

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

WIZ, INC.,
Petitioner,

v.

ORCA SECURITY LTD,
Patent Owner.

Case IPR2025-00093
Patent No. 11,582,257

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 '257 PATENT.....	8
	A. Challenged Claims.....	11
	B. Prosecution History of the '257 Patent.....	13
IX.	CLAIM CONSTRUCTION.....	13
X.	OVERVIEW OF THE PRIOR ART.....	14
	A. Keren (U.S. Patent No. 11,374,982, EX1005).....	14
	B. Engelberg (U.S. Publication No. 2022/0263855, EX1006).....	18
XI.	GROUND 1: CLAIMS 1-20 WERE OBVIOUS OVER KEREN.....	19
	A. Independent Claims 1, 11, and 20.....	20
	1. Preambles.....	20
	2. Element 1.i.....	26
	3. Elements 1.1, 11.1, and 20.1.....	26
	4. Elements 1.2, 11.2, and 20.2.....	36
	5. Elements 1.3, 11.3, and 20.3.....	38
	6. Elements 1.4, 11.4, and 20.4.....	40

7.	Elements 1.5, 11.5, and 20.5.....	53
8.	Elements 1.6, 11.6, and 20.6.....	59
9.	Elements 1.7, 11.7, and 20.7.....	65
B.	Dependent Claims	71
1.	Claims 2-3 and 12-13.....	71
2.	Claims 4 and 14.....	73
3.	Claims 5 and 15.....	74
4.	Claims 6 and 16.....	75
5.	Claims 7 and 17.....	76
6.	Claims 8 and 18.....	77
7.	Claims 9 and 19.....	78
8.	Claim 10.....	79
XII.	GROUND 2: CLAIMS 1-20 WERE OBVIOUS OVER KEREN AND ENGELBERG.....	83
A.	Reasons to Combine Keren and Engelberg.....	83
B.	Claims 1, 11, and 20.....	88
C.	Claim 10	91
XIII.	CONCLUDING STATEMENTS.....	92
XIV.	APPENDIX A – MATERIALS CITED.....	94

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. 11,582,257 (the “’257 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. 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 '257 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 '257 patent and the '257 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 '257 patent issued from U.S. Application No. 17/716,491 ("the '491 application"), filed April 8, 2022. EX1001, Face. The '491 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.

VIII. OVERVIEW OF THE ’257 PATENT

23. The ’257 patent is entitled “Prioritizing Internet-Accessible Workloads for Cyber Security.” EX1001, Face. The ’257 patent discloses techniques “for securing cloud infrastructure to help meet compliance mandates, without disrupting business operations in live environments.” EX1001, 1:15-18. As part of these techniques, the ’257 patent describes a cloud-based cybersecurity system “for

assessing internet exposure of a cloud-based workload,” as illustrated in Figure 6, reproduced below. *Id.*, Abstract, 33:46-61.

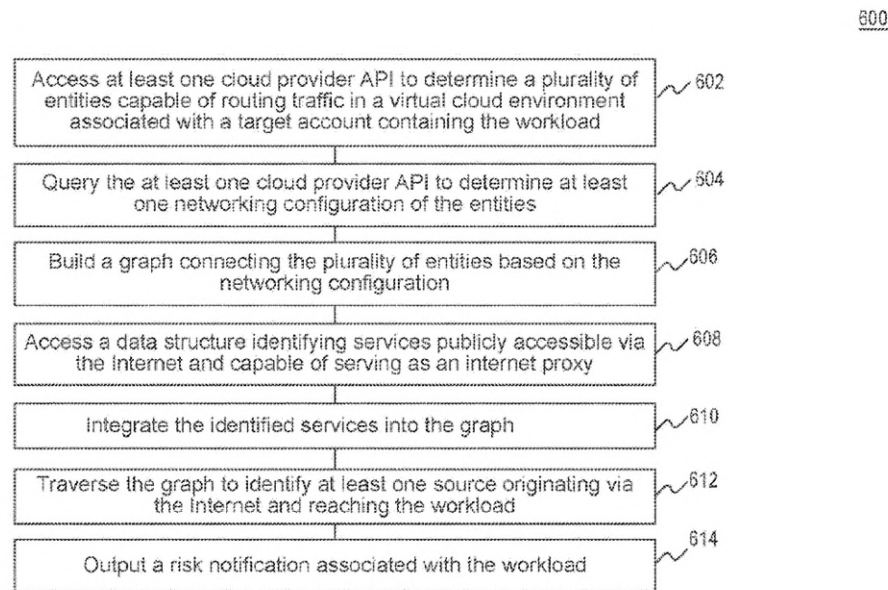


FIG. 6

24. Embodiments of the disclosed system are “configured to access at least one cloud provider API to determine a plurality of entities capable of routing and/or filtering traffic in a virtual cloud environment associated with a target account containing [a] workload.” EX1001, 33:49-55; *see also id.*, 33:62-34:6, 37:53-59, Fig. 1 (illustrating cloud infrastructure 106 containing “examples of a workload, such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D,

keystores 113A-113D, or load balancer 115”). The system is also configured to “query the at least one cloud provider API to determine at least one networking configuration of the entities.” *Id.*, 35:19-22; *see also id.*, 34:18-25 (system may “query devices and systems capable of routing and filtering traffic (e.g., load balancer 115, routers, switches, firewalls, security groups, API gateways, and proxies) using an API provided through a cloud service provider’s system to determine network configurations”), 37:60-65.

25. The system is further “configured to build a graph connecting the plurality of entities based on the networking configuration” to “show[] the relation between variable quantities (e.g., entities).” *Id.*, 36:14-19; *see also id.*, 36:22-29 (system “may construct a graph or diagram linking the plurality of entities based on the networking configuration” and may “include[] a data structure sequentially connecting entities”), 36:44-48 (graph may consist of linked nodes representing individual entities), 37:66-38:4. The system may also “access a data structure identifying services publicly accessible via the Internet and capable of serving as an internet proxy,” such as “a system or router that provides a gateway between users and the internet,” and “integrate the identified services into the graph.” *Id.*, 36:60-37:4; *see also id.*, 37:8-11 (“For example, scanning system 101 may integrate

information on services available in each node into the data structure, associated with each node in the data structure.”), 38:5-14.

26. Finally, the cybersecurity system is “configured to traverse the graph to identify at least one source originating via the Internet and reaching the workload” and further “output a risk notification associated with the workload.” *Id.*, 37:12-15, 37:32-34; *see also id.*, 38:15-25.

A. Challenged Claims

27. The '257 patent has 20 claims, with claims 1, 11, 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 cloud-based cybersecurity system for assessing internet exposure of a cloud-based workload, the system comprising:

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

[1.1] access at least one cloud provider API to determine a plurality of entities capable of routing traffic in a virtual cloud environment associated with a target account containing the workload;

[1.2] query the at least one cloud provider API to determine at least one networking configuration of the entities;

[1.3] build a graph connecting the plurality of entities based on the networking configuration;

- [1.4] access a data structure identifying services publicly accessible via the Internet and capable of serving as an internet proxy;
- [1.5] integrate the identified services into the graph;
- [1.6] traverse the graph to identify at least one source originating via the Internet and reaching the workload; and
- [1.7] output a risk notification associated with the workload.

28. Furthermore, independent claim 11 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 specifying the plurality of entities includes a virtual network appliance, which can be a load balancer, a firewall, a proxy, or a router (claims 2-3 and 12-13); specifying the network configuration can be a routing configuration, proxy configuration, load balancing configuration, firewall configuration, or VPN configuration (claims 4 and 14); specifying the graph includes a data structure sequentially connecting entities (claims 5 and 15), directional vectors indicating directions of dataflow (claims 6 and 16), port numbers associated with the workload (claims 8 and 18), or a path from the at least one source to the workload (claims 9 and 19); building the graph by identifying individual entities as nodes and connecting the nodes (claims 7 and 17); and specifying the risk notification includes one or more resolution recommendations (claim 10).

B. Prosecution History of the '257 Patent

29. I understand that the '491 application (i.e., the patent application that led to the '257 patent) never received a rejection under §102 or §103. I understand that the first office action rejected some of the claims under §101 as being “drawn to a computer program per se” and allowed the remaining claims. EX1004, 406-07. I understand that, in order to secure allowance, the rejected claims were amended to recite that the system includes “at least one hardware processor configured to” perform the claimed steps. *Id.*, 216, 222. Finally, I understand that in allowing the claims, the Examiner identified four references as the closest prior art but found that “none of the prior art of record alone or in combination teaches *accessing a data structure identifying services publicly accessible via the internet and capable of serving as an internet proxy; integrating the identified services into the graph.*” *Id.*, 407-08.

IX. CLAIM CONSTRUCTION

30. 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. I understand that this is the same claim construction approach used in district court litigation. I have followed this approach in my analysis below.

X. OVERVIEW OF THE PRIOR ART

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

31. Keren is a patent that relates to cybersecurity methods and systems “for automated generation of network analysis insights” of objects based 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. 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, 9:6-15 (security objects may further include other computing resources such as load balancers, virtual machines, and containers).

32. The cloud environment is configured to connect with a cybersecurity system 150 and a graph database 160. *Id.*, 4:43-47, Figs. 1A-1B. “The cybersecurity 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. The graph database 160 “is a storage or memory component, device, system, or the like, configured to provide one or more functionalities relevant to storage of graph-related data.” *Id.*, 7:3-6; *see also id.*, 7:6-11 (graph database stores “graph-related data features,” such as “raw data, graphs, graph edges, graph vertices, graph schemas, and the like”).

33. To perform its analysis, Keren teaches that the cybersecurity system generates security graphs that detail the network topology of the security objects based in the cloud environment. *See id.*, Abstract, 10:27-30, Fig. 2. Security graphs are constructed by querying an API provided by the cloud environment to identify the security objects contained within the network 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.*, 10:27-30, 10:49-57. By identifying the security objects and their relationships, the system constructs a security graph of the network, which can be in various formats, such as a table, chart, a list of objects, or as a “node and link” graph, in which objects are represented as nodes connected by links that represent the objects’ network relationships with one another. *Id.*, 7:18-24, 10:33-48, Figs. 4A (visual node and link graph), 4B (list of objects illustrating network route).

34. Keren further teaches that the network properties of an object, as determined by the collected security object data, may be integrated into the security graph by tagging security objects with data labels, tags, or the like. *Id.*, 11:13-27 (data tags may describe “connection source and destination, connection type, connection direction, connection status, connection protocol, and the like”), 13:6-25 (data tags may provide “descriptions of an object’s open ports and counts thereof, descriptions of an object’s network address or addresses, descriptions of protocols relevant to an object”).

35. From this information, Keren teaches that additional properties can be determined and populated into the graph. For example, Keren teaches that a “reachability property” of each security object can be calculated. *Id.*, Abstract, 11:61-12:10, Figs. 3A-3B. A reachability property “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.*, 13:26-34; *see also id.*, 15:15-24 (“Reachable properties are object properties describing if, and how, a given path element is reachable through the selected path, and specifically, from an external/internal network.”). Keren teaches calculating a reachability property for a security object by traversing a path in the security graph to the security object and determining the minimal set of network accessibility properties needed to reach that security object. *Id.*, 13:66-14:5 (“[P]aths can be identified by traversing the security graph. The traversal can start or end at objects that are connected to an external network (the internet).”), 15:25-33, 15:44-58. By doing so, the system can identify the type of network traffic originating from the internet to which the security object may be exposed. *See id.*, 19:66-20:12, Fig. 5.

36. “[B]ased on the known and calculated properties” of the security objects in the security graph, Keren explains that “security insights may be generated.” *Id.*, 12:11-12. Security insights provide natural-language descriptions

of aspects of the security graph, including descriptions of risky network configurations. *Id.*, 12:12-14, 12:45-13:5. For example, a security insight may notify a user that ““VM thirty normally connects to load balancer twenty and firewall eight but is currently only connected to load balancer twenty”” or ““objects E, a database, and F, an administrator interface, are currently exposed to external networks.”” *Id.*, 12:57-61, 12:66-13:5.

B. Engelberg (U.S. Publication No. 2022/0263855, EX1006)

37. Engelberg is a patent publication related to cyber risk mitigation. EX1006 (Engelberg), Title, Abstract; EX1007 (Engelberg Provisional), Title, Abstract.¹ Engelberg recognizes that “[c]omputer networks are susceptible to attack by malicious users (e.g., hackers)” and “[t]o defend against such attacks, enterprises use security systems to monitor occurrences of potentially adverse events occurring

¹ In my discussion of Engelberg here and in my analysis below (§XII), I provide parallel citations to Provisional Application No. 63/150,664 (EX1007), which I understand is the provisional application to which Engelberg claims priority. EX1006 (Engelberg), ¶1. The parallel citations show where support for the cited portions of Engelberg can be found in the equivalent disclosure of the provisional.

within a network, and alert security personnel to such occurrences.” EX1006 (Engelberg), ¶¶2-3; *see also id.*, ¶26; EX1007 (Engelberg Provisional), ¶¶1-2, 24. To aid security personnel in managing computer networks, Engelberg teaches “recommending remedial actions for cyber security” as part of identifying risks associated with the network. EX1006 (Engelberg), ¶30; *see also id.*, ¶59 (“In this manner, the AgiSec platform can convey information to enable users (e.g., security teams) to understand immediate remediation actions corresponding to each issue.”); EX1007 (Engelberg Provisional), ¶¶28, 57.

XI. GROUND 1: CLAIMS 1-20 WERE OBVIOUS OVER KEREN

38. As I explain below, it is my opinion that claims 1-20 were obvious over Keren. Keren describes a cybersecurity system that teaches, or at least renders obvious, each of the limitations recited in claims 1-20. Specifically, Keren teaches cybersecurity systems and methods for assessing the internet exposure of a cloud-based workload by accessing and querying a cloud provider API to identify security objects, including those capable of routing traffic in a virtual cloud environment, and determining the networking configuration of those objects. *Supra*, §X.A (Keren summary). Keren also teaches that a security graph is built based on the determined networking configuration, and properties associated with the security objects, including the services provided by those objects, are integrated into the graph. Based on this network information, the security graph is traversed to assess whether and

how a workload is reachable from the internet. *Supra*, §X.A; *see also infra*, §§XI.A-B (detailed claim analysis).

