren also teaches that the identified relationships may include a “trust” or a “mechanism of network proxying,” as recited. Specifically, Keren explains that, to determine the relationships between security objects, “analysis of

object and protocol code and rules” may be performed. *Id.*, 10:58-11:10 (describing example of data collected from a firewall, including the firewall’s port configurations, communication protocols, and log file transmissions), 11:56-60 (further teaching that security object relationships may be determined “by application of observational or active logging methods, such as those methods providing for detection of object-to-object connections by monitoring traffic of a network in use”). A POSA would have understood that this analysis would identify relationships that were of a “trust” relationship or a “mechanism of network proxying.” For example, identifying the objects’ protocol code and rules would identify whether transmissions occurring between security objects were managed by secure communication protocols (e.g., encryption protocols), which a POSA would have understood to identify that a trust relationship was established between the objects. *See, e.g.*, EX1017 (Lichtenstein), 4:41-54 (explaining workloads, client devices, and servers in cloud computing environments “may communicate using encrypted communication, for example by using an implementation of SSH

protocol”); EX1018 (NIST TLS Implementations),³ ii-iii, 1 (“Transport Layer Security (TLS) protocols are used to secure communications in a wide variety of online transactions.... TLS provides a protected channel for sending data between a server and a client.”); *compare* EX1001, 16:17-24 (stating a “trusted relationship ... may refer to a secure communication channel,” such as “a connection or communication network configured with security protocols”).

98. Moreover, Keren’s analysis would also identify whether a relationship between security objects was managed by a mechanism of proxy networking. This is illustrated in Fig. 5 (reproduced below), which depicts a network path determined by identification of security object relationships. *See id.*, 18:40-43. As shown below, the network path includes a public proxy 540 that services traffic between the internet and a database 510 based on “a translation rule, included in the element’s properties, which specifies translation of frontend port 1234 to backend port 4321” (i.e., a mechanism of network proxying). *Id.*, 18:55-58; *see also id.*, 19:56-20:12.

³ EX1018 is a copy of an August 2019 NIST publication obtained from <https://doi.org/10.6028/NIST.SP.800-52r2>.

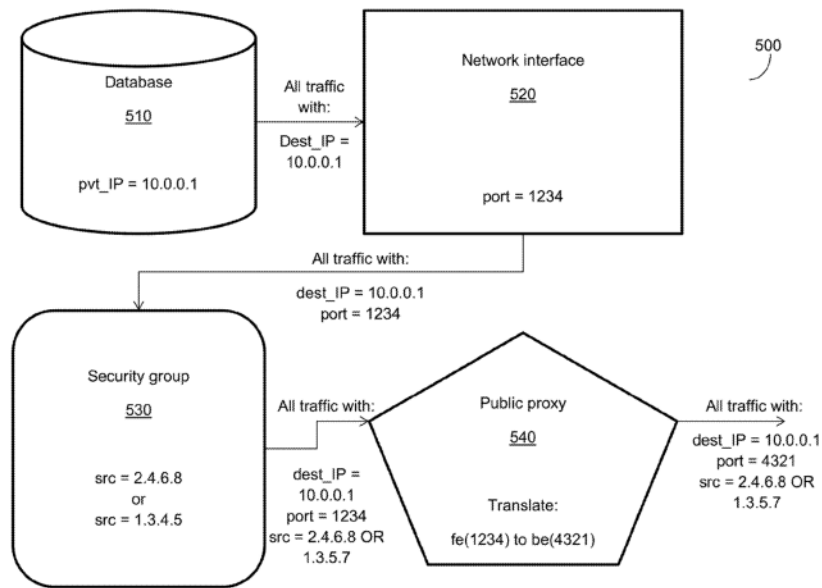


FIG. 5

EX1005 (Keren), Fig. 5; *see also id.*, Figs. 3A (illustrating proxy mechanism between a user account and application), 3B (illustrating proxy mechanisms between the internet and a container).

99. Further, Morgan also teaches identifying relationships between devices in a network that include a “trust” relationship. Specifically, to facilitate attack path analysis, Morgan teaches additionally gathering “access rights data,” such as “node data [and] permissions,” and “application configuration data,” such as “policies, application configurations, node configurations, [and] device configurations.” EX1006 (Morgan), 2:1-5, 7:9-19, Fig. 2. A POSA would have understood that this would include the types of accounts (e.g., user or administrative accounts) and scope of permissions that could be used to access or control a given asset, such as virtual

machines, containers, or databases. *See id.*, 5:14-32 (“[A]n ‘asset’ may refer to any data that is stored or otherwise maintained within the computer network. Examples of assets include files, folders, virtual machines, devices, databases, application, service, account data, or any data object (or set of multiple data objects) that can be accessed via the computing network.”), 11:48-56 (system access rights and application configuration data used to generate attack graph), 14:46-54 (access rights and application configurations permit “a simulated adversary review of network vulnerabilities”), 15:16-38, 23:20-23 (“access rights data indicates levels of privilege”), claim 2. Morgan teaches that this information may be related as nodes in the attack vectors to show relationships between assets that an attacker can traverse by exploiting vulnerabilities. *Id.*, 4:36-67, 15:16-20, 16:45-48. For example, a node may represent an administrator account on an asset that provides fully privileged access to other assets within the network and thus, by gaining access to the node, an attacker can gain entry into connected assets. *Id.*, 4:36-43 (“[A] node may refer to an administrative node or an administrator account having high or fully privileged access to other nodes within a graph.”), 4:54-57 (“[A]n administrator node that is fully connected may include edges connecting the administrator node to all or almost all nodes within a graph.”), Fig. 4B (illustrating relationship between an administrator node of host and a domain). Thus, Morgan also teaches identifying

relationships between assets that include a “trust” relationship, as recited. *Compare* EX1001, 16:15-27 (stating a “trusted relationship” may refer to “a secure communication channel,” such as “an administration link”).

100. It is therefore my opinion that the combination of Keren and Morgan teaches all aspects of elements 1.3, 14.3, and 20.3.

6. Elements 1.4, 14.4, and 20.4

101. These elements recite “[receive/receiving] an identification of a specific asset under investigation.” As I explain below, Keren teaches these elements.

102. Keren teaches that the cybersecurity system is configured to receive an identification of a security object selected by a user so that the user can view detailed network information regarding the selected security object. For example, as I explained above in §XII.B.1, Keren teaches that the cybersecurity system includes a “security graph utility,” which as shown below, generates a “security graph visualization 400” that provides a visual representation of the network topology of the cloud environment, including the security objects contained within the network and their relationships:

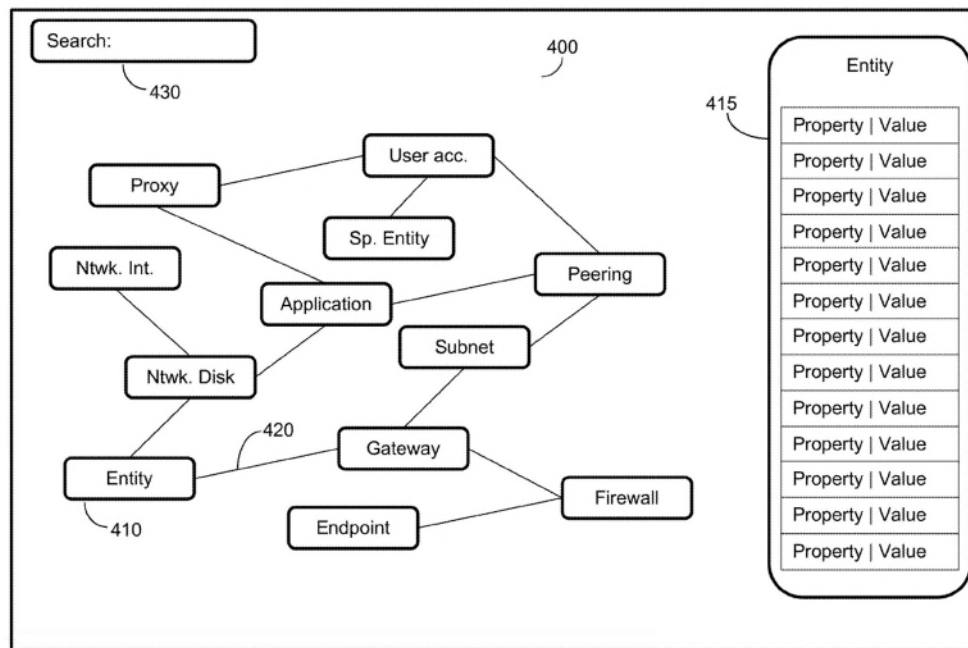


FIG. 4A

EX1005 (Keren), 16:41-45, Fig. 4A.

103. Keren explains that the security graph visualization 400 includes various interactive features. The user, for example, can click on a vertex visualization 410, which represents a security object in the network (i.e., receiving an identification of a specific asset under investigation). *Id.*, 16:54-17:8. This, in turn, may cause the system to generate “a vertex overview pane 415,” which may be an information panel “describing various object data features” of the selected security object, such as “object names, types, statuses, metadata, and the like.” *Id.*, 17:3-18; *see also id.*, 17:19-29 (system may further include “a search tool 300” for

“providing for location and selection of one or more user-specified vertex visualizations 410 or edge visualizations 420 within the graph”).

104. Keren also describes that the system may provide a “security graph object list 450,” which may be “a feature of a security graph management tool or utility, such as a tool or utility configured to provide the security graph visualization of FIG. 4A.” *Id.*, 17:58-63. Keren teaches that the security graph object list 450 “may be configured to list security objects 470 in one or more orders based on factors including ... mode-specific factors....” *Id.*, 17:63-18:1. “Modes are selected list-organization profiles, configured to provide for population of a security graph object list 450 in one or more configurations.” *Id.*, 18:8-10. An example security graph object list 450 is provided below in which a “route” mode has been selected by the user (as indicated by the mode indicator 460):

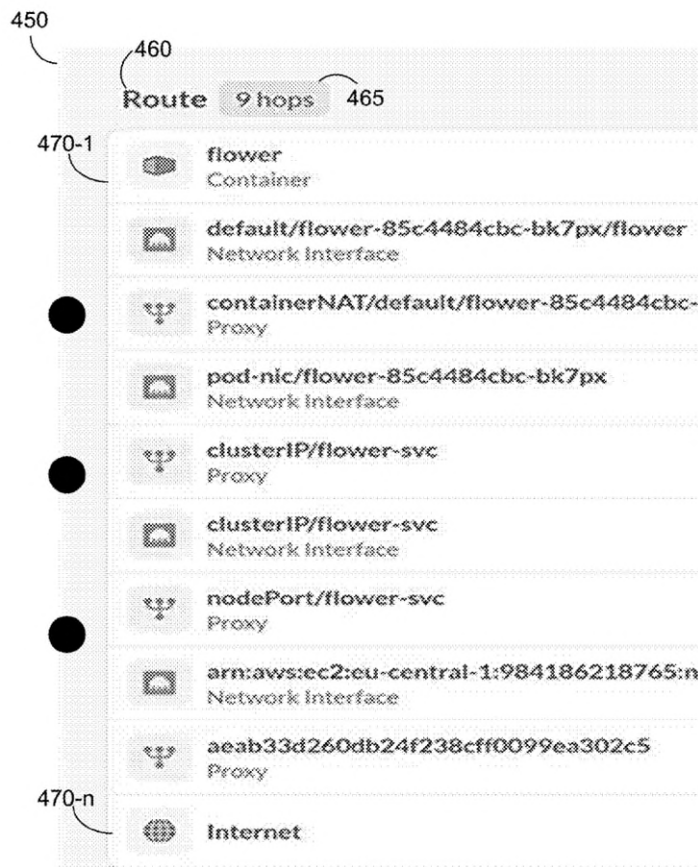


FIG. 4B

EX1005 (Keren), Fig. 4B; *see also id.*, 18:1-22.

