

UNITED STATES PATENT AND TRADEMARK OFFICE

---

BEFORE THE PATENT TRIAL AND APPEAL BOARD

---

ORCA SECURITY LTD

Petitioner

v.

WIZ, INC.

Patent Owner

---

*Inter Partes* Review Case No. IPR2025-01086

U. S. Patent No. 12,001,549

**DECLARATION OF EREZ ZADOK**

## TABLE OF CONTENTS

|                                                                                                                                                                                      |           |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| <b><i>I. INTRODUCTION</i></b> .....                                                                                                                                                  | <b>5</b>  |
| <b>A. Background and Qualifications</b> .....                                                                                                                                        | <b>5</b>  |
| <b><i>II. OVERVIEW AND LEGAL STANDARDS</i></b> .....                                                                                                                                 | <b>18</b> |
| <b>A. Person of Ordinary Skill in the Art</b> .....                                                                                                                                  | <b>19</b> |
| <b>B. Obviousness</b> .....                                                                                                                                                          | <b>20</b> |
| <b>C. Analogous Art</b> .....                                                                                                                                                        | <b>25</b> |
| <b>D. Claim Construction</b> .....                                                                                                                                                   | <b>26</b> |
| <b><i>III. BACKGROUND OF TECHNOLOGY</i></b> .....                                                                                                                                    | <b>27</b> |
| <b>A. Cybersecurity Alert Detection and Response</b> .....                                                                                                                           | <b>27</b> |
| <b>B. Artificial Intelligence and LLMs</b> .....                                                                                                                                     | <b>28</b> |
| <b><i>IV. OVERVIEW OF THE '549 PATENT</i></b> .....                                                                                                                                  | <b>30</b> |
| <b>A. Summary of the '549 Patent</b> .....                                                                                                                                           | <b>30</b> |
| <b>B. Field of Endeavor</b> .....                                                                                                                                                    | <b>32</b> |
| <b>C. Problem Solved by the Inventor</b> .....                                                                                                                                       | <b>32</b> |
| <b>D. File History of the '549 Patent</b> .....                                                                                                                                      | <b>33</b> |
| <b><i>V. OVERVIEW OF THE PRIOR ART</i></b> .....                                                                                                                                     | <b>33</b> |
| <b>A. U. S. Patent No. 11,303,666 (“Peters”) (EX1004)</b> .....                                                                                                                      | <b>33</b> |
| <b>B. U. S. Patent Application Publication No. 2024/0403428 (“Lal Publication”)<br/>        (EX1005) and U. S. Provisional Patent Application No. 63/472,227 (“Lal”) (EX1006)...</b> | <b>37</b> |
| <b><i>VI. OPINIONS REGARDING GROUND 1</i></b> .....                                                                                                                                  | <b>38</b> |
| <b>A. Motivations to Combine Peters and Lal</b> .....                                                                                                                                | <b>39</b> |
| <b>B. Claim 1</b> .....                                                                                                                                                              | <b>46</b> |
| 1. (Pre): “A method for providing cybersecurity incident response, comprising:” .....                                                                                                | <b>46</b> |
| 2. Claim 1(a): “receiving an incident input based on a cybersecurity event;” .....                                                                                                   | <b>46</b> |
| 3. Claim 1(b): “generating a prompt for a large language model (LLM) based on the<br>received incident input;” .....                                                                 | <b>48</b> |
| 4. Claim 1(c): “configuring the LLM to generate an output based on the generated<br>prompt;” .....                                                                                   | <b>52</b> |