A. Independent Claims 1, 11, and 20

39. Claims 1, 11, and 20 recite a system, method, and computer-readable medium, respectively, while reciting analogous limitations, which are discussed together below.

1. Preambles

40. Claim 1 recites a “cloud-based cybersecurity system for assessing internet exposure of a cloud-based workload, the system comprising.” Claim 11 recites a “method for assessing internet exposure of a cloud-based workload, 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.” As I explain below, Keren teaches the preambles of claims 1, 11, and 20.

41. First, Keren describes a “cloud-based cybersecurity system” and corresponding methods. Specifically, Keren teaches methods and systems for assessing “security objects” 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 that include “multiple security objects”), Figs. 1A-1B, claims

1, 11. Keren describes using a cybersecurity system to assess the security objects and automatically generate network analysis insights of the cloud environment. *See id.*, 1:15-18, 2:34-55 (“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”), 4:43-55 (“[T]he cloud environment may be configured to connect, via a network 108, with a cyber-security system 150 and a graph database (graph DB) 160 for one or more purposes including, without limitation, those described hereinbelow.”), 21:34-37. The cybersecurity system may be cloud-based and configured to implement assessment methods that include, for example, generating security graphs, determining relationships between security objects, generating insights describing network configurations or events, and determining reachable properties of security objects. *See id.*, 3:54-62, 6:33-37, 6:47-56 (cybersecurity system provides “network analysis functionalities including, without limitation, network analysis, traffic analysis, object querying, graph generation, and the like”), 6:57-7:2 (“cyber-security system 150 may be included within any of the various elements of the network system 102, including the cloud environment 103, the various cloud platforms 104, and subparts thereof, and the network 108”), 9:22-25, 12:11-19, 13:26-34, 14:31-33, Figs. 2-3B, claim 11.

42. Keren further teaches that the cloud-based cybersecurity system and corresponding methods are “for assessing internet exposure of a cloud-based workload,” as recited.

43. First, the security objects based within the cloud environment include “a cloud-based workload.” The ’257 patent broadly describes workloads as “systems, devices, or resources in a network (or available via a network) such as a cloud infrastructure.” EX1001, 16:56-59; *see also id.*, 39:44-46 (“A workload, as used herein, may refer to a specific application, service, capability, or a specific amount of work that can be run on a cloud resource, system, or infrastructure.”). Specific examples of workloads described by the ’257 patent include, for instance, virtual machines, databases, storage servers, load balancers, containers, and applications. *Id.*, 16:60-63, 39:44-50, 54:59-55:3, Fig. 1.

44. Keren teaches that the security objects based within the cloud environment contain the same types of workloads as those described by the ’257 patent. For example, similar to the ’257 patent, Keren generally explains that the security objects 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. These security objects may include “computing objects,” which “are objects configured to provide one or more data

management or processing functionalities,” such as “dedicated processing systems, file servers, databases, and the like.” EX1005 (Keren), 5:32-44, Fig. 5. Other security objects identified by Keren include virtual machines, containers, and load balancers. EX1005 (Keren), 6:10-19, 7:46-56, 9:9-15, 10:20-26, 14:59-61, 20:34-42, Figs. 4A-4B.

45. Second, Keren’s system and method further assess the “internet exposure” of a workload. As discussed in more detail below (*infra*, §§XI.A.5, XI.A.8), Keren teaches that the cybersecurity system is configured to construct “[a] security graph describing the various objects included in, and adjacent to, a network, as well as the relationships between such objects.” EX1005 (Keren), 10:27-30; *see also id.*, 11:11-23 (stored security object relationships include “connection source and destination, connection type, connection direction, connection status, connection protocol, and the like”), 13:35-47, 16:54-67 (security graph provides information representing connections between objects). The security graph may be used to determine reachable properties of security objects. *Id.*, 13:26-14:30. “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,” where “[e]xternal means outside of the cloud environment of an organization” such as the internet. *Id.*, 13:28-34, 13:46-14:1 (describing

identification of network paths that “start or end at objects that are connected to an external network (the internet)”). By determining these reachable properties, Keren’s system can assess the ways in which a security object, such as a “workload” that I described above, may be accessible from the internet (and thus exposed to traffic originating from the internet). *See id.*, 12:62-13:5 (generating insights describing, e.g., “third-party networks A, B, and C currently have access to the internal network” or “objects E, a database, and F, an administrator interface, are currently exposed to external networks”), 15:44-58 (reachable properties may include “object network access control lists (NACLs)” that may “describe the properties of an object with respect to the object’s internet accessibility”), 20:34-42 (“reachable properties may, for various scopes, such as the internet,” include “whether a virtual machine (VM) or resource is accessible from a given scope”), 19:34-20:12 (determining reachable property “accessibleFromInternet=true” for database object), Figs. 4B (determining network routing between a container and the internet), 5 (determining network routing between a database and a public proxy).

46. Finally, Keren 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 used to perform the operations of claim 20 “may be

configured to execute one or more instructions, methods, processes, and the like.” *Id.*, 6:51-56, Figs. 2-3B; *infra*, §§XI.A.3-9. For example, Keren teaches the cybersecurity system includes a “processing circuitry 610 coupled to ... a storage 430.” EX1005 (Keren), 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.

47. It is therefore my opinion that Keren teaches the preambles of claims 1, 11, and 20.

2. Element 1.i

48. Keren teaches this element, which recites “at least one hardware processor configured to....” As I discussed above, regarding claim 20, Keren teaches that the cybersecurity system includes a hardware 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 hardware component, such as a processor), 21:16-33 (processing circuitry executes stored instructions to perform methods of Keren).

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

3. Elements 1.1, 11.1, and 20.1

50. These elements recite “[access/accessing] at least one cloud provider API to determine a plurality of entities capable of routing traffic in a virtual cloud environment associated with a target account containing [the/a] workload.” As I explain below, Keren teaches these elements.

51. First, Keren describes “a plurality of entities capable of routing traffic in a virtual cloud environment,” as recited. As I described above, Keren teaches a cybersecurity system that provides network analysis of security objects based within a cloud computing environment, which is a virtual cloud environment. EX1005 (Keren), 5:4-13 (cloud platforms may be implemented as “a virtual network, providing for interconnection of various virtual systems” and may “simulate or

emulate ... a local area network, a wide area network, the Internet, the World-Wide Web (WWW), and the like”), 8:29-53 (same); *supra*, §XI.A.1. Keren explains that the security objects may also include the same types of entities that are capable of routing traffic in the virtual cloud environment as identified in the ’257 patent. For example, the ’257 patent explains that “devices and systems capable of routing and filtering traffic” include load balancers, routers, switches, firewalls, security groups, API Gateways, and proxies. EX1001, 34:18-21; *see also id.*, 34:36-51 (entities capable of routing traffic include virtual network appliances such as “a firewall, WAN optimizer, application delivery controllers, routers, load balancers, IDS/IPS, proxies”), 34:51-53 (same), claims 2-3, 12-13. Like the ’257 patent, Keren teaches the security objects may include “network objects,” which “are objects configured to provide one or more traffic-management functionalities.” EX1005 (Keren), 5:32-37. “Examples of network objects include, without limitation, proxies, firewalls, routing tables, and the like.” *Id.*, 5:38-39. Other security objects providing traffic-management functionalities described by Keren include gateways, load balancers, and security groups. *Id.*, 5:66-6:4 (“[S]ecurity objects may include virtual entities, devices, and the like, to process layer-7 (application layer) traffic, such as ... Amazon Load Balancer® (ALB) layer seven services and applications.”), 6:14-19 (security objects include “firewalls, network interface cards, proxies, gateways ...

hubs, [and] virtual private networks (VPNs)”), 9:9-15 (additional security object examples include “peering connections, load balancers, [and] route tables”), 10:14-20 (querying for firewall object data), Figs. 4A (illustrating proxy, gateway, and firewall objects), 4B (illustrating network interfaces and proxies), 5 (illustrating network path from database 510 to network interface 520, security group 530, and public proxy 540).

52. Further, Keren teaches “accessing at least one cloud provider API” to determine the plurality of entities in the virtual cloud environment. Specifically, Keren explains that, to “construct[] security graphs, including network and computing objects,” “security objects are identified, and security object data is collected.” EX1005 (Keren), 9:21-27, Fig. 2. The “security objects may be identified by querying a cloud platform,” which can “include generation of one or more queries through an application programming interface (API).” *Id.*, 9:27-44 (“Through the API, security objects’ identities and description data are provided in response to such API queries.”); *see also id.*, 9:56-10:10 (security object data, including “object types, object names or unique identifiers (IDs),” collected through “generation and transmission of one or more API queries”), 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.”),

10:14-26 (discussing other examples of querying APIs for security object data). Keren explains that the API includes “platform-specific APIs,” such as those provided by “Amazon AWS®, Microsoft Azure®, and the like,” which a POSA would have understood to be a cloud provider API. *Id.*, 9:47-55.

53. Finally, Keren teaches that the plurality of entities in the virtual cloud environment is “associated with a target account containing [the/a] workload.” As I describe more below, a POSA would have understood Keren as teaching this element in two ways. First, Keren teaches that the security objects, which include workloads and a plurality of entities, may be provided on a service basis through a commercially-available cloud provider, which a POSA would have understood would be accessed and managed through a subscriber account. Second, Keren teaches that individual user accounts may be associated with the cloud environment, which a POSA would have understood allowed individual users access to particular workloads (e.g., a particular virtual machine) located within the cloud environment.

54. First, a POSA would have understood that the security objects taught by Keren would have been associated with a subscriber/owner account (i.e., a target account). As reproduced below, Keren teaches that the cloud environment includes “a number of cloud platforms” that each contain “multiple security objects,” which, as I discussed above, include both a plurality of entities (e.g., firewalls, proxies,

gateways) and workloads (e.g., databases, containers, virtual machines). EX1005 (Keren), 4:31-43, 5:20-6:19; *supra*, §XI.A.1.

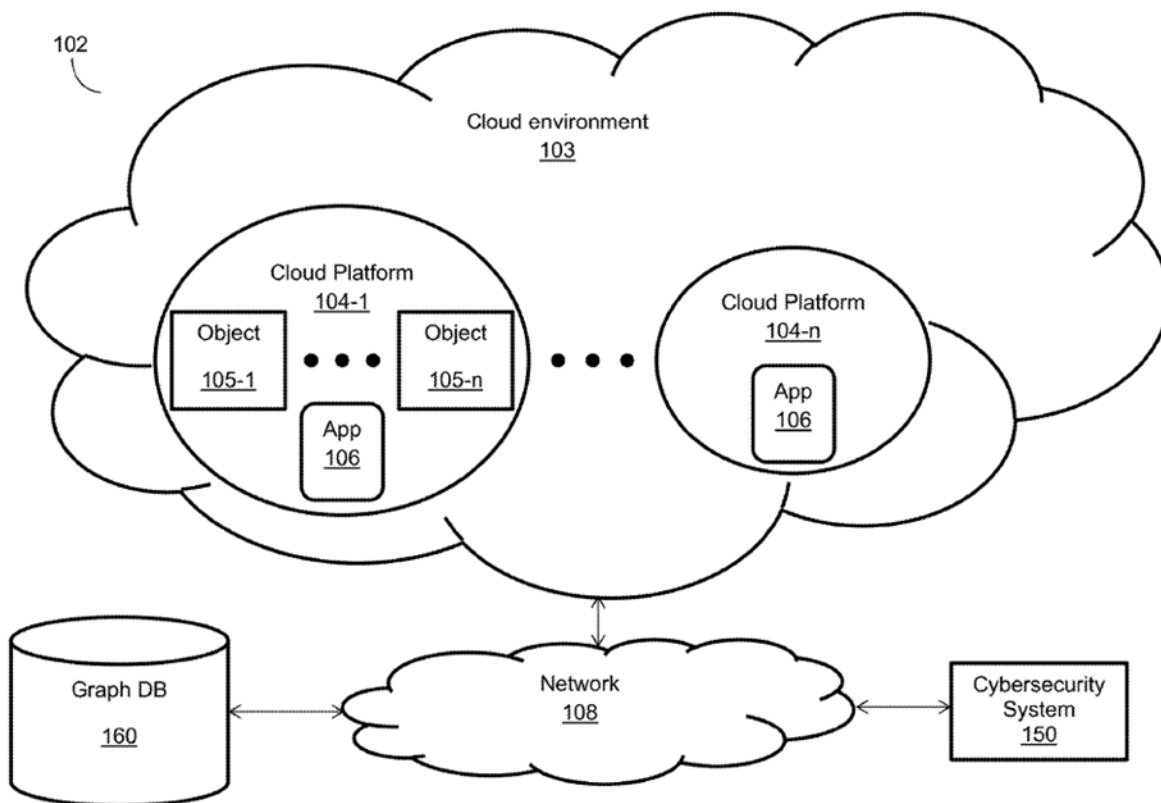


FIG. 1A

EX1005 (Keren), Fig. 1A.