105. As shown above, when a route mode is selected, the security graph object list shows the network path between a first object 470-1 (a container) and a destination object 470-n (the internet), where data is transmitted over “nine hops” along intervening security objects within the path. *Id.*, 18:23-27. In order to generate this security graph object list, a POSA would have understood (or at least found it obvious) that, in addition to receiving the specific mode (e.g., route mode)

selected by the user, the security graph utility would also receive an identification of the security object that the user selects (e.g., the container object 470-1) for viewing the network path for that object. *See id.*, 18:27-34 (“According to the same example, the mode indicator 460 may be configured to display ‘Route,’ indicating that the security graph object list 450 is populated to provide information describing a route between the ***specified*** objects....” (emphasis added)).

106. Thus, Keren teaches that the cybersecurity system is configured to receive an identification of a specific asset under investigation (i.e., a security object selected by a user), as recited, so the user can view object-specific network information, such as the object’s properties and network paths. Keren expressly contemplates that other types of object-specific network information may be outputted as part of the user’s selection of a security object. *See, e.g., id.*, 17:46-51 (stating “while the provided security graph object list is configured to provide a list of objects 470 arranged according to a defined mode, other, like, modes and object 470 arrangements may be similarly applicable without loss of generality or departure from the scope of the disclosure”), 18:10-15 (modes may include “routing paths, object utilization or availability descriptions, other, like, configurations, and any combination thereof”). As I explained above in §XII.A, and as I detail more below in §§XII.B.7-8, a POSA would have had reason to output attack vector

visualizations, such as taught by Morgan, in a graphical user interface system like Keren's in response to receiving an identification of a security object under investigation so that the user could view the likely attack paths to and from the selected security object.

107. It is therefore my opinion that Keren teaches all aspects of elements 1.4, 14.4, and 20.4.

7. Elements 1.5, 14.5, and 20.5

108. These elements recite “[perform/performing] a forward analysis of the specific asset under investigation to identify at least one possible attack vector reaching the specific asset via a network outside the cloud environment.” As I explain below, the combination of Keren and Morgan teaches these elements.

109. Keren teaches that the system performs a forward analysis of a specific security object to identify at least one possible vector reaching the specific object via a network outside the cloud environment in several ways. For example, Keren teaches that the system performs a forward analysis that identifies vectors reaching a specific security object via a network outside the cloud environment through the calculation of the security object's “reachable” property. *See* EX1005 (Keren), 13:26-28, Fig. 3A. As Keren explains, “[a] reachable property defines if and how an object on the generated security graph can be reached from an external or internal network, and/or an external or internal object.” *Id.*, 13:28-31; *see also id.*, 12:4-7.

“External means outside of the cloud environment of the organization,” such as the internet. *Id.*, 13:31-32, 13:66-14:1; *see also id.*, 15:50-58 (reachable properties can describe “object, path, or network accessibility properties specific to a given scope, where the given scope may be the internet, various given accounts, various given environments, and the like”), 20:34-42. For example, as shown below, Keren teaches identifying a network path from a public proxy 540 to a database 510 and calculating a reachable property of the database 510 that describes the scope of internet-originating traffic that can reach the database. *Id.*, 18:40-58, 19:34-20:12. Thus, Keren teaches performing a forward analysis of an asset (e.g., database 510) to identify at least one possible vector reaching the specific asset via a network outside the cloud environment.

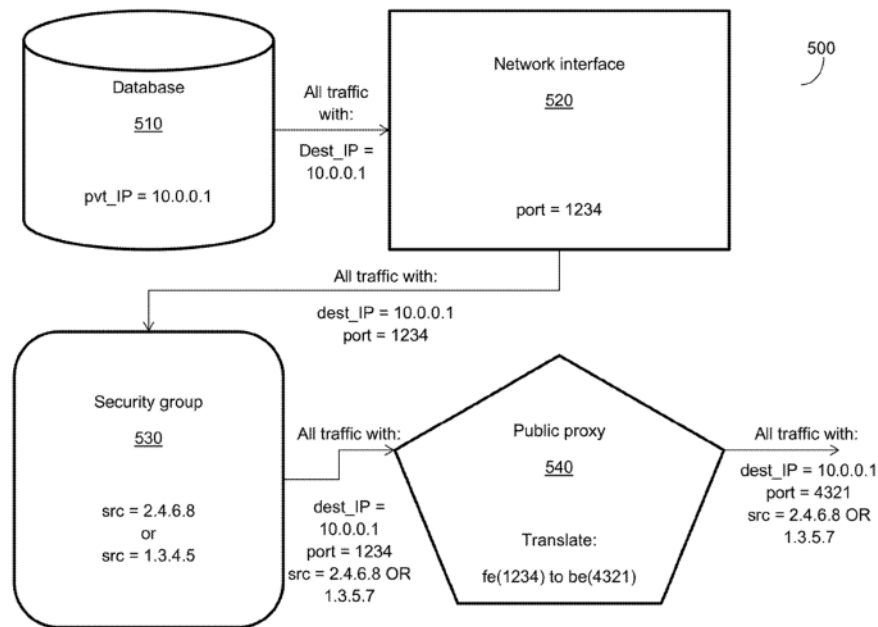


FIG. 5

Id., Fig. 5.

110. In addition, as I explained above in §XII.B.1, Keren teaches that its system is configured to provide forward and backward facing vector visualizations through the security graph utility. For example, as shown below, the security graph utility may output a security graph visualization 400 that provides forward facing vector visualizations of the network paths between security objects within the cloud environment. *See id.*, 16:54-67. I have annotated one example of a network path that a POSA would have understood to be a forward facing vector visualization for a specific security object (e.g., an entity, highlighted by the yellow box). I note that while I have highlighted only one possible vector in the figure, it is done for illustrative purposes. A POSA would have recognized that other possible vectors for a specific security object are possible.

111. As shown below, the visualization includes a forward facing vector from an object that is configured to service internet traffic (e.g., the proxy) to the specific security object (as shown by the red double-headed path, which indicates the data communication exchange occurring between each security object along the path). A POSA would have understood (or at least found obvious) that, to provide such visualizations, the system would perform a forward analysis of the specific

security objects shown in the security graph visualization to identify at least one possible vector (e.g., the red annotated path) reaching the specific objects.

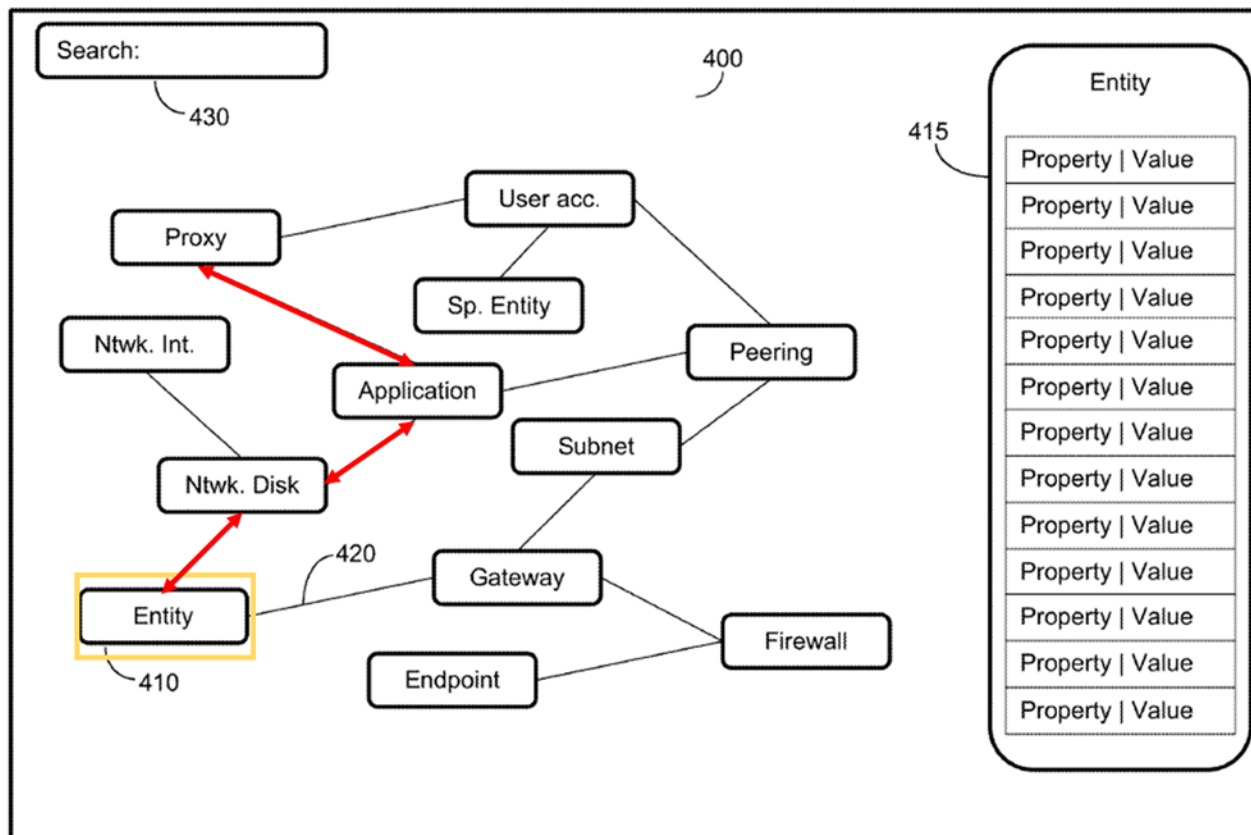


FIG. 4A

EX1005 (Keren), Fig. 4A (annotated).

112. Moreover, as I detailed in §§XII.B.1 and XII.B.6, Keren teaches that the system may provide forward facing vector visualizations through a security graph object list 450 showing the communication route between the internet (i.e., a network outside the cloud environment) and selected security objects within a network path. EX1005 (Keren), 18:23-34, Fig. 4B (reproduced below). In response

to the user's selection, "the security graph object list 450 is populated to provide information describing a route between the specified objects." EX1005, 18:27-30. To provide these visualizations, a POSA would have understood (or at least found obvious) that when a user selected a security object to view the object's network paths, the system of Keren would perform a forward analysis of the selected security object to identify at least one vector reaching the selected object, including those vectors reaching via a network outside the cloud environment, such as the internet.

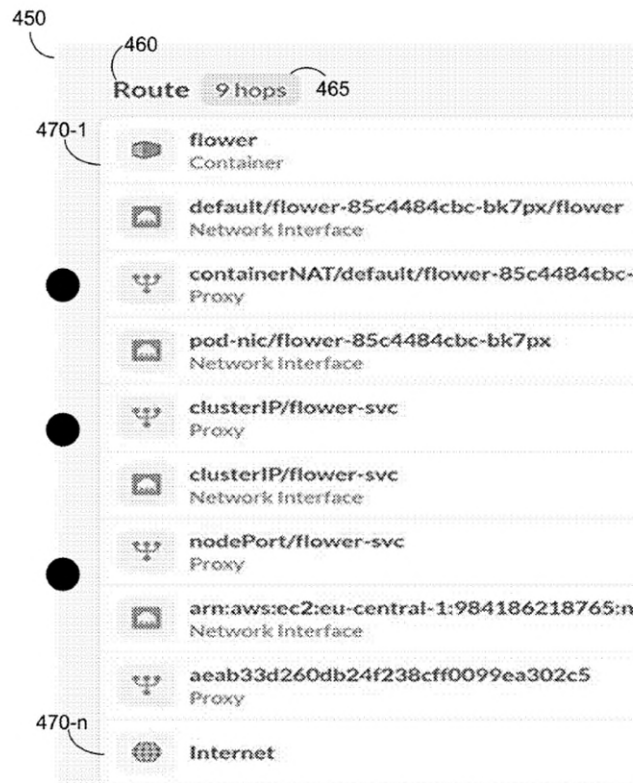


FIG. 4B

Id., Fig. 4B.