|           |                                                                                                                                                                                                                                                                                                                                                    |           |
|-----------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| 5.        | Claim 1(d): “mapping the received incident input into a scenario of a plurality of scenarios based on the output of the LLM, wherein each scenario is associated with an incidence response;” .....                                                                                                                                                | 53        |
| 6.        | Claim 1(e): “generating a query based on the received incident input and the mapped scenario;” .....                                                                                                                                                                                                                                               | 56        |
| 7.        | Claim 1(f): “executing the query on a security database, the security database including a representation of a computing environment; and” .....                                                                                                                                                                                                   | 57        |
| 8.        | Claim 1(g): “initiating a mitigation action based on a result of the executed query.” .....                                                                                                                                                                                                                                                        | 62        |
| <b>C.</b> | <b>Claim 2: “The method of claim 1, wherein the incident input includes any one of: a query, a statement, and a combination thereof.” .....</b>                                                                                                                                                                                                    | <b>65</b> |
| <b>D.</b> | <b>Claim 3: “The method of claim 1, wherein the LLM is trained on any one of: a data schema utilized in representing the computing environment, incident data classified to a scenario, the plurality of scenarios, and a combination thereof.” .....</b>                                                                                          | <b>67</b> |
| <b>E.</b> | <b>Claim 4: “The method of claim 3, wherein generating the query further comprises: generating a second prompt for the LLM which when executed by the LLM outputs the query, wherein the second prompt is generated based on any one of: the received incident input, the data schema, the plurality of scenarios, and a combination thereof.”</b> | <b>70</b> |
| <b>F.</b> | <b>Claim 5: “The method of claim 3, further comprising: training the LLM further on a plurality of database queries, each database query executable on the security database.” .....</b>                                                                                                                                                           | <b>74</b> |
| <b>G.</b> | <b>Claim 11 .....</b>                                                                                                                                                                                                                                                                                                                              | <b>78</b> |
| 1.        | Claim 11(Pre): “A non-transitory computer-readable medium storing a set of instructions for providing cybersecurity incident response, the set of instructions comprising: one or more instructions that, when executed by one or more processors of a device, cause the device to:” .....                                                         | 78        |
| 2.        | Claim 11(a): “receive an incident input based on a cybersecurity event;” .....                                                                                                                                                                                                                                                                     | 80        |
| 3.        | Claim 11(b): “generate a prompt for a large language model (LLM) based on the received incident input;” .....                                                                                                                                                                                                                                      | 80        |
| 4.        | Claim 11(c): “configure the LLM to generate an output based on a cybersecurity event;” .....                                                                                                                                                                                                                                                       | 80        |
| 5.        | Claim 11(d): “map the received incident input into a scenario of a plurality of scenarios based on the output of the LLM, wherein each scenario is associated with an incidence response;” .....                                                                                                                                                   | 80        |
| 6.        | Claim 11(e): “generate a query based on the received incident input and the mapped scenario;” .....                                                                                                                                                                                                                                                | 80        |
| 7.        | Claim 11(f): “execute the query on a security database, the security database including a representation of a computing environment; and” .....                                                                                                                                                                                                    | 80        |
| 8.        | Claim 11(g): “initiate a mitigation action based on a result of the executed query.” .....                                                                                                                                                                                                                                                         | 81        |

|             |                                                                                                                                                                                                                                                                                                                                                                                                                                                                |           |
|-------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| <b>H.</b>   | <b>Claim 12</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                | <b>81</b> |
| 1.          | Claim 12(Pre): “A system for providing cybersecurity incident response comprising:”                                                                                                                                                                                                                                                                                                                                                                            | 81        |
| 2.          | Claim 12(a): “a processing circuitry;”                                                                                                                                                                                                                                                                                                                                                                                                                         | 82        |
| 3.          | Claim 12(b): “a memory, the memory containing instructions that, when executed by the processing circuitry, configure the system to:”                                                                                                                                                                                                                                                                                                                          | 83        |
| 4.          | Claim 12(c): “receive an incident input based on a cybersecurity event;”                                                                                                                                                                                                                                                                                                                                                                                       | 84        |
| 5.          | Claim 12(d): “generate a prompt for a large language model (LLM) based on the received incident input;”                                                                                                                                                                                                                                                                                                                                                        | 84        |
| 6.          | Claim 12(e): “configure the LLM to generate an output based on a cybersecurity event;”                                                                                                                                                                                                                                                                                                                                                                         | 84        |
| 7.          | Claim 12(f): “map the received incident input into a scenario of a plurality of scenarios based on the output of the LLM, wherein each scenario is associated with an incidence response;”                                                                                                                                                                                                                                                                     | 85        |
| 8.          | Claim 12(g): “generate a query based on the received incident input and the mapped scenario;”                                                                                                                                                                                                                                                                                                                                                                  | 85        |
| 9.          | Claim 12(h): “execute the query on a security database, the security database including a representation of a computing environment; and”                                                                                                                                                                                                                                                                                                                      | 85        |
| 10.         | Claim 12(i): “initiate a mitigation action based on a result of the executed query.”                                                                                                                                                                                                                                                                                                                                                                           | 85        |
| <b>I.</b>   | <b>Claim 13: “The system of claim 12, wherein the incident input includes any one of: a query, a statement, and a combination thereof.”</b>                                                                                                                                                                                                                                                                                                                    | <b>85</b> |
| <b>J.</b>   | <b>Claim 14: “The system of claim 12, wherein the LLM is trained on any one of: a data schema utilized in representing the computing environment, incident data classified to a scenario, the plurality of scenarios, and a combination thereof.”</b>                                                                                                                                                                                                          | <b>85</b> |
| <b>K.</b>   | <b>Claim 15: “The system of claim 14, wherein the memory contains further instructions that, when executed by the processing circuitry for generating the query, further configure the system to: generate a second prompt for the LLM which when executed by the LLM outputs the query, wherein the second prompt is generated based on any one of: the received incident input, the data schema, the plurality of scenarios, and a combination thereof.”</b> | <b>86</b> |
| <b>L.</b>   | <b>Claim 16: “The system of claim 14, wherein the memory contains further instructions which when executed by the processing circuitry further configure the system to: train the LLM further on a plurality of database queries, each database query executable on the security database.”</b>                                                                                                                                                                | <b>86</b> |
| <b>VII.</b> | <b>CONCLUSION</b>                                                                                                                                                                                                                                                                                                                                                                                                                                              | <b>87</b> |