55. As Keren explains, a cloud platform “is a platform, architecture, or other, like, configuration providing for connectivity of the various objects” within the platform and “may be a commercially-available cloud system, provided on a service basis, such as ... Amazon AWS®, Microsoft Azure®, and the like.” EX1005 (Keren), 4:56-65; *see also id.*, 4:32-35 (cloud environment “represents an organization’s cloud-based resources, and the various connections between such

resources”). It was commonly understood that when an entity, such as a business or organization, sought to use the services of commercially-available cloud providers, like Amazon AWS or Microsoft Azure, as a cloud platform, a subscriber account (or administrator/owner account) was required. *E.g.*, EX1008 (Microsoft Azure Fundamentals),² vii (“Microsoft Azure is Microsoft’s cloud computing platform, providing a wide variety of services you can use without purchasing and provisioning your own hardware.”), 1 (requiring an Azure account to subscribe to and access services), 22-23 (describing subscription account tiers). The subscriber account allowed the account owner to purchase resources, create virtual workloads and entities, and manage their network connections. *See, e.g., id.*, 2 (“Cloud

² EX1008 is an 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.

environments provide an online portal experience, making it easy for users to manage compute, storage, network, and application resources.”), 5-6 (Azure Resource Manager to deploy and manage, e.g., VMs, databases, virtual networks, and storage), 11 (Azure portal), 74-78 (managing load balancers, network interfaces, security groups). Thus, given Keren’s teachings that a cloud platform may be provided on a service basis by commercially-available cloud providers, a POSA would have understood that the cloud platform, which contains the security objects (i.e., plurality of entities and workloads), would be associated with a target account containing the workload, as recited. This understanding is consistent with the ’257 patent. *See, e.g.*, EX1001, 7:57-67 (“[C]loud infrastructure 106 may comprise devices and systems that are part of Amazon Web Services, Microsoft Azure, Google Cloud Platform, IBM Cloud, Alibaba Cloud, or any other cloud platform provider.”), 16:17-40 (“A source account, as used herein, may refer to a domain, a location on a network server, or anything used to access system resources.”), 34:33-37 (“A target account containing the workload, as used herein, may refer to a location on a network server selected as an aim of connection....”).

56. Second, Keren also teaches that the security objects in the cloud environment may be associated with a specific user account containing a workload (another type of target account). For example, as reproduced below, Fig. 4A

illustrates a user account that is connected to proxy peering, gateway, and firewall objects, which—as I discussed above—are capable of routing traffic in the virtual cloud environment.

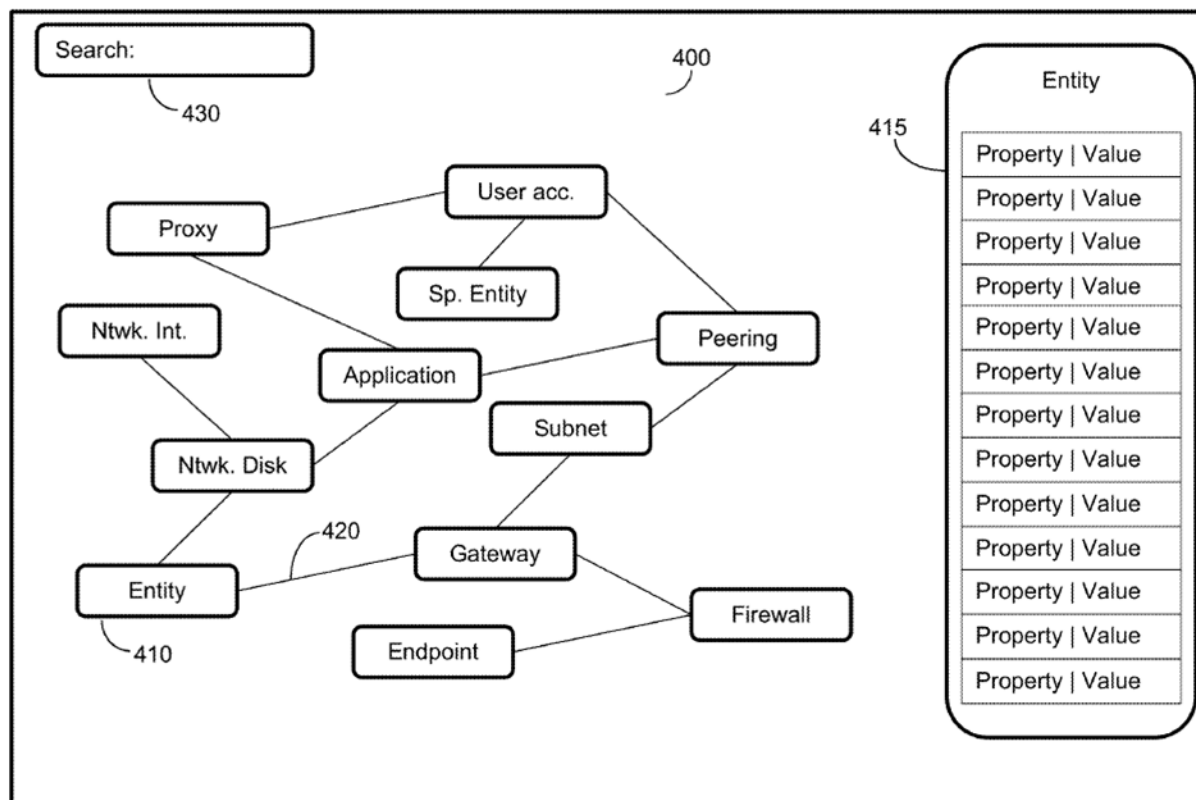


FIG. 4A

EX1005 (Keren), Fig. 4A.³

³ I note that while Fig. 4A lists only two nodes as an “Entity” in the illustrated security graph, a POSA would not have understood the claimed entities to be limited to only those nodes. Indeed, Fig. 4A illustrates nodes that are consistent with the plurality of entities described by the ’257 patent (as I noted above), such as a firewall, a proxy, and a gateway, for example. *See* EX1001, 34:18-21 (plurality of entities include load balancers, routers, switches, firewalls, security groups, API gateways, and proxies), 34:36-53. Moreover, Keren uses the term “entity” interchangeably with “security object” to refer to particular nodes of interest, and a POSA would have understood the “Entity” labeled in Fig. 4A represented a generic reference to the various types of security objects described in Keren. *See, e.g.*, EX1005 (Keren), 2:42-53 (describing goal of “provid[ing] network information which describes the role of an individual entity within such a network or cloud environment), 11:61-12:10 (“Each node, corresponding to each entity, stored in the database may include a recording describing its properties, including a “reachability property [that] defines if and how an entity can be

57. A POSA would have understood, or at least found it obvious, that a user account, such as that shown in Fig. 4A, represented a point of authorized access for a particular user (e.g., an individual within the organization that owns the subscriber account) to use computing resources (e.g., a given virtual machine, as may be illustrated, for example, by the endpoint in Fig. 4A) maintained in the virtual cloud environment. *See, e.g., id.*, 12:45-55 (detecting “re-connection of a user’s device to a subnet which the user is not permitted to access”), 15:44-58 (teaching use of “scope specific NACLs” for each security object, which “describe[e] object, path, or network accessibility properties specific to a given scope, where a given scope may be ... various given accounts”), 20:34-42 (reachable properties may describe “whether a virtual machine (VM) or resource is accessible from a given scope,” such as “other accounts”); EX1008 (Microsoft Azure Fundamentals), 6-7 (describing provision of “resource groups” to control user access to certain resources), 9-11, 15-21 (describing role-based access control features). In this manner, a POSA would have understood that an individual user account, which

reached from an external or internal network”), 13:62-65 (“[A]n object may be a virtual, physical, or logical entity.”).

would provide access to particular workloads and be associated with a plurality of entities within the virtual cloud environment, would represent “a target account” as recited by the claims. Again, this is consistent with the ’257 patent. *See, e.g.*, EX1001, 34:33-37 (“A target account containing the workload, as used herein, may refer to a location on a network server selected as an aim of connection....”); 39:67-40:3 (“A target account of the target system in this disclosure may refer to an account of a user of a cloud infrastructure (e.g., a cloud environment or a cloud storage environment).”).

58. It is therefore my opinion that Keren teaches all aspects of elements 1.1, 11.1, and 20.1.

4. Elements 1.2, 11.2, and 20.2

59. Keren teaches these elements, which recite “[query/querying] the at least one cloud provider API to determine at least one networking configuration of the entities.”

60. As I discussed above, Keren teaches querying a cloud provider API to determine security objects, such as the recited plurality of entities, in the cloud environment. *Supra*, §XI.A.3; *see also* EX1005 (Keren), 9:38-55 (identifying security objects includes querying cloud platforms using “an application programming interface (API)”), Fig. 2 (step S210). Keren teaches that the cloud provider API is also queried to determine additional data relating to the security

objects, including information relating to an object's network configuration, such as an object's network addresses, port configurations, or access configurations. *Id.*, 9:38-55 ("identification of security objects and collection of security object data ... includes querying each cloud platform" using an API), 9:56-10:10 (security object data includes network configuration information, such as "object network addresses," "object port configurations," "object [input/output] signal configurations," and "object security or access configurations"), Fig. 2 (step S210). For example, Keren describes an API query in which information relating to the networking configuration of a firewall is collected, including "the collection of firewall rules, collection of firewall event logs, collection of firewall port configurations, and the like." *Id.*, 10:14-20; *see also id.*, 19:62-20:5 (describing use of determined public proxy port configuration for network path analysis); *compare* EX1001, 35:35-38 ("[T]he networking configuration is at least one of a routing configuration, a proxy configuration, a load balancing configuration, a firewall configuration, or a VPN configuration.").

61. It is therefore my opinion that Keren teaches all aspects of elements 1.2, 11.2, and 20.2.

5. Elements 1.3, 11.3, and 20.3

62. These elements recite “[build/building] a graph connecting the plurality of entities based on the networking configuration.” As I explain below, Keren teaches these elements.

63. Keren teaches that after identifying and collecting data of the security objects, as previously described (*supra*, §§XI.A.3-4), the system constructs a security graph (i.e., builds a graph), which is “a data feature describing the various objects included in, and adjacent to, a network, as well as the relationships between such objects” based on the collected object data. EX1005 (Keren), 10:27-33, Fig. 2. The “security graph may be constructed in one or more various formats,” including “a table, chart, or other, nonvisual, data organization format, a list of objects,” or as a “visual ‘node and link’ graph.” *Id.*, 10:33-48; *see also id.*, Figs. 4A (illustrating an example security graph generated as a visual node-and-link graph (reproduced further below)), 4B (illustrating security graph list feature showing object-to-object network path).

64. Keren also teaches that the security graph connects the plurality of entities based on the determined network configuration of the entities. Specifically, Keren teaches determining “relationships between security objects,” which “are descriptions of the various connections between the [identified] security objects.” *Id.*, 10:49-52, Fig. 2 (step S230). Keren describes using a static analysis process that

analyzes the collected security object data, including the data collected on the network configuration of the entities, to determine the relationships between security objects. *Id.*, 10:58-64 (“[T]he static analysis may include analysis of object and protocol code and rules, based on simulated network operation, as collected at S210, to provide for identification of security object relationships based on security object configurations.”); *supra*, §XI.A.4. As one example of a static analysis determination, Keren explains:

“[D]ata collected from a firewall at S210 may specify, in the firewall’s port configurations, communication with a first device on a first port using a first protocol and connection with a second device on a second port using a second protocol. Further, according to the same example, the firewall object may include one or more instructions specifying transmission of a specific log file, via a third port, to a connected repository. According to the same example, security object relationships determined at S230 may include connections between the firewall and the first device, connections between the firewall and the second device, and connections between the firewall and the repository.”

EX1005 (Keren), 10:64-11:10.

65. Keren further teaches that the determined relationships are included in the constructed graphs. *Id.*, 11:11-13 (“Determination of security object relationships at S230 may further include updating the graph or graphs constructed at S220 to include the determined relationships.”); *see also id.*, 11:14-27. The determined relationships may be included, for example, as “links” that connect “nodes” representing security objects located within the environment. *Id.*, 11:28-37.

Shown below is an example of a node-and-link security graph visualization built from the determined relationships between the security objects:

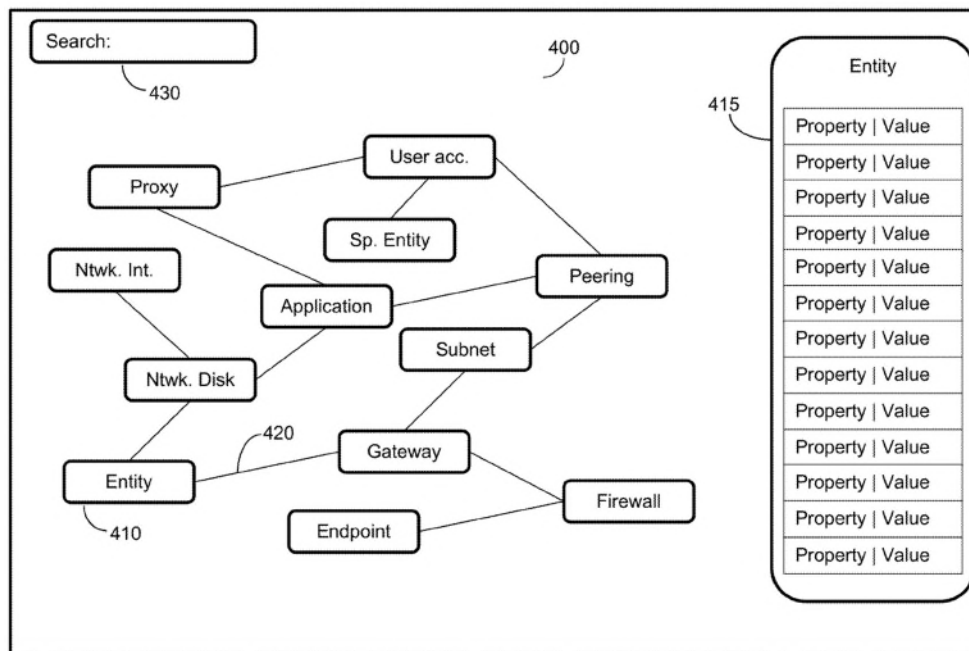


FIG. 4A

Id., Fig. 4A; *see also id.*, 16:41-67, Fig. 4B (showing object relationships in list format).