113. As I explained above in §XII.A, while the vectors identified by Keren describe the communication paths by which a specific security object can be reached from an outside source, such as the internet, Keren does not explicitly disclose that the vectors analyzed by its system are “attack” vectors. Nevertheless, as I explained above, a POSA would have had reason to provide attack path visualizations, such as taught by Morgan, in a graphical user interface system such as Keren’s when a user selects a security object to view specific network information relating to that object. *Supra*, §XII.A.

114. Morgan teaches performing analyses to generate a “threat graph,” which “define[s] pathways through which data may be processed or communicated and through which assets (e.g., sensitive assets) may be accessed by an attacking entity” (i.e., attack vectors). EX1006 (Morgan), 4:28-35; *see also id.*, 2:10-15 (threat graph includes “a plurality of pathways between an attacking entity (or multiple attacking entities) and one or more sensitive assets where each pathway includes one or more edges connecting access nodes of the computing network”), 3:7-12 (identifying pathways “that correspond to associated likelihoods that a given edge will be used by an attacker to potentially gain access to a sensitive asset”)

115. Morgan teaches that the analyses performed to identify these pathways include identifying pathways to a specific asset from a network outside the cloud

environment. For example, Morgan explains that the analysis may be performed based on an attacking entity that is external to the network that contains the assets. *Id.*, 1:54-60 (computing network may be a “cloud computing network”), 5:1-7 (“[A] pathway may refer to multiple edges that connect an attacking entity or other external computing device with an asset implemented within the computing network.”), 11:57-65 (stating nodes in the graph “may originate from outside the devices on the computing network 102 (e.g., an entry point or an external node)”), 12:25-30 (“an attacking entity may refer to any point of origin of a potential attack,” including “an external device or attack point”), 16:52-66, 24:44-48 (“In one or more embodiments, the attacking entity includes one or more of an external attacking entity originating from outside the computing network....”).

116. To identify attack paths, Morgan further teaches performing a forward analysis that identifies at least one possible attack vector reaching the specific asset from the external attacking entity. For instance, as shown below in an example illustrating a simplified attack graph, the system can identify an attack vector (annotated in red) from an attacking entity 304 (annotated by the red box) reaching a first asset 308a (annotated by the yellow box), which includes a chain of four edges that traverse a corresponding set of nodes. *Id.*, 12:44-50, Fig. 3A (pathway A). As with the previous figure, the attack vector that I have annotated is for illustrative

purposes only. A POSA would have appreciated that other attack vectors were possible (e.g., by exploiting other vulnerabilities, such as CVE-D or CVE-E, to access the first asset).

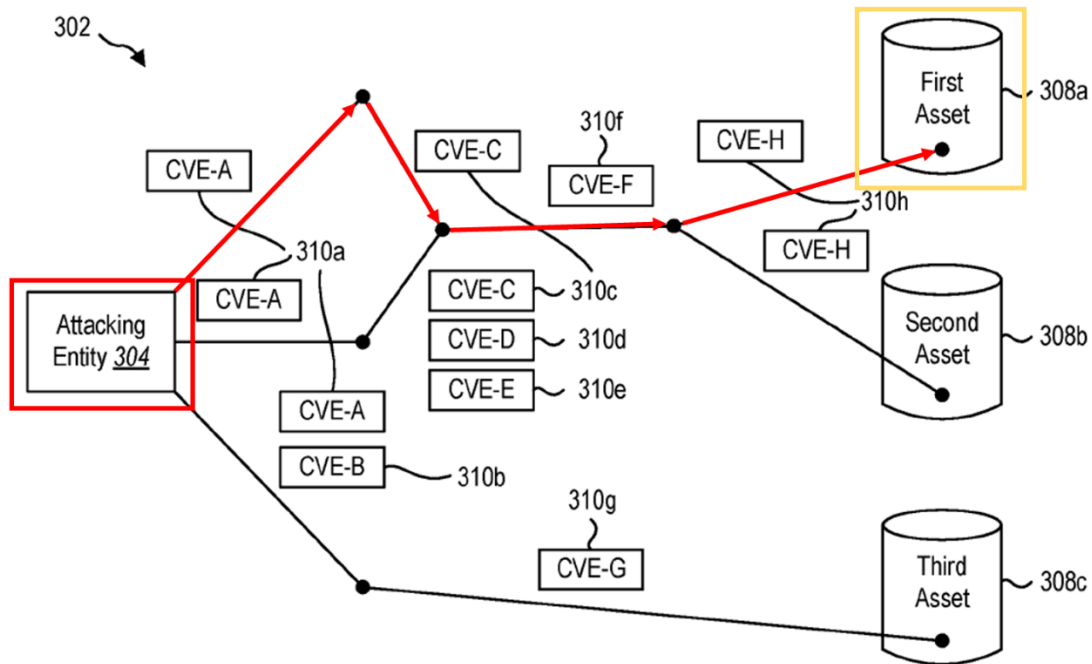


FIG. 3B

EX1006 (Morgan), Fig. 3B (annotated).

117. “[E]ach of the edges within the example graph may have an associated one or more escalations,” which are represented by successful exploitation of vulnerabilities present in the network. *Id.*, 13:8-17. In this manner, Morgan teaches that a forward analysis is performed by determining that an attacking entity can exploit a first vulnerability (e.g., represented by the first edge labeled CVE-A) and

access a first node (representing, for example, a user account), and then can continue to exploit subsequent vulnerabilities until the attacking entity gains access to the first asset 308a. *Id.*, 12:44-63, 14:7-9 (“[A] pathway may be identified between a point of origin (e.g., a point of entry to the network or an attacking entity) and a sensitive asset.”), 20:56-61 (“In this process, an entry point in the graph representation of the customer security architecture, e.g., an adversary, and an endpoint, e.g., a sensitive asset, are selected to establish possible path(s) through a graph structure that represent possible attack vectors.”). Thus, Morgan teaches performing a forward analysis of an asset to identify at least one possible attack vector reaching the asset via a network outside the computing network (e.g., a cloud computing network).

118. Morgan teaches that an attack path analysis can be performed “any number of times with respect to a computing network 102 over time” (e.g., “periodically (e.g., daily, weekly)”), which “enables security personnel to quickly identify new vulnerabilities.” *Id.*, 11:30-47. A POSA would have therefore appreciated the benefits of performing an attack path analysis multiple times, including at the request of a user as taught by Keren, in order to allow security personnel to quickly identify network risks. This is consistent with, for example, Keren’s teaching of populating the network paths for security objects selected by the user, as I discussed above. EX1005 (Keren), 18:23-34, Fig. 4B (route-mode security

object list). Thus, in view of Keren and Morgan, when a user selected a security object to view attack path information for that object, a POSA would have understood (or at least found obvious) that the system would perform a forward analysis of the selected object to identify at least one possible attack vector reaching the security object selected for investigation via a network outside the cloud environment, as recited. *Supra*, §XII.A. A POSA would have understood that this analysis would illustrate to the user the network path that an outside attacker could use to access the selected security object. *Id.*

119. It is therefore my opinion that the combination of Keren and Morgan teaches all aspects of elements 1.5, 14.5, and 20.5.

8. Elements 1.6, 14.6, and 20.6

120. These elements recite “[perform/performing] a backward analysis of the specific asset to identify at least one exposure risk to one or more assets that is in a downstream of the specific asset, wherein the at least one exposure risk includes an identification of an exposed asset, an entry point to the exposed asset, and a lateral movement risk associated with the exposed asset.” As I explain below, the combination of Keren and Morgan teaches these elements.

121. Keren teaches that the system performs a backward analysis of a specific security object that identifies network paths reaching security objects that are in a downstream of the specific object. For example, as I explained above, the

system includes a security graph utility that provides backward facing vector visualizations in a security graph visualization 400. *Supra*, §XII.B.1. One example of a backward facing vector visualization is shown below. As annotated by the blue path, data reaching the specific security object (e.g., the entity, annotated by the yellow box) can be further transmitted downstream to other security objects along the network path (e.g., the endpoint). *See* EX1005 (Keren), 16:54-67; *see also id.*, 18:23-39. As with the forward analysis I discussed above (*supra*, §XII.B.7), a POSA would have understood (or at least found obvious) that, to provide such visualizations, the system would perform a backward analysis of the specific security objects shown in the security graph visualization to identify the network paths reaching objects that are downstream from those objects. I also note that although I have annotated one communication path from a specific security object to other downstream objects as an illustrative example, a POSA would have appreciated that other backward facing vectors could be identified and visualized by the system.

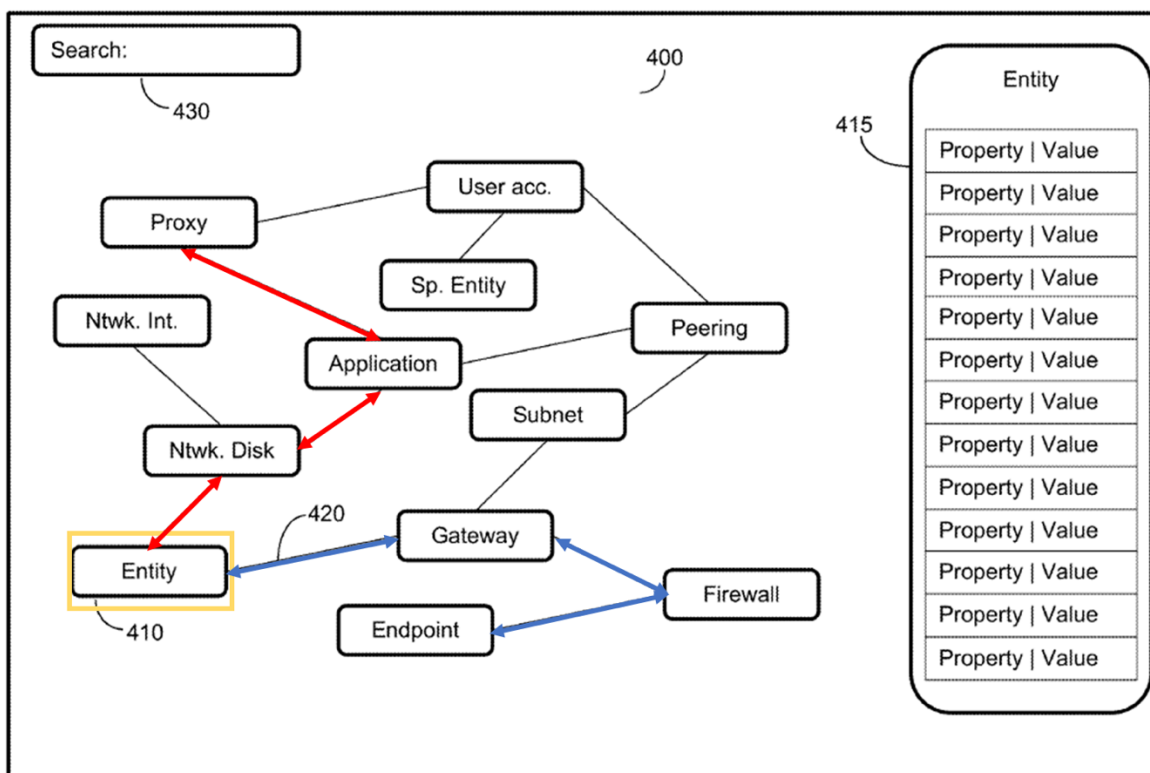


FIG. 4A

EX1005 (Keren), Fig. 4A (annotated).