I, Erez Zadok, declare as follows:

## **I. INTRODUCTION**

1. My name is Erez Zadok, and I am over 21 years of age and otherwise competent to make this Declaration. I make this Declaration based on facts and matters within my own knowledge and on information provided to me by others, and if called as a witness, I could and would competently testify to the matters set forth herein.

2. I have been retained as an expert in this matter by Counsel for Orca Security Ltd. (“Petitioner”) to provide my independent opinions on certain issues requested by Counsel for Petitioner relating to the accompanying petition for Inter Partes Review of U. S. Patent No. 12,001,549 (“the ’549 Patent”).

3. The compensation my company receives in this matter is not based on the substance of my opinions or the outcome of this matter. I have no financial interest in Petitioner. I am being compensated at my standard hourly rate for my analysis and testimony in this case.

### **A. *Background and Qualifications***

4. I am a Professor in the Computer Science Department at Stony Brook University (part of the State University of New York (“SUNY”) system). I direct the File-systems and Storage Lab (FSL) at Stony Brook’s Computer Science Department. My research interests include file systems and storage systems,

operating systems, transactional systems including database technologies, information technology and system administration, security/privacy and information assurance, networking, energy efficiency, performance and benchmarking, virtualization, cloud systems, compilers, applied machine learning, and software engineering.

5. I studied at a professional high school in Israel, focusing on electrical engineering (“EE”), and graduated in 1982. I spent one more year at the high school’s college division, receiving a special Certified Technician’s degree in EE. I then went on to serve in the Israeli Defense Forces for three years (1983–1986). I received my Bachelor of Science degree in computer science (“CS”) in 1991, my Master’s degree in CS in 1994, and my PhD in CS in 2001—all from Columbia University in New York.

6. When I began my undergraduate studies at Columbia University, I also started working as a student assistant in the various campus-wide computer labs, eventually becoming an assistant to the head labs manager, who was managing all public computer labs on campus. During that time, I also became more involved with research within the CS Department at Columbia University, conducting research on operating systems, file and storage systems, distributed and networked systems, security, and other topics. I also assisted the CS department’s computer

administrators in managing the department's computers, which included storage, IT, networking, and cyber-security related duties.

7. In 1991, I joined Columbia University's CS department as a full-time systems administrator, studying towards my MS degree part-time. My MS thesis topic is related to file system reliability, fault tolerance, replication, and failover in mobile networked storage systems using file virtualization. My main duties as a systems administrator involved installing, configuring, and managing many networked servers, proxies, and desktops running several operating systems, as well as network devices setup; this included many software and hardware upgrades, device upgrades, and BIOS firmware/chipset updates/upgrades. My duties also included ensuring reliable, secure, authenticated access to networked systems/storage and licensed software, as well as software updates, security and bug fixes. Examples of servers and their protocols included email (SMTP), file transfer (FTP), domain names (DNS), network file systems (NFS), network news systems (NNTP), and Web (HTTP).

8. In 1994, I left my systems administrator position to pursue my doctoral studies at Columbia University. My PhD thesis topic was on versatile file system development using stackable (virtualized) file systems, with examples in the fields of security and encryption, efficiency, reliability, and failover. I continued to work part-time as a systems administrator at the CS department, and eventually I was

asked to serve as manager to the entire information technology (“IT”) staff. From 1991 to 2001, I was also a member of the faculty-level Facilities Committee that oversaw all IT operations at the CS department.

9. As part of my PhD studies at Columbia, I collaborated on projects to develop advanced AI-like techniques to detect previously unknown viruses (a.k.a. “zero-day malware”), using data mining and rule-based detection. This work led to several highly cited papers (over 1,600 citations for one of the papers alone) and two patents. I also became a Teaching Assistant (“TA”) for a first-ever Computer Security course given at Columbia University’s CS department with Dr. Matt Blaze as instructor.