66. It is therefore my opinion that Keren teaches all aspects of elements 1.3, 11.3, and 20.3.

6. Elements 1.4, 11.4, and 20.4

67. These elements recite “[access/accessing] a data structure identifying services publicly accessible via the [I/i]nternet and capable of serving as an internet proxy.” As I explain below, Keren teaches these elements. First, I address how a

POSA would have understood Keren teaches, or at least would have found it obvious from Keren's teachings, that the cybersecurity system accesses a data structure that stores security object data that is used for generating and updating constructed security graphs. I then address how the stored security object data identifies services publicly accessible via the internet and capable of serving as an internet proxy.

68. First, as I described above, Keren teaches that the cybersecurity system identifies security objects and collects security object data that is then used to generate security graphs. EX1005 (Keren), 9:; *supra*, §§XI.A.2-4. In addition to using this collected security object data to determine the network relationships between the identified security objects and building a security graph connecting the objects based on those relationships (*see supra*, §XI.A.5), as I explained in more detail below in §XI.A.7, Keren also teaches that the security graph may be updated (or integrated) with recordings (via, for example, the use of labels or tags) that describe the properties of the security objects, which can be based on the collected security object data. EX1005 (Keren), 11:61-12:10, 13:6-25. A POSA would have understood that, to perform this analysis, the collected security object data would have been stored in a memory storage or database (i.e., a data structure) so that the cybersecurity system could access and process the security object data for integration into the security graph.

69. Keren suggests at least two ways in which the system can store and access collected security object data in a data structure. For example, as I explained above in §XI.A.1, Keren teaches that the cybersecurity system generally provides “network analysis functionalities,” which include functions relating to the querying and collection of security object data, network analysis, and graph generation. *Id.*, 6:47-51. To perform these functions, Keren explains that the cybersecurity system includes, among other things, “a storage 630.” *Id.*, 20:61-67, Fig. 6. A POSA would have understood Keren as teaching, or at a minimum rendering obvious, storing the security object data collected by the system in this storage so that the cybersecurity system could later access the data for subsequent analysis and integration into a security graph. *See, e.g., id.*, 9:26-29 (“[S]ecurity objects are identified, and security object data is collected.”).

70. In addition, Keren further teaches the use of a “graph database (graph DB),” which is “configured to provide one or more functionalities relevant to storage of graph-related data” and “store graph-related data features of one or more types or formats,” such as “raw data, graphs, graph edges, graph vertices, graph schemas, and the like.” EX1005 (Keren), 7:3-11; *see also id.*, 7:28-30, 7:67-8:3 (explaining graph DB “may be directly connected to, or realized as a component of, the graph analysis system 150”). Generated security graphs are stored in the graph database, which

Keren explains may be updated on an ongoing basis as collected security object data is processed and analyzed. *See, e.g., id.*, 11:11-17 (graphs “may be updated ... by associating one or more data labels, tags, or other, like, features with a graph entry for a security object”), 11:61-62, 15:36-3 (reachable properties may be populated into a constructed graph through labeling or tagging), 16:15-21 (graphs may be updated to include calculated properties). These updates include integration of tags that may include descriptions of object details determined by the collected security object data. *Id.*, 13:6-25 (tags include “object details, such as may be collected at S210”), Fig. 2. Based on these teachings, a POSA would have understood that the collected security object data would also be stored as “raw data” in the graph database (*see id.*, 7:3-11) that the cybersecurity system would access and process during the generation of security graphs, including during updates made to stored security graphs with object-specific tag information. In view of the above, a POSA thus would have understood Keren teaching, or at minimum rendering obvious, “[access/accessing] a data structure” as recited.

71. Second, a POSA would have understood that the stored security object data accessed by the system identifies “services publicly accessible via the internet and capable of serving as an internet proxy,” as recited. The ’257 patent explains that “[s]ervices capable of serving as an internet proxy ... may refer to a system or

router that provides a gateway between users and the internet” and “[a] proxy ... may refer to a device or system that translates traffic between networks and protocols.” EX1001, 35:8-18, 36:64-66. Consistent with the ’257 patent, the security objects taught by Keren include objects that provide services that are publicly accessible via the internet and capable of serving as an internet proxy (e.g., by acting as a gateway and translating traffic between users and the internet). For example, Keren explains that the security objects can include Nginx (an HTTP web server), web-access firewalls (WAFs), public proxies, and entities that “process layer-7 (application layer) traffic, such as objects relevant to Amazon AWS® layer seven services and applications, Amazon Load Balancer® (ALB) layer seven services and applications, Kubernetes ingress, and the like.” EX1005 (Keren), 5:56-6:4, Figs. 4B (illustrating internet-facing proxy for routing traffic to a container), 5 (illustrating public proxy 540 for routing internet traffic to a database). A POSA understood that these security objects provided management and routing services of internet traffic to and from resources within the network, and thus, provided “services that are publicly accessible via the internet and capable of serving as an internet proxy,” as recited. *See, e.g.*, EX1009 (ALB Overview) (Amazon Load

Balancer provides load balancing services of HTTP and HTTPS traffic);⁴ EX1010 (Kubernetes Ingress Overview) (Kubernetes Ingress provides routing services of HTTP traffic);⁵ EX1016 (Nginx) (“nginx (*‘engine x’*) is an HTTP web server, reverse proxy, content cache, load balancer, TCP/UDP proxy server, and mail proxy

⁴ EX1009 is a copy of “Application Load Balancer,” published by Amazon.com, Inc. and archived on February 20, 2021, by the Internet Archive’s “Wayback Machine” at <https://web.archive.org/web/20210220224806/https://aws.amazon.com/elasticloadbalancing/application-load-balancer/>.

⁵ EX1010 is a copy of “Ingress,” published by Kubernetes, archived on January 10, 2021, by the Internet Archive’s “Wayback Machine” at <https://web.archive.org/web/20210110103139/https://kubernetes.io/docs/concepts/services-networking/ingress/>.

server.”);⁶ *compare* EX1001, 14:19-25 (identifying web server Nginx as a type of service), 35:8-18, 36:64-66.

72. Keren also teaches that the security object data collected and stored by the system identifies these services. For example, as I previously discussed, security object data may include object types, object names, rules, event logs, object port configurations, protocols, and internal libraries of the security objects. EX1005 (Keren), 6:5-9 (“The security objects 105 may be configured to include one or more communication ports, where the included communication ports provide for connection of various objects according to one or more protocols, and at different communication layers of the OSI model.”), 9:58-66, 10:11-26 (data collected includes rules, event logs, port configurations, and internal data resources/libraries); *supra*, §XI.A.4. A POSA would have understood that this collected data would have identified services that were publicly accessible via the internet and capable of serving as an internet proxy. As an example, collecting object names of security

⁶ EX1016 is a copy of “nginx,” archived on April 10, 2021, by the Internet Archive’s “Wayback Machine” at <http://web.archive.org/web/20210410094516/https://nginx.org/en/>.

objects would indicate whether a given object was a Nginx web server, an Amazon Load Balancer, or a Kubernetes Ingress. EX1005 (Keren), 5:56-6:4. These object names would identify that the services provided by these objects were publicly accessible via the internet and capable of serving as an internet proxy because, as I noted above, these objects were known to service internet traffic for a network. See EX1009 (ALB Overview); EX1010 (Kubernetes Ingress Overview); EX1016 (Nginx); *compare* EX1001, 14:19-25 (identifying web server Nginx as a type of service), 36:64-66 (internet proxy services refer to systems providing an internet gateway), 35:8-18 (a proxy translates traffic between networks or protocols).

73. Similarly, collecting data on object types would indicate whether an object was a web-access firewall (also known as a web-application firewall) or a public proxy, which were common resources that provided services for managing internet traffic. See EX1005 (Keren), 5:56-6:4, 19:56-20:12, Fig. 5; EX1011 (Proxy Servers Overview), 2-10 (describing internet traffic management services provided by proxies, including public proxies));⁷ EX1012 (Fortinet Proxy Server), 1 (“A

⁷EX1011 is a copy of “What is a Proxy Server? In English, Please,” published by freeCodeCamp, archived on April 21, 2021, by the Internet Archive’s

proxy server provides a gateway between users and the internet.”), 4 (“A public proxy is accessible by anyone free of charge. It works by giving users access to its IP address, hiding their identity as they visit sites.”);⁸ EX1009 (ALB Overview), 5 (“AWS WAF is a web application firewall that helps protect your web applications from common web exploits that could affect application availability, compromise security, or consume excessive resources.”); EX1013 (Rapid7 WAF), 1 (“A web application firewall (WAF) helps protect a company’s web applications by

“Wayback Machine” at

<https://web.archive.org/web/20210421111614/https://www.freecodecamp.org/news/what-is-a-proxy-server-in-english-please/>.

⁸EX1012 is a copy of “Proxy Server,” published by Fortinet, archived on April 24, 2021, by the Internet Archive’s “Wayback Machine” at <https://web.archive.org/web/20210424234713/https://www.fortinet.com/resources/cyberglossary/proxy-server>.

inspecting and filtering traffic between each web application and the internet.”).⁹ An object’s port configurations and the protocols handled by the object would also identify whether the services provided by the object were publicly accessible via the internet and capable of serving as an internet proxy. *See, e.g.*, EX1014 (Port Security), 1 (identifying open ports can identify “what kind of services (HTTP, SMTP, FTP, DNS, SSH, Telnet, or VCN) are running” on the system or server);¹⁰ EX1015 (Ports In Networking), 1 (“Each port is associated with a specific process

⁹ EX1013 is a copy of “Web Application Firewall,” published by Rapid7, archived on April 10, 2021, by the Internet Archive’s “Wayback Machine” at <https://web.archive.org/web/20210410115503/https://www.rapid7.com/fundamentals/web-application-firewalls/>.

¹⁰ EX1014 is a copy of “Ports and Its Security,” published by W3schools, archived on April 12, 2021, by the Internet Archive’s “Wayback Machine” at <https://web.archive.org/web/20210412193421/https://www.w3schools.in/cyber-security/ports-and-its-security/>.

or service.”);¹¹ *compare* EX1001, 14:19-25 (identifying SSH network protocol as a type of service), 36:51-55 (recognizing that port numbers can “identify a specific process to which an internet or other network message is to be forwarded when it arrives at a server and/or which services are running on a server”). For instance, a port configuration in which port 80 is in use would identify that the object services internet traffic. EX1014 (Port Security), 1 (logical connections are assigned specific numbers that are end-points that determine which service is being used e.g., “[w]henever a data communication is handled by TCP/IP as client-server architecture, it uses the TCP port 80”), 2 (port number 80 supports HTTP traffic); EX1015 (Ports In Networking), 2 (HTTP messages go to port 80), 5 (listing other commonly used ports).

¹¹EX1015 is a copy of “What is a computer port? Ports in networking,” published by Cloudflare, Inc., archived on March 3, 2021, by the Internet Archive’s “Wayback Machine” at <http://web.archive.org/web/20210303124521/https://www.cloudflare.com/learning/network-layer/what-is-a-computer-port/>.

74. Additionally, Keren teaches that security object relationships and connectivity may be determined by monitoring traffic of a network in use or other logging methods. EX1005 (Keren), 10:14-20 (security object data may include “collection of firewall event logs”), 11:56-60 (“[O]bject relationships may be determined at S230 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.”). The active logging methods, which a POSA would have understood may include, for example, proxy logs, firewall logs, web-access firewall (WAF) logs, NAT logs, CloudTrail logs, web server logs, load balancer logs, or logs from other services provided by a third-party (e.g., Amazon Web Services, Microsoft Azure, Google Cloud, Alibaba Cloud, IBM Cloud, Oracle, Salesforce, SAP, or similar services), would identify services publicly accessible via the internet and capable of serving as an internet proxy, as recited. *See, e.g.*, EX1017, 6 (Using Proxy Logs) (discussing proxies containing proxy logs, which log network events such as denying requested traffic);¹² EX1018, 1 (Apache Web

¹²EX1017 is a copy of “Notes from the Field. Using Proxy Logs to Detect Suspicious Activity – Interview with Aaron Hambleton,” published by

Server) (describing log file information of Apache web server, including log records of all HTTP requests processed by the server);¹³ EX1019 (AWS WAF Logging), 2-5 (describing log field information captured by AWS WAF logs).¹⁴ Thus, the stored

SecurityHQ, archived on April 16, 2021, by the Internet Archive’s “Wayback Machine” at

<https://web.archive.org/web/20210416073813/https://www.securityhq.com/blog/using-proxy-logs-to-detect-suspicious-activity/>.

¹³ EX1018 is a copy of “Apache HTTP Server Version 2.4, Log Files,” published by Apache HTTP Server Project, archived on March 3, 2021, by the Internet Archive’s “Wayback Machine” at <http://web.archive.org/web/20210303195600/https://httpd.apache.org/docs/2.4/logs.html>.

¹⁴ EX1019 is a copy of “Logging web ACL traffic information,” published by Amazon AWS, archived on November 11, 2020, by the Internet Archive’s “Wayback Machine” at <http://web.archive.org/web/20201111183928/https://docs.aws.amazon.com/waf/latest/developerguide/logging.html>.

security object data identifies services publicly accessible via the internet and capable of serving as an internet proxy as recited.

75. It is therefore my opinion that Keren teaches all aspects of elements 1.4, 11.4, and 20.4.

7. Elements 1.5, 11.5, and 20.5

76. These elements recite “[integrate/integrating] the identified services into the graph.” As I explain below, Keren teaches these elements.

77. Keren teaches that the security object data, including the data identifying the services discussed above and accessed in data structures such as Keren’s storage 630 and/or graph DB 160 (*supra*, §XI.A.6), is integrated into the constructed security graph. Specifically, Keren explains that the security objects contained within the generated security graph may “include a recording describing its properties,” such as “‘known network properties’ (information that can be retrieved from querying the entity or the cloud platform)”—i.e., the collected security object data. EX1005 (Keren), 11:61-12:10. This may be accomplished by “[t]agging of security objects” such as by “association of one or more data labels, tags, or other, like, features, with graph entries for one or more security objects” (i.e., integrating the data into the graph). *Id.*, 13:6-11; *compare* EX1001, 37:3-11 (“[S]canning system 101 may integrate information on services available in each node into the data structure, associated with each node in the data structure.”).