122. Furthermore, as I discussed above in §§XII.A and XII.B.7, Keren teaches determining reachable properties for security objects within the cloud environment and recognizes that the network properties of a given security object affect the reachability of a security object positioned downstream from that object in the network path. This is illustrated, for example, in the network path of Fig. 5 (reproduced below). As Keren explains, the database 510 is configured to accept all traffic with a destination IP address of 10.0.0.1. *Id.*, 18:44-48, 19:34-43. However, other security objects positioned upstream along the network path further limit the type of traffic that can access the

database 510—for instance, network interface 520 can receive traffic only through port 1234, security group 530 limits transmission to traffic having a source IP address of 2.4.6.7 or 1.3.5.7, and public proxy 540 further provides that only frontend traffic received at port 4321 is forwarded along the network path. *Id.*, 18:48-58, 19:43-20:5. Thus, Keren teaches that the properties of a given security object (e.g., network interface 520, security group 530, or public proxy 540) can inform the risk of exposure a downstream object (e.g., database 510) has to external traffic. *See* 19:66-20:12 (noting database’s exposure to the internet as “true”); *see also supra*, §X.

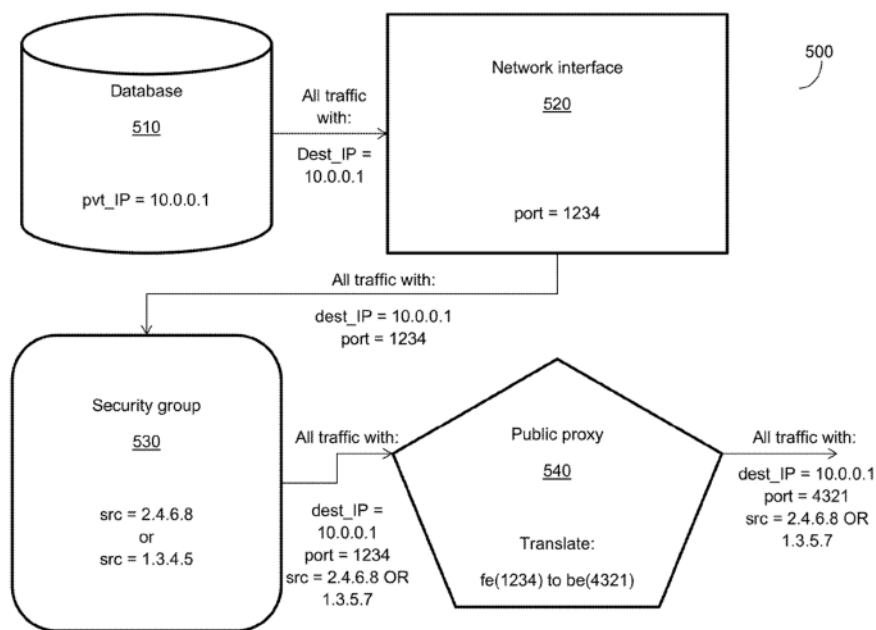


FIG. 5

Id., Fig. 5.

123. As I also explained above in §XII.A, a POSA would have sought to include attack path visualizations, as taught by Morgan, in a graphical user interface

system like Keren's that would allow the user to select a security object to view the object's network paths that the system identifies as attack vectors.

124. Morgan teaches that the attack path analysis identifies attack vectors that identify at least one exposure risk to one or more assets that is in a downstream of a specific asset, where the exposure risk includes an identification of an exposed asset, an entry point to the exposed asset, and a lateral movement risk associated with the exposed asset. More specifically, based on the attack paths described in Morgan, a POSA would have understood that the attack vectors identified by the system includes backward paths that an attack can use to gain access into other assets exposed along the network path. As one example, in the simplified attack graph illustrated in Fig. 3B of Morgan, a POSA would have understood that, after gaining access to the first asset 308a via a forward facing vector (annotated in red), the attacking entity could be able to further leverage its position in the network to move from the first asset 308a to a second asset 308b by exploiting additional vulnerabilities (e.g., instances of CVE-H) along the attack path (annotated in blue).

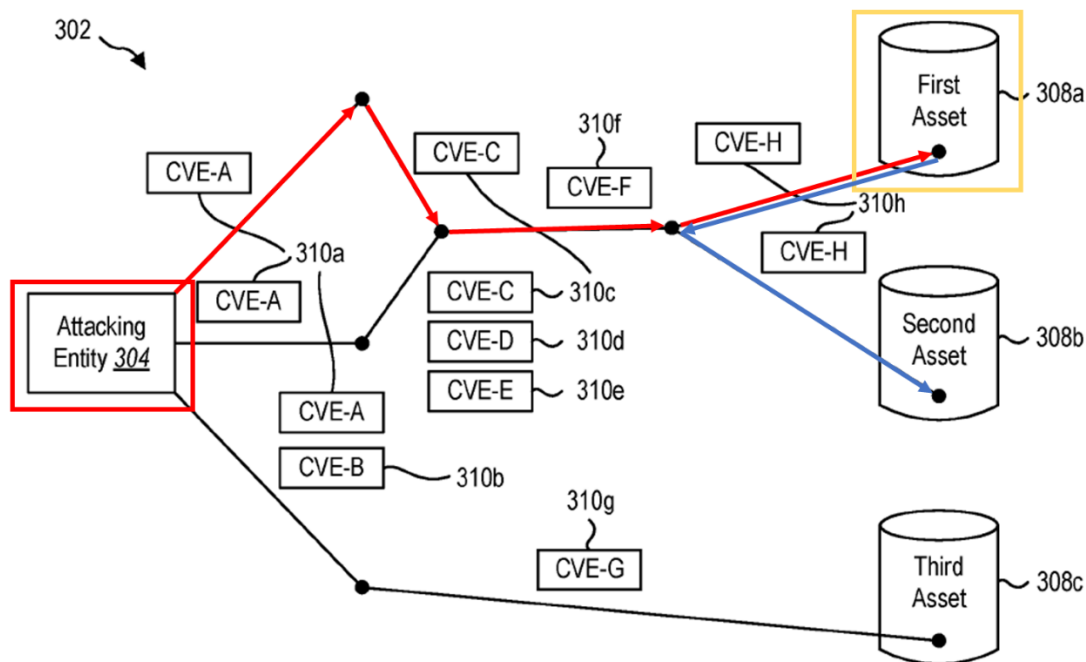


FIG. 3B

EX1006 (Morgan), Fig. 3B (annotated).

125. In addition, Morgan recognizes that an attack path may involve multiple “hops” across hosts within the network in order to reach a target asset. EX1006 (Morgan), 5:10-13 (“[W]here multiple hops are involved in routing a first node to an asset, the pathway may include the multiple edges connecting the first node to the asset.”), 8:19-26 (graph generator may identify “a shortest path between an origin (e.g., an attacker) and a sensitive asset based on a number of hops along edges between the origin and the sensitive asset”), 8:40-57 (system determines “likelihood that an attacker will traverse each of multiple edges in route to accessing a sensitive

asset via the respective pathways”), 14:7-17 (“The network pathway may traverse any number of edges via multiple nodes on one or multiple devices of the computing network.”), 18:22-32. This is illustrated in the example graph below, which shows a number of network devices (as indicated by the black boxes in the figure) that provide network pathways between attacking entities and a target sensitive asset.

Id., 1:43-46, 13:66-14:17.

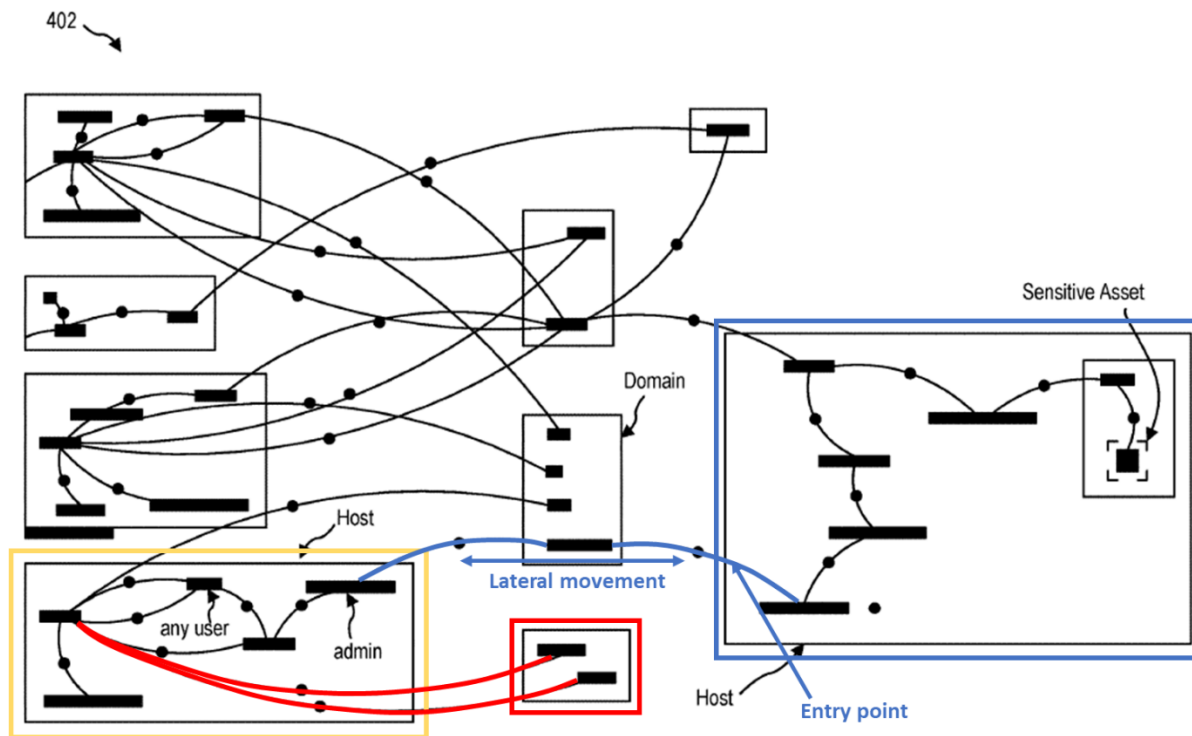


FIG. 4A

EX1006 (Morgan), Fig. 4A (annotated).

126. As shown above, the attack vectors can identify at least one exposure risk to one or more downstream devices relative to a specific host device.

Specifically, a specific host under investigation (e.g., the host annotated by the yellow box) may be accessed via an attack vector (e.g., red annotated paths) by an attacker (e.g., originating from the device annotated by the red box), who can then traverse multiple nodes within the specific host by, for example, exploiting vulnerabilities (represented by edges) to escalate the attacker's privileges to any user then to admin (represented by nodes). *See id.*, 14:12-13 (“the nodes may be associated with any user, administrator, or group of users”). Using admin access, the attacker can then move laterally within the network (as indicated by the blue path) to gain entry into another downstream host (annotated by the blue box), where the attacker can gain access to the sensitive asset contained on the host.⁴ *Id.*, 14:1-

⁴ In the context of Morgan, an “asset” is generally described as “any data that is stored or otherwise maintained within the computing network” and can include “files, folders, virtual machines, devices, databases, application, service, account data, or any data object (or set of multiple data objects) that can be accessed via the computing network.” EX1006 (Morgan), 5:14-32. Thus, while Fig. 4A illustrates the target sensitive asset as being within a host, a POSA would have understood that the term “asset” as used in Morgan broadly applied to any accessible object

12. As I previously noted with other figures, the paths that I have annotated are exemplary, and a POSA would have understood that Morgan's attack path analysis may identify other paths showing other means by which an attacker can use to gain entry into other assets within a network.

127. Thus, a POSA would have understood Morgan as teaching that an identified attack vector can identify an exposure risk to attack to a downstream host (blue) as indicated by the possible pathways that an attacker may be likely to traverse to gain access to the downstream host. As shown above, this exposure risk identifies the exposed asset (e.g., the downstream host (blue)) and an entry point to the exposed asset (e.g., the node that can be accessed from the upstream host (yellow) into the downstream host (blue)).