“Examples of relevant descriptions” that may be integrated into the graph include “object details, such as may be collected at S210, ... descriptions of an object’s open ports and counts thereof,” and “descriptions of protocols relevant to an object,” which a POSA would have understood would include the security object data identifying services I discussed above. EX1005 (Keren), 13:12-25; *see also id.*, 11:14-23 (data label descriptions may include “connection source and destination, connection type, connection direction, connection status, connection protocol, and the like.”), 13:43-46 (“Each object in the graph may be associated with known properties of the object. Examples of such properties may include an object’s name, its IP address, various predefined security rules or access rules, and the like.”), 14:42-46 (identifying, for path identification, “element properties or attributes including, without limitation, names, network addresses, rulesets, port configurations, and the like”); *supra*, §XI.A.6.

78. Keren’s teaching of integrating the identified services into the graph is further illustrated by the system’s calculation of a security object’s reachability properties, which I discuss in more detail below. *Infra*, §XI.A.8. Specifically, based on the generated graph and the integrated object properties, Keren teaches that a network path within the security graph may be determined. EX1005 (Keren), 13:26-65. An example of a path determination is shown in Fig. 5 (reproduced below), in

which the reachable properties of a database 510 are calculated. *Id.*, 4:1-2, 15:15-35, 18:40-43, Fig. 5. As Keren describes, the database's reachable properties are determined based on the properties of objects within the network path, which includes a public proxy 540 having a translation rule in which internet traffic into a frontend port 1234 is translated to a backend port 4321 (i.e., an identified service publicly accessible via the internet and capable of serving as an internet proxy). *Id.*, 18:55-58, 19:62-20:12, Fig. 5; *see also infra*, §XI.A.8.

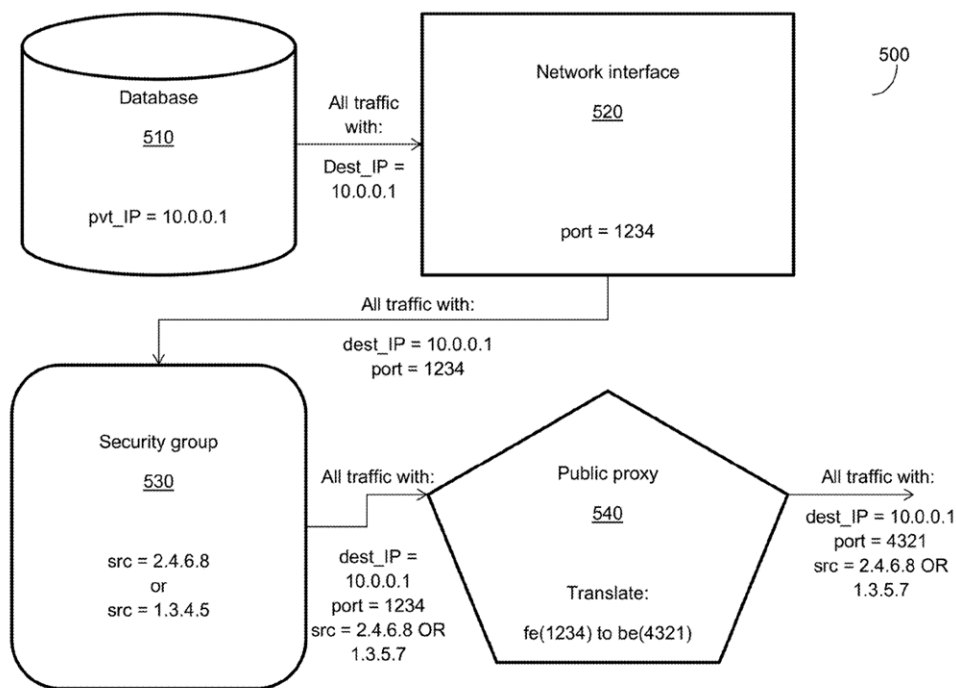


FIG. 5

EX1005 (Keren), Fig. 5.

79. Additional evidence demonstrating the integration of the identified services into the constructed security graph is shown by Keren's description of user

visualization features, as illustrated in Figs. 4A-4B. For example, Keren teaches that a “security graph utility” may be provided as “an application, interface, or other, like, means of providing a visual representation of a security graph.” EX1005 (Keren), 16:41-53, Fig. 4A. The security graph utility (shown below) may include various interactive functionalities, including a display functionality where a user may click on a security object (represented by a node vertex visualization 410) to display a vertex overview pane 415. *Id.*, 17:5-8. Keren explains that the vertex overview pane “may be an information panel, including data relating to the given vertex visualization 410 and describing various object data features, such as those object data features identified or determined during the execution of a process similar or identical to that described with respect to FIG. 2.” *Id.*, 15:8-13; *see also id.*, 15:13-18 (vertex visualization information may include “object names, types, statuses, relevant metadata, and the like, as well as any combination thereof”).

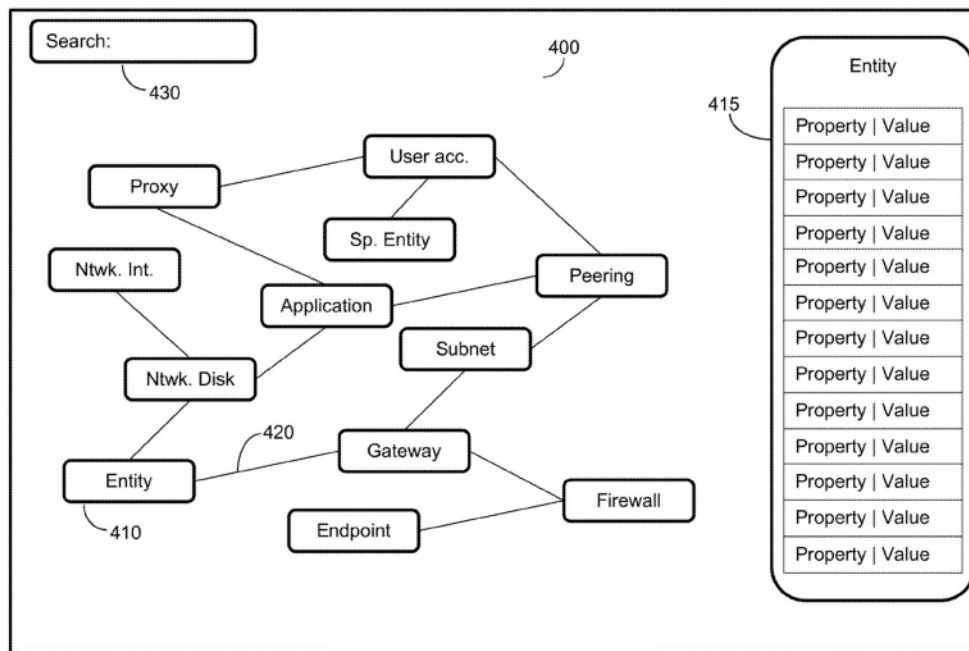


FIG. 4A

EX1005 (Keren), Fig. 4A.

80. Keren also describes that a mode-specific feature may be provided, where a user may select “list-organization profiles,” which provides “for population of a security graph object list 450 in one or more configurations.” *Id.*, 18:8-22. One example, as shown in Fig. 4B (reproduced below), is “a ‘route’ mode” in which a security graph object list 450 representing “a data path between a first object 470-1 and a destination object 470-n” is provided. *Id.*, 18:23-27; *see also id.*, 17:39-43. In the illustrated example, the mode-specific data display 465 displays “‘9 hops,’ indicating that a transmission from a first object 470-1,” a container, “makes nine

‘hops,’ or transmissions between objects, before reaching the destination object 470-n,” the internet. *Id.*, 18:27-39, Fig. 4B.



FIG. 4B

EX1005 (Keren), Fig. 4B. As shown in the figure, a proxy services transmission from the container (and intervening security objects) to the internet. Thus, in both cases, Keren further demonstrates that the identified services are integrated into the constructed security graph so that the system may provide visualization features to a user.

81. It is therefore my opinion that Keren teaches all aspects of elements 1.5, 11.5, and 20.5.

8. Elements 1.6, 11.6, and 20.6

82. These elements recite “[traverse/traversing] the graph to identify at least one [source/vector] originating via the [I/i]nternet and reaching the workload.” As I explain below, Keren teaches these elements.

83. As an initial matter, I note that claims 1 and 11 recite identifying “at least one source” originating via the internet, while claim 20 recites identifying “at least one vector” originating via the internet. The ’257 patent explains that a “source” that originates via the internet may be “a potential malicious source,” where, in some embodiments, “the Internet at large may be considered to be a potentially malicious source.” EX1001, 37:25-31; *see also id.*, 38:15-20 (the scanning system “may analyze the graph to detect sources coming from the internet and reaching the workload”). The ’257 patent describes a “vector” as “connecting the plurality of systems and devices” in the graph, where the scanning system “may traverse the map to identify vectors originating in the Internet and reaching the devices and systems” in the cloud environment. *Id.*, 12:65-13:10; *see also id.*, 36:30-39 (“Directional vectors, as used herein, may refer to a quantity having direction as well as magnitude (indicating, for example, what kinds of traffic are allowed to flow, what ports are open, what protocols are allowed), especially as determining the position of one point in space relative to another.”). Keren teaches both these elements.

84. Based on the constructed graph and integrated object information, Keren teaches that the system calculates a “reachability property” of the security objects represented in the graph. *Id.*, 11:62-12:5, 13:35-46, Figs. 3A-3B. A reachability property of a security object “defines if and how an entity can be reached from an external or internal network,” where the external network may be the internet. *Id.*, 12:5-7, 14:1 (“an external network (the internet)”); *see also id.*, 13:26-34 (“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.”), 15:15-24 (“Examples of reachable properties include, without limitation, binary properties describing whether an element is reachable, protocols by which the element is reachable, network addresses at which an element is reachable, ports by which an element is reachable, access rules, and the like, as well as any combination thereof.”). As detailed more below, to calculate a security object’s reachability property, Keren’s system traverses the graph to identify a source or vector originating from the internet and reaching the security object. Security objects, as I explained above, include workloads, such as virtual machines, containers, load balancers, storage, and databases. *Supra*, §XI.A.1.

85. Specifically, Keren teaches accessing a security graph stored in the graph database, where each security object in the graph is associated with known

properties of the object, such as those that I discussed above. EX1005 (Keren), 13:26-46; *supra*, §§XI.A.6-7. Based on this information, the system identifies “possible network paths within the obtained security information,” in which each “path shows a connection between security objects and/or computing objects that communicate over a network” (i.e., a vector). EX1005 (Keren), 13:47-48, 13:62-65; *see also id.*, 13:51-61 (“[A] network path may include sequential representations of possible data/control flows between two or more objects in a graph.”). Keren explains that “paths can be identified by **traversing** the security graph.” *Id.*, 13:66-67 (emphasis added). “The traversal can start or end at objects that are connected to an external network (the internet),” and can be performed using known means. *Id.*, 13:67-14:5.

86. Keren further describes that each path is analyzed to calculate the reachability properties of the security objects, including workloads, contained within the path. *Id.*, 14:15-26, 14:31-58, Fig. 3B; *see also id.*, 14:59-15:2 (describing example network path from a virtual machine (VM), connected to a NIC, connected to a load balancer, connected to a firewall). “[A] reachable property is determined as a minimal set of reachable properties of all other objects in the path.” *Id.*, 15:25-27. As a simple example of a reachability property calculated for an object, Keren teaches identifying sources that may reach the object:

[I]f a path includes two objects where one object can receive traffic from any source IP address through port 1515, and the other object can receive traffic only from a source IP address of 173.54.189.188, the reachable property of the second object may be that the second object is reachable through **source IP address 173.54.189.188** and port 1515.

Id., 15:27-33 (emphasis added).

87. Keren expressly teaches that a reachability property for an object may be calculated based on whether and how an object is accessible from a source (or vector) originating from the internet. For example, Keren teaches that reachability properties for objects may be populated into the security graph as “object network access control lists (NACLs).” *Id.*, 15:36-50. Keren explains that the NACLs may be “[s]cope-specific,” which are NACLs “describing 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.” *Id.*, 15:50-53 (emphasis added). Thus, an NACL having a scope of the internet would “describe the properties of an object with respect to the object’s internet accessibility.” *Id.*, 15:53-58; *see also id.*, 20:34-42; *compare* EX1001, 37:29-31 (“In some embodiments, the Internet at large may be considered to be a potentially malicious source.”).

88. Keren provides a specific example in which a path of the graph is traversed to identify at least one source (or vector) originating via the internet and reaching a workload. As shown in Fig. 5 (reproduced below), an exemplary network

path containing a database 510 (i.e., a workload), a network interface 520, a security group 530, and a public proxy 540 (i.e., an internet-accessible service) is analyzed to calculate the reachability properties of the database 510. EX1005 (Keren), 18:40-43.

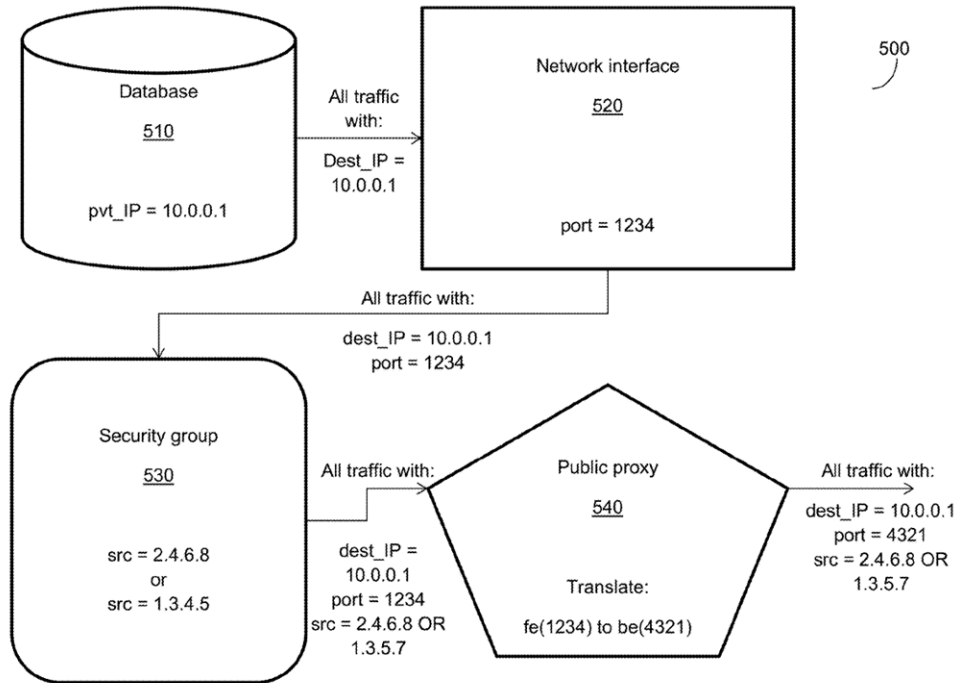


FIG. 5

Id., Fig. 5.

89. As Keren explains, each of the objects within the network path include accessibility properties that determine the source of internet-originating traffic that can be received at the database 510. See *id.*, 18:59-19:18. For example, the database itself “includes a private IP address value, included in the element’s properties, of 10.0.0.1, providing for accessibility of the database 510 only to traffic with a

destination address of 10.0.0.1.” *Id.*, 18:44-48; *see also id.*, 19:34-43. Next, the system then analyzes the collected properties of the network interface 520 to determine that “the only network interface object 520 port relevant to database object traffic is port number 1234,” which would restrict traffic to the database to “all traffic with a destination IP address of 10.0.0.1, a port number of 1234, and any source.” *Id.*, 19:43-50; *see also id.*, 18:48-51. By further analyzing the security group 530’s properties, the system determines that traffic to the database is additionally restricted to only traffic from sources 2.4.6.8 and 1.3.5.7. *Id.*, 19:50-61; *see also id.*, 18:52-55. Finally, the public proxy 540’s properties “include[] a rule which translates frontend port 4321 to backend port 1234.” *Id.*, 19:62-66; *see also id.*, 18:55-58. Taken together, the system determines that allowable traffic that matches the specified restrictions of “all traffic with a destination IP address of 10.0.0.1, a port number of 4321, and a source of 1.3.5.7” and “all traffic with a destination IP address of 10.0.0.1, a port number of 4321, and a source of 2.4.6.8” can be associated with the database 510. *Id.*, 19:66-20:12. Moreover, based on this analysis, Keren explains that other reachability properties of the database can be determined, including properties that identify the number of ports open to the internet (“numPortsOpenToInternet=1”) or whether the database is accessible from the internet (“accessibleFromInternet=True”). *Id.*, 20:5-12. Thus, Keren teaches both

identifying a source originating via the internet (e.g., sources 1.3.5.7 and 2.4.6.8) and reaching the workload (e.g., database 510) and identifying a vector originating via the internet and reaching the workload (e.g., the data path entering public proxy 540 and reaching database 510), as recited.

90. It is therefore my opinion that Keren teaches all aspects of elements 1.6, 11.6, and 20.6.

9. Elements 1.7, 11.7, and 20.7

91. These elements recite “[output/outputting] a risk notification associated with the workload.” Keren teaches these elements.

92. Keren teaches that an aim of its system is to provide risk notifications associated with workloads in a network. Keren explains 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), 1:22-25. Keren then explains that prior network management systems “fail 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” to network administrators. *Id.*, 2:42-49. Keren thus provides a risk-notification solution “which allows for network analysis and includes granular, comprehensive representation of security

object properties and insights,” including “reachability properties of security objects.” *Id.*, 2:49-55, 3:6-7.

93. More specifically, Keren describes that a risk notification associated with the workload may be outputted by the system in the form of security insights. For example, Keren teaches that, “based on the known and calculated properties” of the security objects, “security insights may be generated,” which “are natural-language representations of aspects of the security graph constructed at S220.” *Id.*, 12:11-14; *see also id.*, 12:14-44 (explaining that security insights may provide “object relationship description[s]” of varying detail). As I explain below, in providing “a solution which allows ... granular, comprehensive representation of security properties and insights” (*id.*, 2:49-55), Keren teaches, or at the very least suggests, that these insights are outputted as notifications to a user.

94. Keren teaches various ways in which users are notified of information relevant to the network analysis performed by the system, including the security insights that are generated by the system. For example, Keren teaches that the security insights may be tagged to security objects in the security graph. *Id.*, 13:6-25 (“Tagging of security objects ... may include, without limitation, association of one or more data labels, tags, or other, like, features, with graph entries for one or more security objects....”). In addition, as I mentioned above (*supra*, §XI.A.7),

Keren also teaches that the security graph and its associated data may be outputted as a “security graph visualization” to a user, as shown below. *Id.*, 16:41-53.

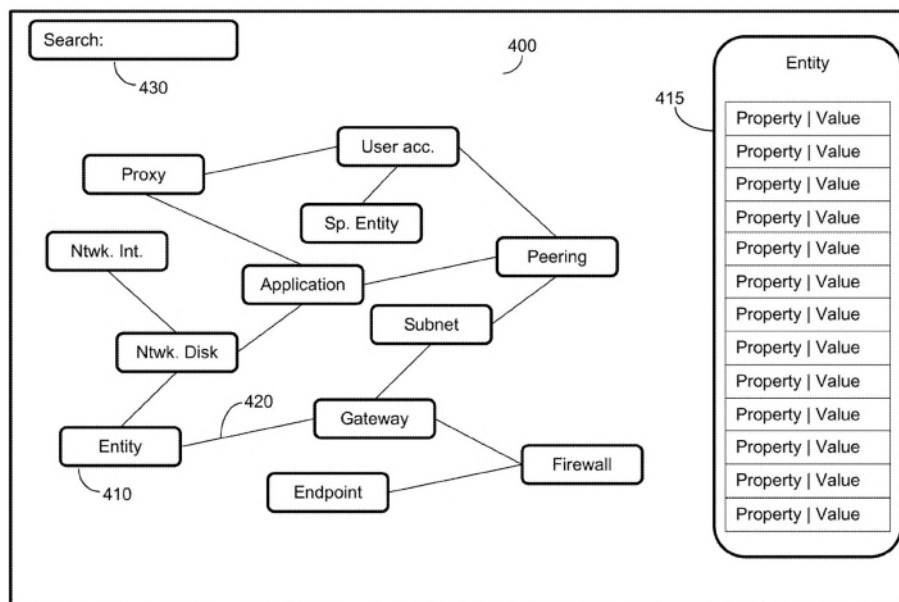


FIG. 4A

EX1005 (Keren), Fig. 4A.

95. Keren teaches that the security graph visualization can notify the user of relevant information, including tagged security insights, through various features, such as an information panel and a search function. For instance, the security graph visualization can include a functionality that allows a user to click on a security object (represented by vertex visualization 410) to display a vertex overview pane 415, which is an information panel that outputs to the user “data relating to the given vertex visualization 410” and “describ[es] various object data features.” *Id.*, 16:54-62, 17:1-13. Keren teaches that the object data features may include “features

identified or determined during the execution of a process similar or identical to that described with respect to FIG. 2” (*id.*, 17:10-18), which includes features provided in data labels or tags associated with the security graph. *Id.*, 11:14-17 (“Graphs may be updated at S320 by associating one or more data labels, tags, or other, like, features with a graph entry for a security object.”). As I mentioned above, Keren teaches that tags associated with the security graph include the security insights generated by the system. *Id.*, 13:6-25.

96. Keren also describes providing “a search tool 430” with the security graph visualization, which allows the users to search for user-specified object data features, such as “object names, types, IDs, statuses, labels or tags associated with various elements of the security graph visualization 400, and the like.” *Id.*, 17:19-34. Thus, Keren teaches that a user may be notified of the generated security insights through, for example, an information panel associated with a given security object or through a search tool that outputs the relevant information.

97. In addition, Keren contemplates providing functionalities that generate alerts associated with the cloud environment. *See id.*, 8:54-63 (describing an orchestrator 115 that “generat[es] cloud platform 110 alerts”). In view of Keren’s described aim of providing network administrators with network information that “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), a POSA would have found it obvious to also notify a user of security insights (especially those describing configurations that present risk to a workload, as I detail more below) via an alert so that the user can be alerted to risks presented by a current network configuration. Thus, in view of the above, Keren teaches, or at the very least renders obvious, outputting to a user the security insights generated by the system.

98. Keren further explains that the generated security insights may provide network descriptions that a POSA would have understood to be notifications of risk to workloads contained within the cloud environment. *See* EX1005, 2:37-49 (explaining need to notify network administrators of "the vulnerabilities presented by the entities' placements within the network configuration"). For example, Keren teaches:

[G]eneration of security insights ... may include the generation of insights providing for, without limitation, description of network management or anomaly detection data, 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., 12:45-55. As one specific example, Keren describes a generated insight for a workload (i.e., a virtual machine) "specifying 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 provided a risk notification associated with the workload because the insight would notify a user that the virtual machine was no longer connected to a network security device designed to prevent unauthorized access to the workload (i.e., a firewall). *See, e.g.*, EX1020 (NIST Incident Handling), 17, 37 (explaining firewalls strengthen network security perimeter);¹⁵ EX1021 (Li), ¶33 (teaching use of firewall to prevent unauthorized asset exploitation); EX1025 (Basavapatna), ¶¶22-25 (discussing use of passive countermeasures, such as firewalls, to protect against vulnerability exploitation).

99. Keren also teaches that the generated insights may include “high-level insights, where high-level insights include information describing one or more features of a network which may not be detectable based on the analysis of individual objects.” *Id.*, 12:62-66. “Examples of high-level insights may include, without limitation, ‘third-party networks A, B, and C currently have access to the internal

¹⁵ EX1020 is a copy of an August 2012 NIST publication obtained from <http://dx.doi.org/10.6028/NIST.SP.800-61r2>.

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.’” *Id.*, 12:66-13:5. A POSA would have understood that these high-level insights also provided a risk notification associated with a workload, such as providing notification as to whether a workload containing sensitive information or providing privileged access (e.g., a database or administrator interface) was exposed to an external network (e.g., the internet). *Compare* EX1001, 37:29-31 (“In some embodiments, the Internet at large may be considered to be a potentially malicious source.”).

100. It is therefore my opinion that Keren teaches all aspects of elements 1.7, 11.7, and 20.7.

101. Accordingly, it is my opinion that independent claims 1, 11, and 20 were obvious.

B. Dependent Claims

1. Claims 2-3 and 12-13

102. Claims 2 and 12 depend from claims 1 and 11, respectively, and further recite “wherein the plurality of entities includes a virtual network appliance.” Claims 3 and 13 depend from claims 2 and 12, respectively, and further recite “wherein the virtual network appliance is at least one of a load balancer, a firewall, a proxy, or a router.”

103. As I explained above, Keren teaches a plurality of entities capable of routing traffic, such as firewalls, proxies, and load balancers. *Supra*, §XI.A.3; *see also* EX1005 (Keren), 5:32-39, 6:10-19, 9:6-15, 10:11-20. Furthermore, Keren teaches that these objects may be “virtual entities or instances of systems, devices, or components, including virtual systems, devices, or components, or any combination thereof.” EX1005 (Keren), 6:10-19; *see also id.*, 5:45-51 (security objects provide network support for application-layer communications, including layer seven of OSI model), 5:66-6:9 (“[S]ecurity objects may include virtual entities, devices, and the like, to process layer-7 (application layer) traffic, such as objects relevant to Amazon AWS® layer seven services and applications, Amazon Load Balancer® (ALB) layer seven services.”), 9:6-15 (security objects include virtual entities such as load balancers), 13:64-65 (“An object may be a virtual, physical, or logical entity.”).

104. These teachings of virtual security objects, as described by Keren, are consistent with a “virtual network appliance” as described by the ’257 patent. EX1001, 34:38-51 (“A virtual network appliance, as used herein, may refer to a machine focused on virtualizing network functionality. In some embodiments, a typical network virtual appliance may include functionality directed to various layer four to seven. That is, such appliances may implement functionality associated with

OSI model layers 4-7 (Transport, Session, Presentation, and Application). Such functionality may include, for example, a firewall, WAN optimizer, application delivery controllers, routers, load balancers, IDS/IPS, proxies, or SD-WAN edge device.”), 34:52-53. Thus, Keren teaches a plurality of entities that includes a “virtual network appliance,” such as a load balancer, firewall, or proxy, as recited.

105. It is therefore my opinion that claims 2-3 and 12-13 were obvious.

2. Claims 4 and 14

106. Claims 4 and 14 depend from claims 1 and 11, respectively, and further recite “wherein the networking configuration is at least one of a routing configuration, a proxy configuration, a load balancing configuration, a firewall configuration, or a VPN configuration.”

107. As I discussed above, Keren teaches determining the networking configuration of the plurality of entities (e.g., security objects providing traffic-management functionalities), which may include “object network addresses,” “object port configurations,” “object [input/output] signal configurations,” and “object security or access configurations.” EX1005 (Keren), 5:32-37, 9:56-10:10; *supra*, §XI.A.4. As I also discussed above, Keren teaches that the entities may include firewalls, proxies, load balancers, routing tables, and VPNs. EX1005 (Keren), 5:38-39, 5:66-6:4, 6:14-19, 9:9-15; *supra*, §XI.A.3. Because Keren teaches determining the networking configuration of these entities, Keren teaches that “the

networking configuration is at least one of a routing configuration, a proxy configuration, a load balancing configuration, a firewall configuration, or a VPN configuration,” as recited. *See, e.g.*, EX1005 (Keren), 10:14-20 (describing collection of a firewall’s rules, event logs, and port configurations), 19:62-20:5 (describing determination of public proxy’s port configuration), Fig. 5.