128. The exposure risk also identifies a lateral movement risk associated with the exposed asset through the identification of the paths from the upstream host (yellow) that the attacker can use to gain access to the downstream host (blue).

within the network, such as the host itself. I further note that the term would have also been understood to encompass objects like those described in Keren (e.g., virtual machines, containers, databases, file servers, etc.).

EX1006 (Morgan), 3:20-36. As was known in the art, “lateral movement” refers to “the ability of a malicious user to move inside the network” and POSAs understood that the pathways between devices visualized in attack graphs were representative of “potential lateral movements of adversaries within computer networks.” EX1007 (Engelberg), ¶¶5, 36; *see also* EX1017 (Lichtenstein), 1:44-47 (“Cloud environments are susceptible to attacks such as lateral movement, in which an attacker attempts to gain access to a resource within the cloud environment, and move through that resource to a next resource in the network.”). It was known that this movement could be accomplished, for example, through one or more network connections between assets or the use of credentials or secrets (e.g., encryption keys) stored on the network, among other methods. *See, e.g.*, EX1017 (Lichtenstein), 1:47-52, 4:30-40, 5:22-33 (detecting potential lateral movement based on exposed encryption keys).

129. Thus, by identifying the potential paths that an attacker could use to move between devices within the computing environment, a POSA would have understood that Morgan’s analysis identifies a “lateral movement risk,” as recited. This understanding is consistent with the ’154 patent. EX1001, 15:14-15, 61:19-39 (a lateral movement risk identifies “potential lateral movement between assets,” for example, where “[a]n attacker who establishes a network foothold usually attempts

to move laterally from one resource to another in search of rich targets such as valuable data”), 69:26-28 (“A lateral movement may include any pathway between assets where a cybersecurity risk can travel from one affected asset to another affected asset.”). Furthermore, as I explain more below in §XII.B.9, in addition to identifying a lateral movement risk as represented by the network paths that an attacker can use to laterally move between devices, Morgan also teaches the lateral movement risk can be further quantified to provide an indication of the likelihood that an attacker will use the particular network path.

130. For the same reasons I discussed above with respect to the forward analysis, a POSA would have appreciated the benefits of performing an attack path analysis at the request of a user as taught by Keren. *Supra*, §XII.B.7; EX1005 (Keren), 18:23-34; EX1006 (Morgan), 11:30-47. Thus, in view of Keren and Morgan, a POSA would have readily adapted a system such that when a user selected a security object to view specific network information for that object, the system would also perform a backward analysis of the selected object to identify at least one exposure risk to one or more security objects that are in a downstream of the selected object, wherein the at least one exposure risk includes an identification of an exposed object, an entry point to the exposed object, and a lateral movement risk associated with the exposed object, as recited. *Supra*, §XII.A. A POSA would have understood

that this analysis would illustrate to the user the overall impacts that a network path of the selected security object would have to the network environment should an outside attacker exploit the path. *Id.*

131. It is therefore my opinion that the combination of Keren and Morgan teaches all aspects of elements 1.6, 14.6, and 20.6.

9. Elements 1.7, 14.7, and 20.7

132. These elements recite “[output/outputting] a signal to cause on a display to present a presentation of forward and backward paths associated with the specific asset, thereby enabling visualization of a plurality of entry points and lateral movement risks associated with the plurality of entry points.” As I explain below, the combination of Keren and Morgan teaches these elements.

133. Keren teaches that the cybersecurity system outputs a signal to cause on a display to present a presentation of forward and backward paths associated with security objects in the cloud environment. For example, as I explained above, and as shown below, Keren teaches that the system includes a security graph utility, which outputs a signal to display the network paths to and from security objects (e.g., the forward (red) and backward (blue) paths annotated below) in the cloud environment via a security graph visualization 400. *Supra*, §§XII.B.1, XII.B.7-8.

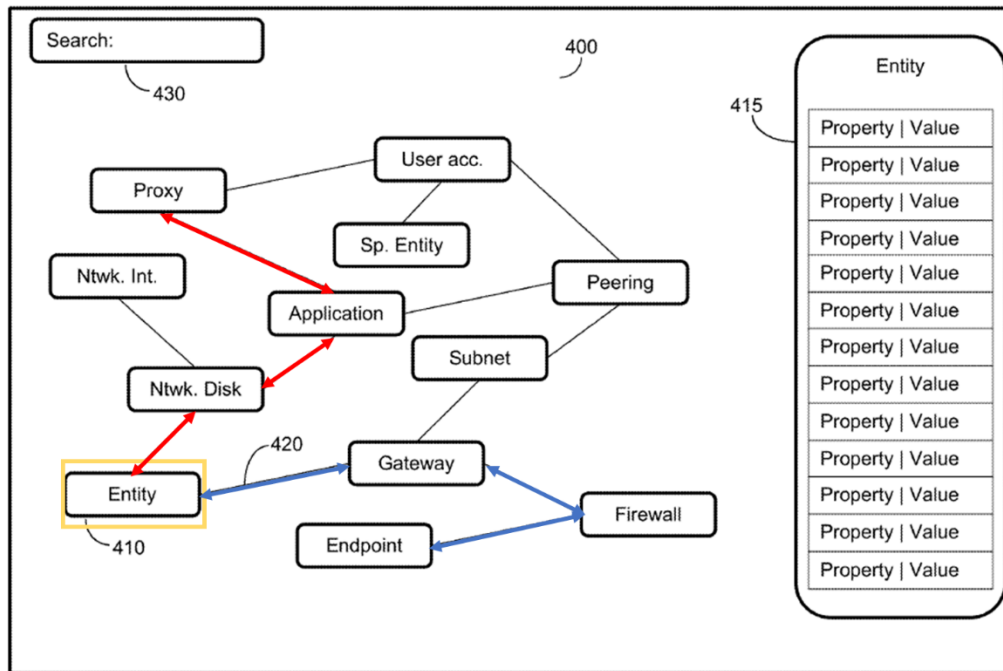


FIG. 4A

EX1005 (Keren), Fig. 4A (annotated).

134. Keren further teaches that the system is also configured to output a signal to display the network paths associated with a specific security object that has been selected by the user, as illustrated by the example shown in Fig. 4B below.

Supra, §§XII.B.1, XII.B.7.

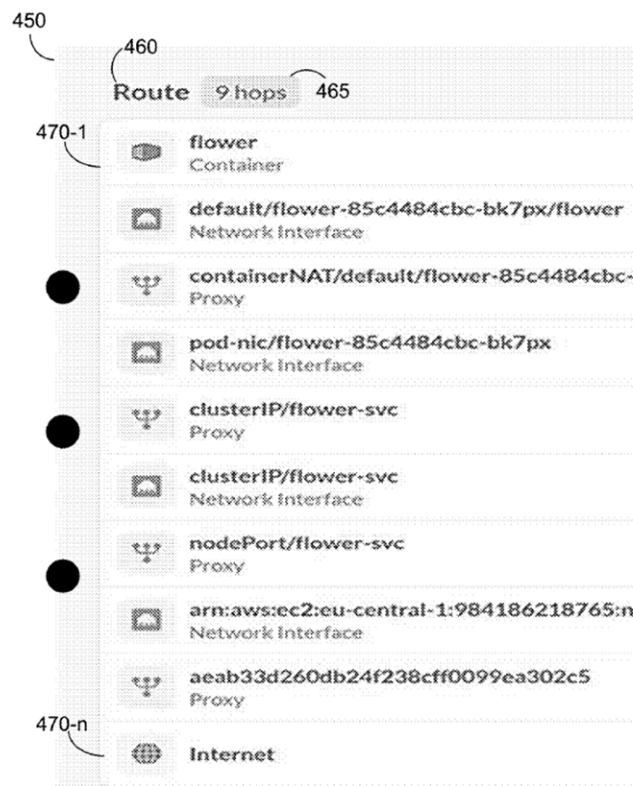


FIG. 4B

EX1005 (Keren), Fig. 4B.

135. As I explained above in §XII.A, while Keren teaches identifying the reachable network paths (i.e., vectors) between security objects in the cloud environment, it does not expressly disclose identifying which of those paths can be attack vectors that an outside attacker can use to gain entry into security objects along a given path. *Supra*, §XII.A. Nevertheless, a POSA would have sought to identify those paths and provide them as attack vector visualizations, as taught by Morgan. *Id.* Specifically, as illustrated below, Morgan teaches visually presenting

to a user the attack vectors identified by the system for a specific host (as shown by pathways illustrated at top of figure). EX1006 (Morgan), 14:18-35, 19:15-34 (“The system may provide visualizations or other data outputs that assist the user in understanding the comprehensive threat assessment.”), 19:55-20:7 (users may view attack graphs for selected host with selectable granularity).

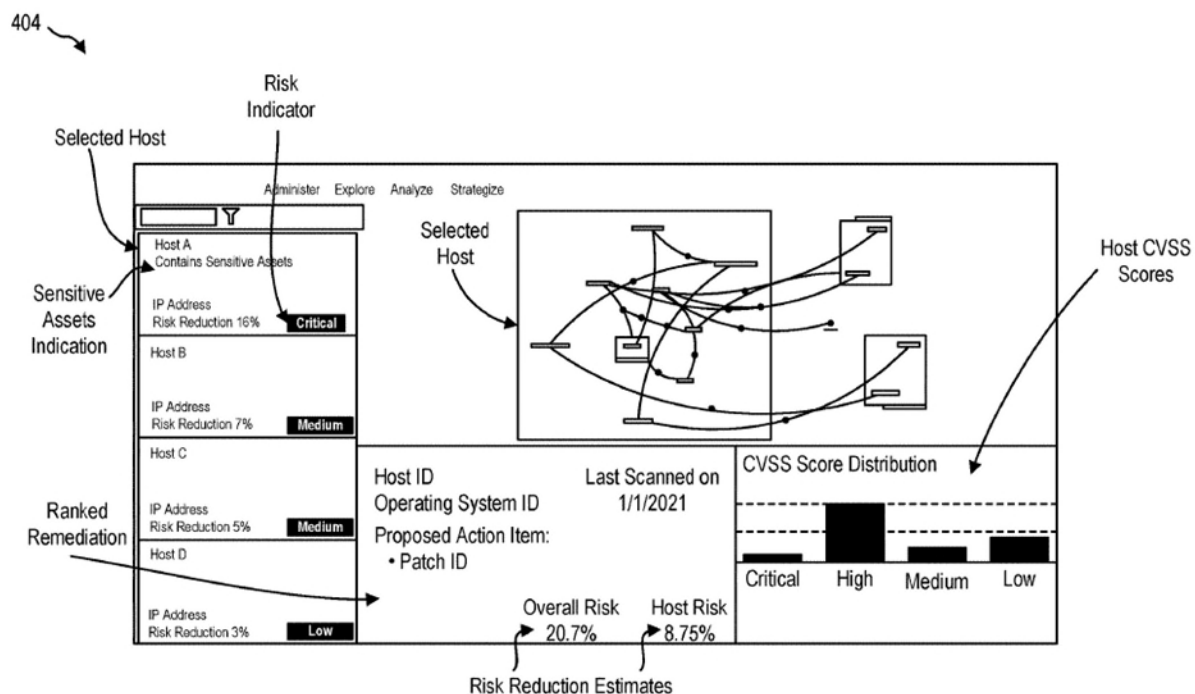


FIG. 4B

EX1006 (Morgan), Fig. 4B.