108. It is therefore my opinion that claims 4 and 14 were obvious.

3. Claims 5 and 15

109. Claims 5 and 15 depend from claims 1 and 11, respectively, and further recite “wherein the graph includes a data structure sequentially connecting entities.”

110. I explained above how Keren teaches building a security graph, which is a data structure that describes the various security objects included within the virtual environment and the relationships between those objects. EX1005 (Keren), 10:27-33, Fig. 2; *supra*, §XI.A.5. Keren teaches that the security graph may be constructed in various formats, including formats where the data structure sequentially connects the security objects. EX1005 (Keren), 10:33-48; *compare* EX1001, 36:26-29 (“A data structure sequentially connecting entities, as used herein, may refer to an organization, management, and/or storage format that enables efficient access and modification of data such as entities.”). For example, the security graph may be a “visual ‘node and link’ graph,” in which security objects, represented by nodes, are sequentially connected along any given path via links.

EX1005 (Keren), 10:33-48, 11:28-37, Fig. 4A; *see also id.*, 7:15-24. Keren teaches that the security graph may provide a routing display feature that sequentially connects security objects along a specific data path. *Id.*, 18:23-39, Fig. 4B. Moreover, Keren discloses that the network paths identified by the system “may include sequential representations of possible data/control flows between two or more objects in a graph.” *Id.*, 13:51-53, Fig. 5; *see also supra*, §XI.A.8.

111. It is therefore my opinion that claims 5 and 15 were obvious.

4. Claims 6 and 16

112. Claims 6 and 16 depend from claims 1 and 11, respectively, and further recite “wherein the graph includes directional vectors indicating directions of dataflow.”

113. Keren teaches that the security graphs may “include one or more functional attributes,” such as “directionality.” EX1005 (Keren), 7:30:36. “A graph including directionality may be configured to include connection between graph nodes or vertices, via graph edges, ... where the edges connecting such vertices may be uni-directional or bi-directional, providing for enhanced analysis of security object structures and relationships.” *Id.*, 7:36-42; *see also id.*, 11:17-23 (graph may include labels indicating “connection direction”). For instance, Keren describes that a virtual machine configured to run a container may include a uni-directional edge (i.e., a directional vector) between the virtual machine and the container that would

indicate how data flows between the virtual machine and the container (e.g., that the virtual machine runs the container). *Id.*, 7:46-56.

114. It is therefore my opinion that claims 6 and 16 were obvious.

5. Claims 7 and 17

115. Claims 7 and 17 depend from claims 1 and 11, respectively, and further recite similar limitations. Claim 7 recites “to build the graph, the at least one processor is configured to: identify individual entities as nodes; and connect the nodes.” Claim 17 recites “wherein building the graph comprises: identifying individual entities as nodes; and connecting the nodes.”

116. Keren teaches that the security graph may be constructed as “a visual ‘node and link’ graph,” as illustrated below. EX1005 (Keren), 11:28-37, Fig. 4A. Keren further teaches that, to construct the graph, individual security objects (including the plurality of entities) are identified as nodes, which are connected via links or edges based on the determined object relationships. *Id.*, 11:64-66, 13:36-42, 16:54-67; *supra*, §XI.A.5.

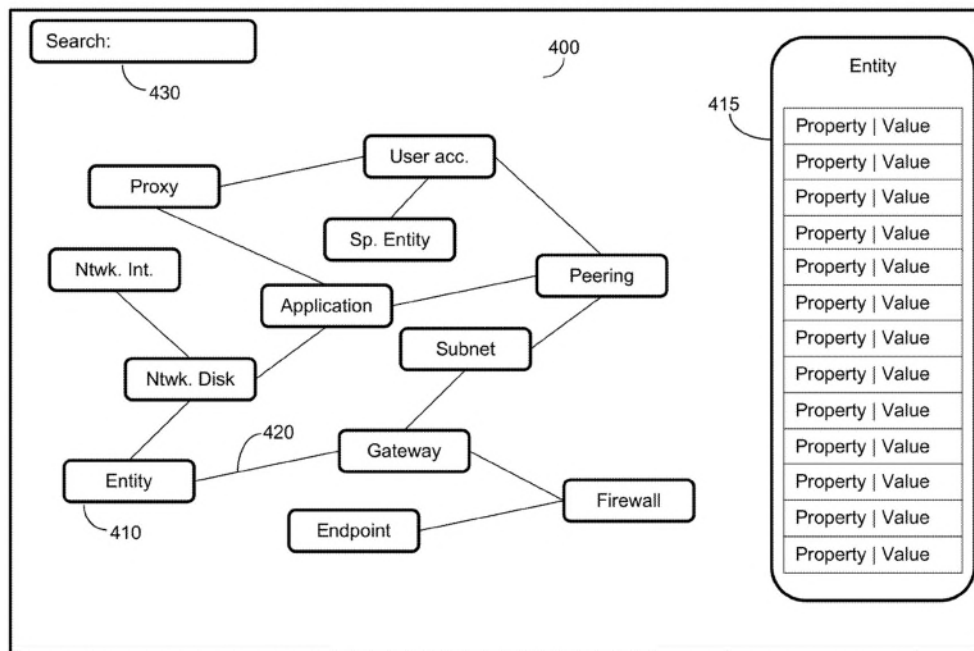


FIG. 4A

117. It is therefore my opinion that claims 7 and 17 were obvious.

6. Claims 8 and 18

118. Claims 8 and 18 depend from claims 1 and 11, respectively, and further recite “wherein the graph includes port numbers associated with the workload.”

119. Keren teaches that the security object data collected by the system may include port configurations of the security objects (including workloads). EX1005 (Keren), 9:56-66; *see also id.*, 10:20-26 (describing collection of “VM port configurations”). Keren further teaches that the security object relationships may be based on the port configurations of the security objects, and descriptions of object relationships may be integrated into the graph. *Id.*, 10:52-57 (“Network

relationships may describe aspects of the connections between objects including ... relevant ports of connected objects”), 11:11-17, 11:66-12:1, 12:11-12, 13:6-25, 14:42-46, 15:19-24 (reachable properties may include “ports by which an element is reachable”), 15:36-37. As one example, Keren describes an insight that specifies, in part, “a path from ‘virtual machine one to load balancer five, where port eighty is routed to port 1337 on virtual machine one’” *Id.*, 12:26-44; *see also id.*, 12:22-25 (“An example of a detailed object relationship description may be ‘the gateway is currently active, and is connected to the VM via the second port, using the first protocol.’”), 19:66-20:12; *supra*, §XI.A.9 (explaining how the security graph is tagged with insights).

120. It is therefore my opinion that claims 8 and 18 were obvious.

7. Claims 9 and 19

121. Claims 9 and 19 depend from claims 1 and 11, respectively, and further recite “wherein the graph includes a path from the at least one source to the workload.”

122. As I explained above, Keren teaches that the system is configured to identify network paths between security objects, including workloads, by traversing the security graph. EX1005 (Keren), 13:47-14:16; *supra*, §XI.A.8. For example, as I also detailed above, Keren describes the analysis of a path in a security graph from

a source originating via the internet to a database. EX1005 (Keren), 18:40-43, 19:34-20:12, Fig. 5; *supra*, §XI.A.8.

123. It is therefore my opinion that claims 9 and 19 were obvious.

8. Claim 10

124. Claim 10 depends from claim 1 and further recites “wherein the risk notification includes one or more resolution recommendations.” As I describe below, Keren renders this limitation obvious.

125. As I explained above with respect to the independent claims, Keren teaches outputting security insights that may notify a user of a risk associated with a workload. *Supra*, §XI.A.9. The security insights can notify a user as to network configurations or events that put a workload at risk. EX1005 (Keren), 12:45-13:5. For example, the security insights may provide notification of rare or new events, “such as connection of a new device to a network,” unauthorized events, “such as re-connection of a user’s device to a subnet which the user is not permitted to access,” or connections that display anomalous behavior, “such as connections displaying spikes of network activity.” *Id.*, 12:45-55. The security insights may also provide notification as to whether important computing resources, such as a database or administrator interface, are exposed to an external network, or if third-party networks have access to an internal network. *Id.*, 12:62-13:5; *see also supra*, §XI.A.9. A POSA would have understood that these insights taught including a

resolution recommendation in the risk notification. For example, an insight noting a current abnormal network condition relative to a normal condition would have been understood to include a recommendation to a user to update the configuration to the identified normal condition. That is, for instance, in an insight specifying “VM thirty normally connects to load balancer twenty and firewall eight but is currently only connected to load balancer twenty,” the insight would convey to the user a recommendation to connect VM thirty to firewall eight. EX1005 (Keren), 12:45-55.

126. Furthermore, a POSA would have found it obvious from Keren’s teachings to include one or more explicit resolution recommendations with the security insights. It was well-known that a strong network security perimeter provided a good defense mechanism for mitigating the occurrence of risky network configurations and events described by Keren. See EX1020 (NIST Incident Handling), 23-24 (“The network perimeter should be configured to deny all activity that is not expressly permitted. This includes securing all connection points, such as virtual private networks (VPNs) and dedicated connections to other organizations.”). POSAs also knew common methods to strengthen the security perimeter of a network, including, for example, installing firewalls to protect important assets and changing rulesets or access control lists of security devices to manage incoming

network traffic. *See id.*, 17 (altering network security controls (e.g., firewall rulesets) for incident handling), 37 (incident prevention includes “tightening network perimeter security (e.g., firewall rulesets, boundary router access control lists)”); EX1021 (Li), ¶33 (changing installed firewall settings to protect against vulnerability exploitation); EX1025 (Basavapatna), ¶¶22-25 (discussing use of passive countermeasures, such as firewalls, to protect against exposure). A POSA would have understood that these routine methods would aid in mitigating the network risks described in Keren. For example, connection of a firewall to resources, such as a virtual machine or database, would reduce the resource’s exposure to external networks, and updating access control lists of a security device would aid in reducing the likelihood of unauthorized connections to an internal network.

127. A POSA would have found that including express resolution recommendations with the security insights taught by Keren to be a straightforward and logical extension of Keren’s teachings. In particular, Keren teaches the benefits of providing management and notification functionalities to a user and combining these functionalities together in a system. *See* EX1005 (Keren), 8:55-63 (“The orchestrator 115 may be configured to provide one or more functionalities, including, without limitation, monitoring of elements or components of the cloud

platform 110, logging and reporting data relating to the cloud platform 110, managing cloud platform 110 updates and maintenance, generating cloud platform 110 alerts, as well as other, like, functionalities, and any combination thereof.”). Thus, to further achieve Keren’s aim in providing network administrators with a system that can enable effective network management of an organization’s cloud environment (*see id.*, 1:36-49, 2:34-55), a POSA would have found it apparent to include resolution recommendations relating to the management of a security object (such as a workload) with the relevant security insight generated for that object.

128. For example, in a security insight specifying 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 readily appreciated that providing an express recommendation that VM thirty be re-connected to firewall eight (e.g., by recommending the VM be reconfigured to connect to the firewall or the network elements reconfigured to route VM traffic to the firewall) could be beneficially included in the security insight to aid the administrator in addressing the detected network risk. Similarly, in security insights notifying a user of unauthorized connections (e.g., third-party networks) to an internal network, a POSA would have understood that including as part of the security insight an explicit resolution recommendation that access control lists governing access into the

network should be updated would also improve the response to addressing the detected risk. Thus, in light of Keren’s teachings, a POSA would have found it obvious to generate a risk notification that “includes one or more resolution recommendations,” as recited.

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

XII. GROUND 2: CLAIMS 1-20 WERE OBVIOUS OVER KEREN AND ENGELBERG

130. In this ground, I discuss a combination of Keren and Engelberg. This builds upon my analysis of Keren discussed in Ground 1, further applying Engelberg’s disclosure of including one or more resolution recommendations in a risk notification. In §§XII.B-C, I explain how these teachings apply to independent claims 1, 11, and 20 and dependent claim 10. The combination with Engelberg does not change my previous explanation in Ground 1 of why the other limitations of 1-20 were obvious (*supra*, §XI), since the applied teachings of Engelberg are consistent with the cybersecurity systems and methods taught by Keren.

131. In the analysis below, I provide parallel citations to Engelberg and its provisional application (EX1007), showing where support for the cited portions of Engelberg can be found in the equivalent disclosure of the provisional.

A. Reasons to Combine Keren and Engelberg

132. In my opinion, a POSA would have had reason to combine the teachings of Keren and Engelberg to arrive at claims 1-20.

133. As an initial matter, Keren and Engelberg each describe systems and methods that are analogous to one another and to the '257 patent. EX1006 (Engelberg), ¶6 (“Implementations of the present disclosure are directed to mitigating cyber security risk in enterprise networks.”); EX1007 (Engelberg Provisional), ¶5; *supra*, §§X.A-B. Engelberg is therefore in the same field as Keren and the '257 patent, and its teachings are pertinent to issues addressed by the '257 patent since those teachings involve detecting security risks in computer networks and resources and identifying resolutions to those risks. *See, e.g.*, EX1001, 33:49-51 (“Aspects of this disclosure may include a cloud-based cybersecurity system for assessing internet exposure of a cloud-based workload...”), 37:32-45 (“[T]he risk notification includes one or more resolution recommendations,” which “may refer to a suggestion or proposal as to the best course of action in view of the identified risk.”).

134. Furthermore, providing risk notifications that included resolution recommendations for risks identified in a computer network, as taught by Engelberg, was well-known and predictable. *See, e.g.*, EX1022 (Mohanty), 9:14-26, 10:27-34 (determining remediation for computer assets and providing alerts); EX1023, ¶¶210-211 (generating report of risk assessment and including “suggestions, courses of corrections, warnings, or the like”); EX1024 (Tan), ¶¶60-63 (providing

recommendation based on anomalous activity or device connection). Thus, the combination of Keren and Engelberg would have merely involved using the familiar technique of additionally including a resolution recommendation in a risk notification generated by the cybersecurity system.

135. In addition, a POSA would have been motivated to apply Engelberg's teachings to Keren to alert an administrator that a security insight identified by Keren's cybersecurity system presented a risk, and further to provide additional information about that risk by presenting a potential remediation activity. For example, Engelberg teaches the benefits to including resolution recommendations when notifying security personnel of risks present in the network environment. Engelberg explains that "[c]omputer networks are susceptible to attack by malicious users (e.g., hackers)," especially as "[e]nterprises, in general, and industrial infrastructures, in particular, are increasingly connected to external networks such as the Internet," making them increasingly vulnerable to external cyber-attacks. EX1006 (Engelberg), ¶2; EX1007 (Engelberg Provisional), ¶1. Thus, Engelberg teaches that, in addition to notifying administrators of risks present in a computer network, it is helpful to identify "mitigation efforts to mitigate risk in enterprise networks" and provide those recommendations to security personnel so that personnel can "understand immediate remediation actions corresponding to each

issue.” EX1006 (Engelberg), ¶¶6, 59; *see also id.*, ¶¶30, 42 (“In some examples, the agile security platform provides one or more dashboards, alerts, notifications, and the like to cyber security personnel that enable the cyber security personnel to react to and remediate security relevant events.”); EX1007 (Engelberg Provisional), ¶¶5, 28, 40, 57.

136. In view of Engelberg’s teachings regarding the benefits of notifying administrators of risks present in a network and providing resolution recommendations for the identified risk, a POSA would have had reason to similarly explicitly identify security insights generated by Keren’s cybersecurity system as risks to administrators and include resolution recommendations in the notification. As I explained above, Keren teaches generating security insights that may provide notification to risks present within the network, such as rare or novel network configurations or events, unauthorized connections, or exposed assets. *Supra*, §§XI.A.9, XI.B.8. Keren explains that, in managing large network environments, these insights aid security personnel in understanding “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.” EX1005 (Keren), 2:42-55. Engelberg similarly teaches identifying risks associated with network or system configuration conditions to a user. EX1006 (Engelberg), ¶¶57,

59; EX1007 (Engelberg Provisional), ¶¶55, 57. Thus, a POSA would have appreciated that Keren’s generation of security insights could readily be improved by expressly identifying risky network configurations as risks in a notification and also including resolution recommendations to aid in decision-making for security personnel, as taught by Engelberg.

137. A POSA would have had a reasonable expectation of success in combining the above teachings of Keren and Engelberg to arrive at the claimed invention. As I mentioned above, Keren recognizes that object management functionality can be readily combined with notification features. *See* EX1005 (Keren), 8:56-63; *supra*, §XI.B.8. As I also explained above, providing resolution recommendations with notifications regarding risks present in a computer network was well-understood and routinely used. *See, e.g.*, EX1022 (Mohanty), 9:14-26, 10:27-34; EX1023, ¶¶210-211; EX1024 (Tan), ¶¶60-63. Moreover, a POSA was well-aware of the types of resolutions that would aid in mitigating risks such as those taught by Keren and would have found it routine to identify such resolutions for inclusion with the corresponding security insight. *See, e.g.*, EX1020 (NIST Incident Handling), 17 (altering network security controls (e.g., firewall rulesets) for incident handling), 23-24 (configuring network perimeter to deny unauthorized connections), 37 (incident prevention includes “tightening network perimeter

security (e.g., firewall rulesets, boundary router access control lists”); EX1021 (Li), ¶33 (changing installed firewall settings to protect against vulnerability exploitation); EX1025 (Basavapatna), ¶¶22-25 (describing use of “passive countermeasures” to protect against cyber risks, including “hardware firewalls, software firewalls, data loss prevention systems, web proxies, mail filters, host-based intrusion prevention systems, network-based intrusion prevention systems, rate-based intrusion prevention systems, content-based intrusion prevention systems, intrusion detection systems, and virus detection software”); *see also supra*, §XI.B.8. Therefore, in my opinion, combining the references’ teachings as discussed above would not have presented meaningful technical challenges or produced unexpected results.

B. Claims 1, 11, and 20

138. Elements 1.7, 11.7, and 20.7 recite “[output/outputting] a risk notification associated with the workload.” As I explained above, these elements were obvious in view of Keren, which teaches a system configured to output risk notifications associated with security objects (e.g., workloads) contained within a cloud environment. *Supra*, §XI.A.9. As I explain below, independent claims 1, 11, and 20 are further obvious in view of Engelberg’s similar teachings regarding risk notifications.

139. Engelberg is directed to “Automated Prioritization of Process-Aware Cyber Risk Mitigation” that involves “generating ... a risk assessment” to “mitigate cyber security risk in enterprise networks.” EX1006 (Engelberg), Title, Abstract (“Implementations are directed to ... generating, by a process-aware risk assessment module, a risk assessment based on the process-aware AAG and the risk profile.”), ¶6 (“Implementations of the present disclosure are directed to mitigating cyber security risk in enterprise networks.”); EX1007 (Engelberg Provisional); Title, Abstract, ¶5. Engelberg teaches the straightforward proposition that it is beneficial to notify a system administrator of network risks once the risk has been identified. For example, Engelberg discloses providing “one or more dashboards, alerts, notifications, and the like to cyber security personnel that enable the cyber security personnel to react to and remediate security relevant events.” EX1006 (Engelberg), ¶42; EX1007 (Engelberg Provisional), ¶40. As shown below in Fig. 6, a dashboard may include a display conveying various risk notifications such as “Business Process Risk Assessment,” “Crown Jewel Risk Assessment,” “Risk Aspect,” and “Risk Tolerance.”

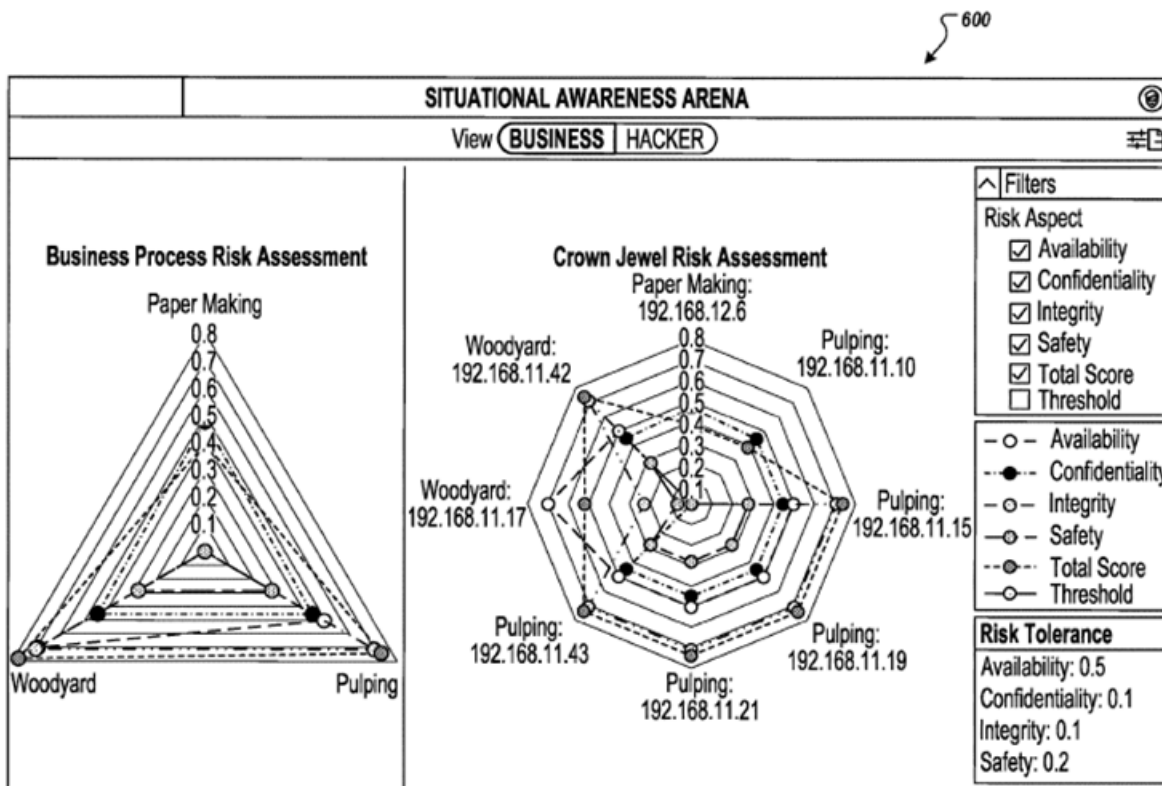


FIG. 6

EX1006 (Engelberg), Fig. 6; *see also id.*, ¶¶3 (“To defend against such attacks, enterprises use security systems to monitor occurrences of potentially adverse events occurring within a network, and alert security personnel to such occurrences. For example, one or more dashboards can be provided, which provide lists of alerts that are to be addressed by the security personnel.”), 7 (describing assessing “asset vulnerability” based on “potential for asset breach and criticality of attack paths towards target(s)”), 20 (“FIG. 6 depicts an example UI displaying an example process-aware risk assessment.”), 46 (using dashboards

“segmented to owners and relevant IT and security users ... security risk concern can be presented and evaluated in the operations-level”); EX1007 (Engelberg Provisional), ¶¶2, 6, 18, 44, Fig. 6.

140. Thus, consistent with the teachings of Keren, Engelberg teaches identifying cyber security threats, vulnerabilities, and other network and workload-related risks, and beneficially presenting them to a user as risk notifications. As I discussed above in Section XI.A, a POSA would have been motivated to report Keren’s security insights describing risky network configurations to a user as risk notifications in order to alert the user to the security insight for remediation consideration.

141. It is therefore my opinion that claims 1, 11, and 20 were obvious.

C. Claim 10

142. Claim 10 depends from claim 1 and further recites “wherein the risk notification includes one or more resolution recommendations.” The combination of Keren and Engelberg teaches this limitation.

143. I explained above in Ground 1 how Keren teaches that its system is configured to output a risk notification associated with security objects (e.g., workloads) contained within the cloud environment. *Supra*, §XI.A.9. Engelberg teaches analyzing risks associated with assets contained within a network environment, identifying “remediation recommendations and courses of action

(COA),” and conveying the information “to enable users (e.g., security teams) to understand immediate remediation actions corresponding to each issue.” EX1006 (Engelberg), ¶¶30, 55, 57, 59; EX1007 (Engelberg Provisional), ¶¶28, 53, 55, 57. Engelberg therefore teaches including one or more resolution recommendations with a risk notification, as recited.

144. As I explained above, a POSA would have had reason to include one or more resolution recommendations with a corresponding risk notification, as taught by Engelberg, as part of outputting generated security insights, as taught by Keren. *Supra*, §XII.A. Thus, the combination of Keren and Engelberg teaches a risk notification that includes one or more resolution recommendations, as recited.

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

XIII. CONCLUDING STATEMENTS

146. 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.

147. 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: 11/01/2024

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. 17/716,491 ('257 Pat.)
1005	U.S. Patent No. 11,374,982 to Keren
1006	U.S. Publication No. 2022/0263855 to Engelberg
1007	U.S. Provisional Application No. 63/150,664 to Engelberg
1008	Collier M. and Shahan R., “Fundamentals of Azure: Microsoft Azure Essentials,” 2 nd 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
1009	Amazon AWS, Application Load Balancer, available at: https://web.archive.org/web/20210220224806/https://aws.amazon.com/elasticloadbalancing/application-load-balancer/ , archived February 20, 2021
1010	Kubernetes, Ingress, available at: https://web.archive.org/web/20210110103139/https://kubernetes.io/docs/concepts/services-networking/ingress/ , archived January 10, 2021
1011	freeCodeCamp, What is a Proxy Server? In English, Please., available at: https://web.archive.org/web/20210421111614/https://www.freecodecamp.org/news/what-is-a-proxy-server-in-english-please/ , archived April 21, 2021

Exhibit No.	Description
1012	Fortinet, Proxy Server, available at: https://web.archive.org/web/20210424234713/https://www.fortinet.com/resources/cyberglossary/proxy-server , archived April 24, 2021
1013	Rapid7, Web Application Firewall, available at: https://web.archive.org/web/20210410115503/https://www.rapid7.com/fundamentals/web-application-firewalls/ , archived April 10, 2021
1014	W3schools, Ports and Its Security, available at: https://web.archive.org/web/20210412193421/https://www.w3schools.in/cyber-security/ports-and-its-security/ , archived April 12, 2021
1015	Cloudflare, Inc., What is a computer port? Ports in networking, available at: http://web.archive.org/web/20210303124521/https://www.cloudflare.com/learning/network-layer/what-is-a-computer-port/ , archived March 3, 2021
1016	Nginx, available at: http://web.archive.org/web/20210410094516/https://nginx.org/en/ , archived April 10, 2021
1017	SecurityHQ, Notes from the Field. Using Proxy Logs to Detect Suspicious Activity – Interview with Aaron Hambleton, available at: https://web.archive.org/web/20210416073813/https://www.securityhq.com/blog/using-proxy-logs-to-detect-suspicious-activity/ , archived April 16, 2021
1018	Apache HTTP Server Project, Apache HTTP Server Version 2.4, Log Files, available at: http://web.archive.org/web/20210303195600/https://httpd.apache.org/docs/2.4/logs.html , March 3, 2021

Exhibit No.	Description
1019	Amazon AWS, Logging web ACL traffic information, available at: http://web.archive.org/web/20201111183928/https://docs.aws.amazon.com/waf/latest/developerguide/logging.html , November 11, 2020
1020	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
1021	EP Patent No. 3,360,071 B1 to Li
1022	U.S. Patent No. 9,692,778 B1 to Mohanty
1023	U.S. Publication No. 2014/0245376 A1 to Hibbert
1024	U.S. Publication No. 2023/0086187 to Tan
1025	U.S. Publication No. 2013/0191919 A1, published July 25, 2013, to Basavapatna