136. As I discussed above in §§XII.B.7-8, outputting the network paths identified as attack vectors for a selected security object would enable the user to view the entry points by which an attacker could access the selected security object,

via the forward analysis, and the paths by which an attacker could move laterally from the selected security object to access entry points associated with downstream objects, via the backward analysis. See EX1006 (Morgan), 8:40-44 (system “may determine likelihoods associated with each of multiple potential pathways between an origin and a sensitive asset in determining and generating pathways”), 8:61-66 (system “may identify multiple pathways”), 11:16-21, 12:59-63 (graph “may include multiple attacking entities as well as multiple pathways to each of any number of corresponding sensitive assets”), Fig. 4A (illustrating multiple pathways with a plurality of entry points and lateral movement risks). Thus, a POSA would have understood that the attack path visualizations displayed to the user for the selected security object would enable the visualization of a plurality of entry points (i.e., potential entry points into the selected security object and into downstream objects) and lateral movement risks associated with the plurality of entry points (i.e., the lateral paths connecting the selected security object to downstream objects), as recited.

137. Morgan further teaches that the lateral movement risks for an attack vector can be further quantified and presented to a user with the presentation of the attack vector visualizations. Specifically, Morgan explains, “rather than simply identifying vulnerabilities associated with higher likelihood of attack or based on

ease of access of specific pathways between respective nodes of a computing network,” the system considers “a combination of factors to determine a more comprehensive metric of risk to a system.” EX1006 (Morgan), 3:20-36. For example, “even where a specific vulnerability may be associated with a very high likelihood of attack when considered in isolation, the risk management system may nonetheless identify a vulnerability or pathway having a lower likelihood of attack as a much higher priority in lowering overall risk to the computing system.” *Id.*, 3:26-31.

138. Morgan teaches that this risk calculation can be based on a number of factors, such as whether an edge is associated with multiple pathways, certain pathways include a fewer number of edges, or a pathway would lead to higher impact to the computing environment because an attacker could gain access to an important asset. *Id.*, 3:31-36; *see also id.*, 8:27-57, 10:4-16 (“the risk to the system may be a function of edge likelihoods and associated asset importance (e.g., impact)”), 13:32-54, Fig. 3C. Other attributes that may be evaluated to determine the risk that an edge will be traversed by an attacker include whether user interaction is needed to successfully exploit the vulnerability, unwanted side effects, such as system disruption, caused by exploitation, difficulty in developing an exploit, difficulty in identifying the presence of the vulnerability, certainty the vulnerability exists, and

completeness of access granted from a successful exploit. *Id.*, 17:23-35. Morgan teaches that this information is then used to “determine risk scores associated with each edge, each vulnerability, each escalation, or other components that make up or otherwise influence the makeup of the graph.” *Id.*, 9:31-46; *see also id.*, 17:4-16 (score may be “a numeric value (weight) that represents the likelihood an attacker would choose a given vulnerability to exploit a possible attack path”).

139. Morgan further teaches that this risk calculation is presented to the user along with the corresponding attack vectors. *Id.*, 18:33-48 (contemplating prioritizing or ranking pathways and edges in terms of remediation), 19:15-34 (teaching an example where “a user may select an edge between a node representing access to a subnet and a host containing a sensitive asset, and the system displays to the user the associated vulnerabilities along with the reasoning behind the scoring of the edge, i.e., its inclusion in many possible attack paths”), 20:55-21:10. By presenting this risk analysis to the user, Morgan explains that the user can visualize the pathways in which remediation efforts will result in a greater reduction of risk to the system. *Id.*, 10:63-11:29. Thus, in addition to enabling visualization of lateral movement risks by presenting to the user the pathways by which an attacker can laterally move within the network, Morgan also teaches presenting a quantification of the lateral movement risk that an attacker will move along an identified pathway.

Consistent with Morgan's teachings, a POSA would have included this risk quantification with the identified attack vectors outputted to the user upon selection of a security object in Keren's graphical user interface system.

140. It is therefore my opinion that the combination of Keren and Morgan teaches all aspects of elements 1.7, 14.7, and 20.7.

141. Accordingly, it is my opinion that independent claims 1, 14, and 20 were obvious.

C. Dependent Claims

1. Claims 2 and 15

142. Claims 2 and 15 which depend from claims 1 and 14, respectively, and further recite "wherein the network outside the cloud environment includes the Internet." The combination of Keren and Morgan teaches these limitations.

143. As I explained above, Keren teaches analyzing reachable network paths of a security object relative to a network outside the cloud environment, which includes the internet. EX1005 (Keren), 13:31-32, 13:66-14:1, 15:50-58, 20:34-42; *supra*, §XII.B.7. Similarly, Morgan teaches identifying attack pathways originating from an attacking entity that is external to the computing network, which may communicate over the internet. EX1006 (Morgan), 1:54-60 (computing network may be a "cloud computing network"), 5:1-7, 6:23-43 (data may be transmitted to

devices within the computing network through the internet), 11:57-65, 12:25-30, 16:52-66, 24:44-48; *supra*, §XII.B.7.

144. Moreover, both Keren and Morgan express a desire to prevent attackers from gaining unauthorized access into assets contained within the network, which a POSA would have recognized would include malicious attackers originating via the internet. *See* EX1005 (Keren), 1:22-35 (recognizing that increased digital presence exposes organizations to attackers seeking to access sensitive systems and data); EX1006 (Morgan), 1:6-19 (recognizing that, “[a]s the size and complexity of computer networks continue[] to grow,” “the number of security threats continues to grow with respect to unauthorized entities gaining access to information maintained on computer networks”); *supra*, §XII.A. Thus, consistent with Keren’s and Morgan’s teachings, a POSA would have understood that the attack path analysis, including the forward analysis I discussed above, would include identification of possible attack vectors reaching the selected security object via the internet.

145. It is therefore my opinion that claims 2 and 14 were obvious.

2. Claims 3 and 16

146. Keren teaches claims 3 and 16 which depend from claims 1 and 14, respectively, and further recite “wherein the assets in the cloud environment include

at least one of: a virtual machine, a network appliance, a storage appliance, a compute [instance/instances], or an engine instance.”

147. As I explained above, Keren teaches identifying the security objects contained within the cloud environment, which Keren teaches can include virtual machines and other compute and engine instances, such as containers, serverless applications (cloud native), and other cloud processing and management systems. EX1005 (Keren), 4:27-51, 5:42-44, 6:13-18; *supra*, §XII.B.3. The security objects may also include network appliances, such as proxies, firewalls, gateways, and load balancers, and storage appliances, such as file servers and databases. EX1005 (Keren), 5:35-39, 5:42-44, 6:13-18.

148. It is therefore my opinion that claims 3 and 16 were obvious.

3. Claims 4 and 17

149. Claims 4 and 17 depend from claims 1 and 14, respectively, and further recite “wherein identifying the assets in a cloud environment includes identifying the assets based on at least one of: an identity and access management policy, an organization policy, or an access policy.” As I explain below, Keren and Morgan teach these limitations.

150. I explained above how Keren describes collecting security object data as part of identifying the security objects contained within the cloud environment. *See, e.g.*, EX1005 (Keren), 9:26-27 (“[S]ecurity objects are identified, and security

object data is collected.”), Fig. 2 (S210); *supra*, §XII.B.3. The security object data includes information that identifies the security objects and their associated properties, which is then used to by the system to determine the relationships between objects and their network paths. *See, e.g.*, EX1005 (Keren), 9:56-58 (“Security object data is data describing one or more security objects...”), 9:58-10:10 (security object data includes “object types, object names or unique identifiers (IDs)”), 10:11-14 (“[C]ollection of security object data at S210 may include collection of the identities of all objects included in a cloud platform by generation and transmission of an API query.”); *supra*, §XII.B.5. Keren teaches that the security object data may include data describing “object security or access configurations.” EX1005 (Keren), 9:56-66 (emphasis added). Keren further teaches that the system may calculate reachable properties of security objects based on “various given accounts,” which a POSA would have understood assesses whether a security object was reachable relative to a specific account (e.g., a specific user or administrator account). *Id.*, 15:44-58. To calculate this type of reachable property, a POSA would have understood, or at least would have found it obvious, that, as part of identifying the security objects contained within the cloud environment, Keren’s system would also identify the type of accounts that could access the security object, as would have been indicated by the collected security object data describing the

object's access configurations (i.e., access policy). *Id.*, 9:56-66; *see also id.*, Fig. 4A (illustrating system identification of a user account related to certain security objects in the cloud environment).

151. Furthermore, Keren teaches that collection of security object data “may be executed via one or more means including, without limitation, generation and transmission of one or more API queries.” *Id.*, 10:6-10. Keren teaches that these queries may be made through “platform-specific APIs,” which “may be configured to provide for one or more predefined interactions with a cloud platform, such [as] Amazon AWS®, Microsoft Azure®, and the like, where such predefined interactions may include, without limitation, security object identification and data collection.” *Id.*, 9:38-55. POSAs understood that platform-specific APIs such as those described by Keren commonly provided network administrators with a mechanism, known as an “Identity and Access Management” or “IAM” services, to manage the relationships of users of assets contained in the cloud environment. *See, e.g.*, EX1015 (AWS IAM),⁵ 1 (“AWS Identity and Access

⁵ EX1015 is a copy of “AWS Identity and Access Management,” published by Amazon Web Services, Inc., archived on December 25, 2014, by the Internet

Management (IAM) is a web service that you can use to manage users and user permissions under your AWS account.”); EX1016 (Microsoft Azure Fundamentals),⁶ 181-82 (“Azure AD is a robust, secure, multitenant directory service that provides identity and access management in the cloud.”). A POSA would have understood, or at least would have found it obvious, that, as part of identifying the security objects contained within the cloud environment, Keren’s system would also identify the users and permissions associated with a security

Archive’s “Wayback Machine” at

<https://web.archive.org/web/20141225215104/https://docs.aws.amazon.com/IAM/latest/APIReference/iam-api.pdf>.

⁶ EX1016 is a copy of “Fundamentals of Azure: Microsoft Azure Essentials,” 2nd Edition, published by Microsoft Press, archived on April 13, 2021, by the Internet Archive’s “Wayback Machine” at https://web.archive.org/web/20210413101249/https://download.microsoft.com/download/6/6/2/662DD05E-BAD7-46EF-9431-135F9BAE6332/9781509302963_Microsoft%20Azure%20Essentials%20Fundamentals%20of%20Azure%202nd%20ed%20pdf.pdf.

object via an API query made to the cloud platform’s IAM service (i.e., identity and access management policy).

152. Thus, Keren teaches identifying the security objects based on an access policy and/or an identity and access management policy, as recited, because Keren teaches that the system identifies the accounts, users, and permissions associated with a security object. *Compare* EX1001, 50:57-67 (“An identity and access management policy may comprise a software tool to verify identities of users and control access to an associated workload or system functionality. ... An access policy may determine the authorized users of an associated system.”).

153. Moreover, Morgan also teaches identifying assets based on an access policy. As I explained above, Morgan teaches, as part of determining attack paths, obtaining asset rights data and application configuration data, which a POSA would have understood included the types of accounts (e.g., user or administrative accounts) and scope of permissions that could be used to access or control a given asset, such as virtual machines, containers, or databases. EX1006 (Morgan), 2:1-5 (“[A] risk management system may receive network information associated with a computing network including asset data, network architecture data, access rights data, and application configuration data.”), 7:9-19, 15:16-38 (“automated queries may be used to obtain user account details (e.g., credentials), **access policies**,

application configurations and the like” (emphasis added)), Fig. 2; *supra*, §XII.B.5. As I previously noted, this information may be related as nodes in the attack graph to show relationships between assets that an attacker can traverse by exploiting vulnerabilities. *Id.*, 4:36-67, 15:16-20, 16:45-48. Thus, in further view of Morgan’s teachings, a POSA would have understood that, as part of identifying the security objects as taught by Keren, the system would collect data (e.g., asset rights data and application configuration data) that would identify the accounts authorized to access the security objects (i.e., an access policy), which, as I explained previously, would facilitate the determination of attack paths within the cloud environment.

154. It is therefore my opinion that claims 4 and 17 were obvious.

4. Claims 5 and 18

155. Claims 5 and 18 depend from claims 1 and 14, respectively, and further recite “wherein the presentation of the forward and backward paths indicates alternative paths connecting between the specific asset and an upstream asset or a downstream asset.” As I explain below, the combination of Keren and Morgan teaches these limitations.

156. Keren recognizes that security objects may be connected through alternative network paths. For example, as illustrated below, multiple network paths are presented between a proxy and an endpoint (as illustrated, for example, by the green annotated path, in addition to the red and blue paths).

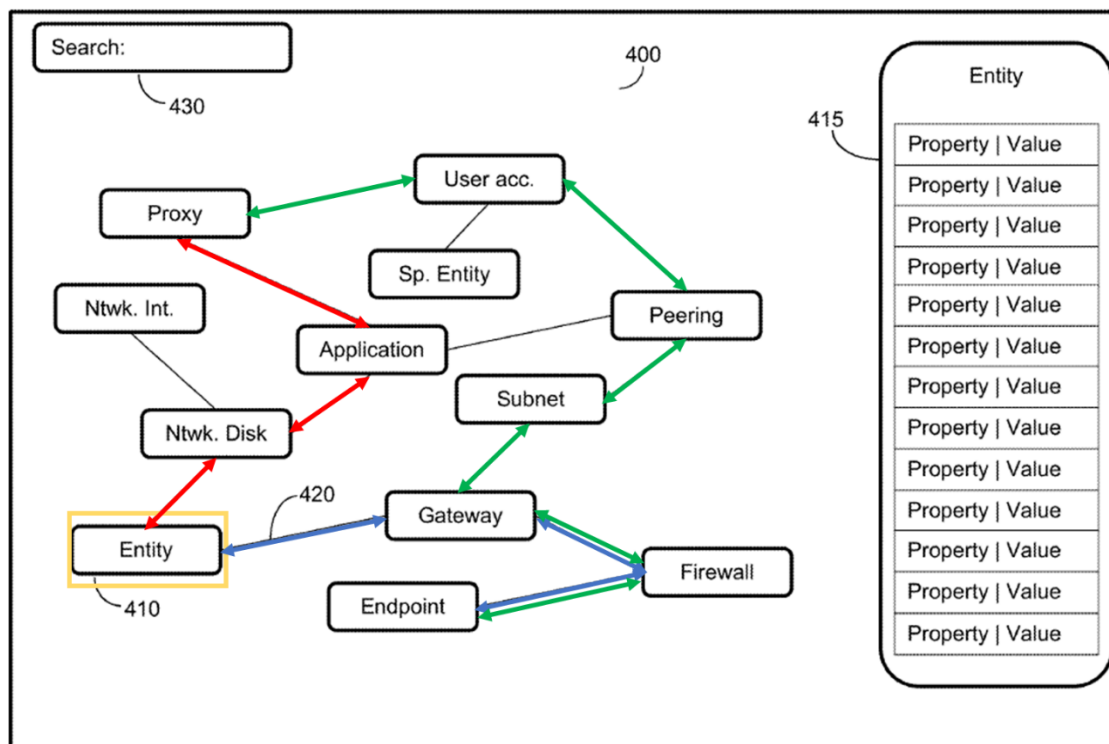


FIG. 4A

EX1005 (Keren), Fig. 4A (annotated).

157. Morgan also recognizes that there may be multiple alternative paths that an attacker can exploit to access an asset. *See, e.g.,* EX1006 (Morgan), 8:40-44, 8:61-66 (system “may identify multiple pathways”), 11:16-21, 12:59-63 (graph “may include multiple attacking entities as well as multiple pathways to each of any number of corresponding sensitive assets”). For example, the graph of Fig. 4A illustrates multiple alternative pathways that can be exploited to access a downstream host. As shown below in the bottom leftmost host (annotated by the yellow box), an attacker that has gained entry into the host (via, for example, the red

paths) can further exploit two alternative edges (annotated in orange and purple) that would allow access to the “any user” node, which can then be used to access the “admin” node. Alternatively, the attacker can exploit another edge (annotated in green) that would bypass the “any user” node and lead to another asset with direct access to the “admin” node. Each of these alternative pathways would ultimately allow access (via the blue path) to the downstream host containing the target sensitive asset (annotated by the blue box).

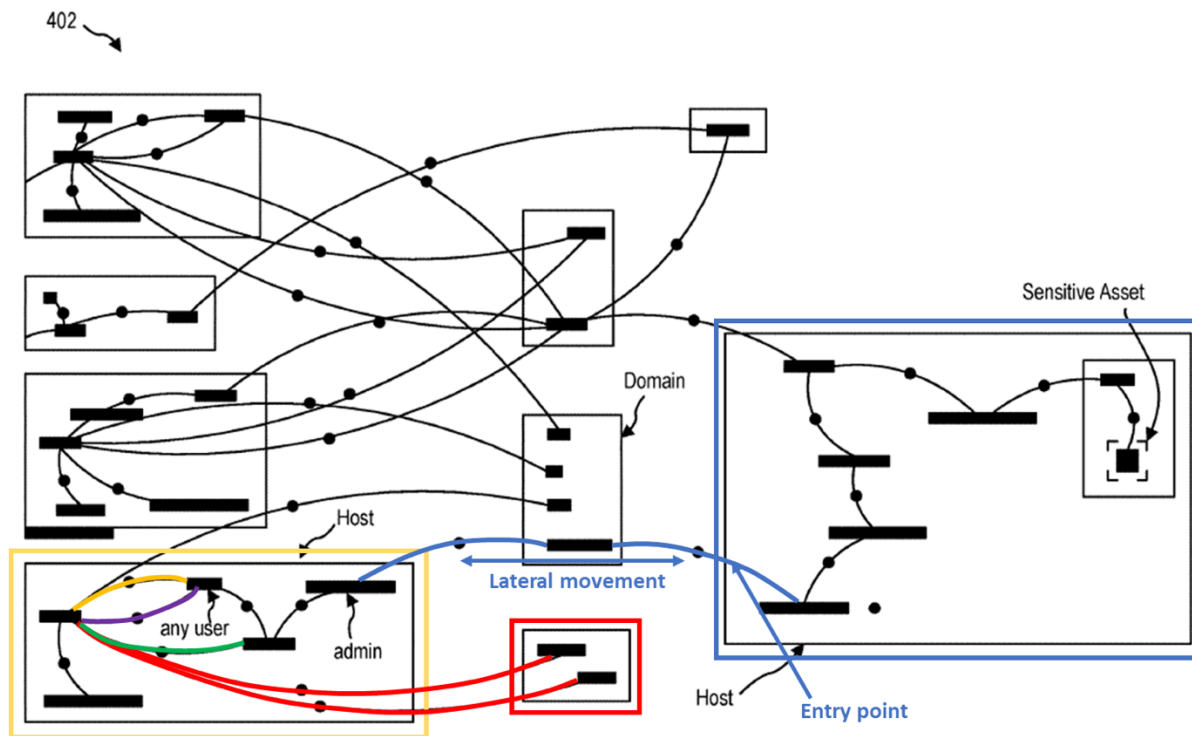


FIG. 4A

EX1006 (Morgan), Fig. 4A (annotated); *see also id.*, Fig. 3B (simplified graph also showing alternative pathways to access an asset).

158. POSAs were well aware that attackers could chain together multiple vulnerabilities along different network pathways in order to access the same target asset. *See, e.g.*, EX1009 (NIST Attack Graphs), 7-8, Fig. 1 (illustrating three attack paths that lead to gaining full privileges on database server); EX1010 (Zeng), Fig. 3; EX1011 (Lallie), Fig. 3 (showing two alternative paths to gain user privileges on a host). Thus, consistent with the teachings of both Keren and Morgan, a POSA would have sought to present forward and backward paths that indicated alternative paths connecting the selected security object and upstream or downstream objects so that the user could visualize the different ways in which an attacker can accomplish an attack.

159. It is therefore my opinion that claims 5 and 18 were obvious.

5. Claims 6 and 19

160. Claims 6 and 19 depend from claims 5 and 18, respectively, and further recite “wherein the visualization includes a presentation of the alternative paths.” The combination of Keren and Morgan teaches these limitations.

161. I explained in the preceding section (§XII.C.4) how a POSA would have sought to present forward and backward paths that indicated alternative paths between the selected security object and upstream or downstream objects. Both Keren and Morgan teach presenting identified paths as a visualization to the user (e.g., as a graph). *See, e.g.*, EX1005 (Keren), Figs. 4A-4B; EX1006 (Morgan), Fig.

4B. Thus, a POSA would have included in the attack path visualization outputted to the user a presentation of the alternative paths identified by the system.

162. It is therefore my opinion that claims 6 and 19 were obvious.

6. Claim 7

163. Claim 7 depends from claim 5 and further recites “wherein the at least one processor is further configured to detect a potential risk associated with the specific asset based on the monitored network activities.” As an initial matter, I note that, while claim 7 recites that risk is detected based on “the monitored network activities,” claim 5 does not reference “monitored network activities.” Nevertheless, as I explain below, the combination of Keren and Morgan teaches this limitation.

164. Keren teaches monitoring network activities to detect a potential risk associated with a specific security object. As I mentioned above, Keren teaches that the system is configured to generate security insights that provide descriptions of the current network configuration, including descriptions relating to risk. EX1005 (Keren), 12:11-16, 12:45-13:5; *supra*, §XII.B.4. Keren teaches that the security insights may be generated based on monitored network activities. For example, the security insights can describe “network management or anomaly detection data,” which include descriptions of risk associated with network activities monitored by the system. EX1005 (Keren), 12:45-55. Keren explains that this includes:

descriptions of network configurations or **events** which are rare or novel, such as connection of a new device to a network, descriptions of connections which are unauthorized, such as **re-connection** of a user's device to a subnet which the user is not permitted to access, descriptions of **connections which display anomalous behavior**, such as connections displaying **spikes of network activity**, and the like.

Id. (emphasis added); *see also id.*, 11:56-60 (teaching system monitors network activity “by application of observational or active logging methods, such as those methods providing for detection of object-to-object connections by monitoring traffic of a network in use”). Each of these examples provide description of a risk associated with a specific security object based on monitored network activities (e.g., new or unauthorized connection of a user device, increased network activity associated with an asset) because they may be indicative of attempts by an outside attacker to gain access to a security object within the network. Thus, Keren teaches that the system (and thus, its processor) is further configured to detect a potential risk associated with the specific asset based on the monitored network activities, as recited.

165. Morgan also teaches detecting a potential risk relating to an identified attack path for a specific asset based on monitored network activities. For instance, as part of quantifying the risk that an attacker would use an identified pathway (*see supra*, §XII.B.9), Morgan teaches establishing “a rule indicating recent use” in weighing the risk that an edge will be exploited by an attacker. EX1006 (Morgan),

17:49-52. As Morgan explains, “it may be known from customer security architecture that a first device, having vulnerability A, is not frequently used, whereas a second device, having vulnerability B, is used frequently” (i.e., monitored network activities). *Id.*, 17:52-55. If both vulnerabilities A and B require user interaction for exploitation (e.g., interacting with a web page), Morgan teaches that the risk associated with the pathway edge corresponding to vulnerability B will be weighted higher than the pathway edge corresponding to vulnerability A. *Id.*, 17:55-63. Thus, Morgan also teaches detecting a potential risk associated with the specific asset based on the monitored network activities (e.g., a risk that an edge will be traversed by an attacker), as recited.

166. It is therefore my opinion that claim 7 was obvious.

7. Claim 8

167. Claim 8 depends from claim 1 and recites “wherein the presentation of the forward and backward paths indicates port numbers for each pathway.” As I explain below, the combination of Keren and Morgan teaches this limitation.

168. In order to determine the relationships between security objects, identify the network paths associated with the cloud environment, and calculate reachability properties for objects, Keren teaches that its system collects security object data, which includes the port configurations, such as the port numbers and statuses, by which a given security object receives communications. *See* EX1005

(Keren), 6:5-9, 9:56-66 (collected security object data includes “object port configurations”), 10:52-57 (network relationships describe “relevant ports of connected objects”), Fig. 5 (illustrating port numbers as part of network path). Keren teaches that these ports numbers may be presented to the user through the security graph utility. *Id.*, 11:66-12:11 (descriptions of reachability properties may be included in the security graph, including “an address, a port number, a protocol, and the like” that “would describe how to reach the object”), 12:26-44 (describing security insight that details the relevant port numbers for data transmission between specific security objects), 17:1-18 (vertex overview pane of security graph utility may present to user “object data features,” including collected security object data), claim 7.

169. A POSA would have also understood that the port numbers associated with a security object was also relevant to the attack path analysis. For instance, POSAs understood that, to exploit a vulnerability present on an asset, an attacker would need to use an open port that was in communication with the vulnerability (e.g., a vulnerable software). This is illustrated below, where Table 1 lists the vulnerabilities present on a number of assets in the network, Table 2 lists the communication rules between the assets, and Figure 3 illustrates an attack graph showing possible paths for vulnerability exploitation. As shown below, in order to

exploit the vulnerability associated with the web server (CVE-2015-1635) and gain privileges to execute arbitrary code, an attacker would need to send a malicious HTTP request through the port configured to receive HTTP requests (port 80) on the web server. The same applies for the subsequent vulnerabilities along the path (e.g., to exploit CVE-2012-2556 or CVE-2013-4465 on the FTP server, communication is made through port 21 on the FTP server).

TABLE 1: Vulnerabilities on each server.

Server	Vulnerabilities	CVE ID
WS	Allow remote execution of code	CVE-2015-1635
DBS	Remote execution of SQL command	CVE-2014-1466
FS	Allow remote execution of code	CVE-2013-4465
	Allow remote execution of code	CVE-2012-2526
AS	Allow remote execution of code	CVE-2009-0241

TABLE 2: Communication rules between servers.

Source server	Destination server	Protocol and port
202.120.234.6	WS	HTTP (80)
WS	DBS	SQL (1433)
	DBS	SQL (1433)
	FS	FTP (21)
AS	WS	HTTP (80)
	DS	DNS (1024)

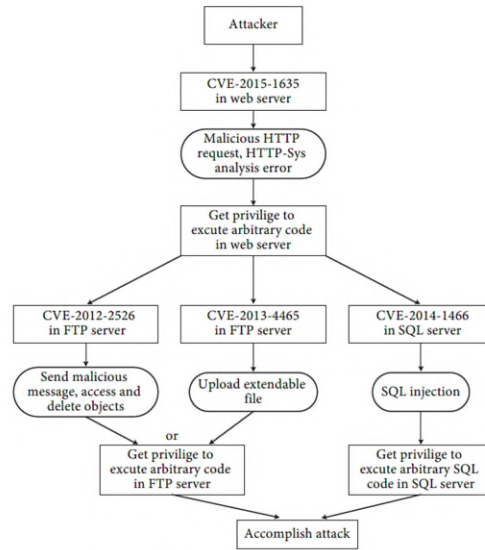


FIGURE 3: The corresponding attack graph.

EX1010 (Zeng), 3 (Tables 1 & 2, Fig. 3).

170. Given the above, because Keren teaches presenting to the user the port numbers by which security objects communicate and because it was understood that port numbers were relevant to the attack path analysis, a POSA would have had reason to also indicate the port numbers that an attacker would use in a given attack path visualization presented to the user so that the user could better visualize the

exact communication means through which an attack could be accomplished. A POSA would have understood that this information would be especially important to visualize to the network administrator when the cloud environment includes security objects that are configured to use non-standard service ports or customized services that are assigned organizationally.

171. It is therefore my opinion that claim 8 was obvious.

8. Claim 9

172. Claim 9 depends from claim 1 and further recites “wherein the visualization of the entry points indicates at least one entry point at risk.” As I explain below, Morgan teaches this limitation.

173. As I explained above in §XII.B.9, a POSA would have had reason to provide attack paths identified by the system as visualizations to the user, which would have visualized a plurality of entry points by which an attacker could access a security object. As I also explained above, Morgan teaches quantifying the risk that an attacker will exploit an identified attack path, which a POSA would have sought to additionally include in the corresponding attack path visualizations. Because Morgan teaches quantifying a risk associated with an identified attack pathway (e.g., the likelihood that an attacker will traverse a pathway edge), a POSA would have understood that this would indicate the entry points associated with the pathway that is at risk.

174. It is therefore my opinion that claim 9 was obvious.

9. Claim 10

175. Claim 10 depends from claim 1 and further recites “wherein the at least one processor is further configured to monitor network activities of the assets in a cloud environment.”

176. As I explained above in §XII.C.6, Keren teaches that its system is configured to detect risk based on monitored network activities of the security objects contained in the cloud environment. EX1005 (Keren), 12:45-61. Keren further teaches that its system is configured to monitor the network activities of the security objects. *Id.*, 6:47-51 (cybersecurity system provides one or more network analysis functionalities, including “network analysis” and “traffic analysis”), 11:56-60 (system may apply “observational or active logging methods, such as those methods providing for detection of object-to-object connections by monitoring traffic of a network in use”).

177. It is therefore my opinion that claim 10 was obvious.

10. Claim 11

178. Claim 11 depends from claim 1 and further recites “the at least one processor is further configured to detect a potential risk associated with the specific asset based on a network activity of the specific asset.” The combination of Keren and Morgan teaches this limitation.

179. As I explained above in §XII.C.6, Keren teaches that its system is configured to detect a potential risk associated with a specific security object based on monitored network activities of the security object, such as new or unauthorized connections associated with a user device, or abnormal network activity. Similarly, Morgan teaches detecting a potential risk associated with a specific asset based on the frequency of use of that asset when analyzing the risk of attack pathway traversal. Thus, both Keren and Morgan teach systems that detect a potential risk associated with the specific asset based on a network activity of the specific asset, as recited.

180. It is therefore my opinion that claim 11 was obvious.

11. Claim 12

181. Claim 12 depends from claim 1 and further recites “the at least one processor is further configured to detect a potential risk associated with the specific asset based on a network activity of an upstream asset of the specific asset.” Morgan teaches this limitation.

182. As I explained above in §XII.B.9, Morgan teaches quantifying a risk that an attacker will use an identified path to access an asset. As I also explained in §XII.C.6, this risk analysis can include consideration of the network activity of an asset within the path, such as whether a given asset (e.g., a user device) is frequently used. See EX1006 (Morgan), 17:49-63. Morgan teaches assigning a weight based on this network activity to an edge of the attack path that corresponds to a

vulnerability on the asset, which “reflects the likelihood of a successful exploitation of the vulnerability impacting the edge.” *Id.*, 18:4-8. “Each edge in a potential path, once assigned a weight, is aggregated along with the other edges to assign a likelihood score for the overall given path, which can then be prioritized and provided to the user.” *Id.*, 18:8-11. Thus, Morgan teaches that the potential risk associated with a specific asset is based on the aggregated risk that an attacker will exploit each of the edges along the attack path to the specific asset. Based on Morgan’s teachings, a POSA would have understood that the potential risk calculated for the specific security object in Keren’s system would therefore include the weight assigned to the edge associated with an upstream object (e.g., the likelihood that a vulnerability present on the upstream object will be exploited by an attacker) based on that object’s network activity (e.g., frequency of use).

183. It is therefore my opinion that claim 12 was obvious.

12. Claim 13

184. Claim 13 depends from claim 1 and further recites “the at least one processor is further configured to detect a potential risk associated with the specific asset based on a network activity of a downstream asset of the specific asset.” The combination of Keren and Morgan teaches this limitation.

185. As I explained above in §§XII.A and XII.B.8, a POSA would have sought to present to a user attack path visualizations for the selected security object

so that the user could visualize how an attacker could potentially use the selected security object to impact downstream objects. As I also explained above in §XII.B.9, a POSA also would have sought to include calculations of potential risk that an attacker would use an identified pathway as part of the attack path visualization outputted to the user, as taught by Morgan. Morgan teaches that this analysis includes assessing the risk that an attacker would exploit a vulnerability associated with a pathway edge based on the network activity of the asset containing the vulnerability (e.g., frequency of use). EX1006 (Morgan), 17:49-18:21; *supra*, §XI.C.6.

186. A POSA would have appreciated that these teachings would equally apply to objects located downstream from the selected security object along a given attack path. For example, consistent with Morgan's teachings, if a downstream security object was not frequently used, then the potential risk that an attacker could leverage the selected security object to move toward the downstream object would be diminished. Thus, a POSA would have further included as part of the risk calculation associated with the selected security object the likelihood that an edge associated with a downstream object will be traversed based on that object's network activity.

187. It is therefore my opinion that claim 13 was obvious.

XIII. CONCLUDING STATEMENTS

188. In signing this declaration, I understand that the declaration will be filed as evidence in a contested case before the Patent Trial and Appeal Board of the United States Patent and Trademark Office. I acknowledge that I may be subject to cross-examination in this case and that cross-examination will take place within the United States. If cross-examination is required of me, I will appear for cross-examination within the United States during the time allotted for cross-examination.

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

Dated: 01/17/2025

By: /Angelos Stavrou/
Angelos Stavrou, Ph.D.

XIV. APPENDIX A – MATERIALS CITED

Exhibit No.	Description
1001	U.S. Patent No. 11,582,257 to Shua
1003	<i>Curriculum Vitae</i> of Dr. Angelos Stavrou
1004	File History of U.S. Patent Application No. 17/657,981 ('154 Pat.)
1005	U.S. Patent No. 11,374,982 to Keren
1006	U.S. Patent No. 12,015,631 to Morgan
1007	U.S. Publication No. 2022/0263855 to Engelberg
1008	U.S. Patent No. 11,831,671 to Bruttomesso
1009	Singhal, “Security Risk Analysis of Enterprise Networks Using Probabilistic Attack Graphs,” NIST Interagency Report 7788 (Aug. 2011) https://doi.org/10.6028/NIST.IR.7788
1010	Zeng et al., “Survey of Attack Graph Analysis Methods from the Perspective of Data and Knowledge Processing,” Security and Communication Networks (Dec. 2019)
1011	Lallie et al., “A Review of Attack Graph and Attack Tree Visual Syntax in Cyber Security,” Elsevier Computer Science (2019)
1012	Cichonski, “Computer Security Incident Handling Guide: Recommendations of the National Institute of Standards and Technology,” NIST Special Publication 800-61, Rev. 2 (Aug. 2012) http://dx.doi.org/10.6028/NIST.SP.800-61r2
1013	EP Patent No. 3,360,071 B1 to Li
1014	U.S. Publication No. 2013/0191919 A1, published July 25, 2013, to Basavapatna
1015	Amazon Web Services, Inc., “AWS Identity and Access Management: API Reference,” available at: https://web.archive.org/web/20141225215104/https://docs.aws.ama

Exhibit No.	Description
	zon.com/IAM/latest/APIReference/iam-api.pdf, archived December 25, 2014
1016	Collier M. and Shahan R., “Fundamentals of Azure: Microsoft Azure Essentials,” 2nd Edition, Microsoft Press (2016), available at: https://web.archive.org/web/20210413101249/https://download.microsoft.com/download/6/6/2/662DD05E-BAD7-46EF-9431-135F9BAE6332/9781509302963_Microsoft%20Azure%20Essentials%20Fundamentals%20of%20Azure%202nd%20ed%20pdf.pdf , archived April 13, 2021
1017	U.S. Patent No. 11,799,874 to Lichtenstein
1018	McKay, “Guidelines for the Selection, Configuration, and Use of Transport Layer Security (TLS) Implementations,” NIST Special Publication 800-52, Rev. 2 (Aug. 2019) https://doi.org/10.6028/NIST.SP.800-52r2