10. From 1990 to 1998, I consulted for SOS Corporation and HydraWEB Technologies as a systems administrator and programmer, managing data storage use and backup/restore duties, databases, web servers, as well as information assurance and cyber-security (*e.g.*, malware protection, software licensing). From 1994 to 2000, I led projects at HydraWEB Technologies, and then became the Director of Software Development—overseeing the development of several products and appliances such as stateful firewalls and HTTP load-balancers, utilizing network-virtualization and high-availability techniques. From 2009 to 2019, I have consulted for Packet General Networks, a startup specializing in secure, virtualized, network storage and applications’ data security in the cloud.

11. In 2001, I joined the faculty of Stony Brook University, a position I have held since that time. In 2002, I joined the Operations Committee, which oversees the IT operations of the CS department at Stony Brook University. From 2006 to 2010, I was the Director of IT Operations of the CS department. My day-to-day duties included setting policies regarding computing, hiring and training new staff, assisting any staff with topics of my specialty, defining requirements for new software/hardware, and purchasing. From 2010 to 2015, I had served as the Co-Chair to the Operations Committee. From 2016 to 2019, I oversaw the IT Operations as the Chair of the Operations Committee. A significant component of these duties included defining and helping implement policies for data management, to ensure the security of users and their data, and data reliability and availability, while minimizing the inconvenience and performance impact to users. I personally helped setup and maintain an initial virtual-host infrastructure in the department. Since late 2019, I've been a member of the department's Executive Committee that also oversees all IT operations.

12. In 2017, I became the department's Graduate Academic Adviser, advising all Master students (over 400 annually on average) and many other graduate students on an assortment of academic matters. In 2024, I took over as the department's Graduate Program Director, overseeing the entire graduate CS program (700-800 students annually on average).

13. Since 2001, I have personally configured and managed my own research lab's network. This includes setting up and configuring multiple storage systems (*e.g.*, NFS, CIFS/SMB, NAS), virtual and physical environments, applications such as database (*e.g.*, MySQL, PostgreSQL), Web servers (*e.g.*, Apache), and mail servers; user access control (*e.g.*, NIS, LDAP, Active Directory), backups and restores, snapshot policies, and more. I have personally installed, configured, changed, replaced parts, and upgraded components in numerous devices including mobile devices, laptops, desktops, and servers, both physical and virtual.

14. Since 1995, I have taught courses on operating systems, storage and file systems, advanced systems programming in Unix/C, systems administration, data structures, data/software security, and more. My courses often use storage, file systems, distributed systems, and system/network security as key teaching principles and practical examples for assignments and projects. I have taught these concepts and techniques to my students, both to my direct advisees as well as in my courses. For example, in my graduate Operating Systems course, I often cover Linux's kernel mechanisms to protect users, applications, and data files, virtual file systems, as well as distributed storage systems (*e.g.*, NFS). And in the System Administration undergraduate course, I covered many topics such as networking, storage, backups, and configuring complex applications such as mail, web, and database servers.

15. My research often investigates computer systems from many angles: security, efficiency, energy use, scalability, reliability, portability, survivability, usability, ease-of-use, versatility, flexibility, and more. My research gives special attention to balancing five often-conflicting aspects of computer systems: performance, reliability, energy use, security, and ease-of-use.

16. Since joining Stony Brook University in 2001, my group in the File-systems and Storage Lab (“FSL”) has developed many file systems and operating system extensions; examples include a highly-secure cryptographic file system, a portable copy-on-write (“COW”) versioning file system, a tracing file system useful to detect intrusions, a replaying file system useful for forensics, a snapshotting and sandboxing file system, a namespace unification file system (that uses stackable, virtualized, file-based COW), an anti-virus file system, an integrity-checking file system, a load balancing and replication/mirroring file system, network file system extensions for security and performance, distributed secure cloud-based storage systems, transactional key-value stores and file systems, OS-level embedded databases, a compiler to convert user-level C code to in-kernel efficient yet safe code, GCC plugins, stackable file system templates, and a Web-based backup system. Many of these projects used one form of virtualization or another (storage, network, host, etc.). I continue to maintain and release newer versions of some of these file systems and software.

17. I have published over 120 refereed publications (in ACM, IEEE, USENIX, and more). To date, my publications have been cited more than 10,000 times (as per Google Scholar as of May 30, 2025). My papers cover a wide range of related technologies such as file systems, storage systems, transactional systems, security, clouds and virtualization, performance benchmarking and optimization, energy efficiency, system administration, web systems, and more. I also published a book titled “Linux NFS and Automounter Administration” (Sybex, 2001), covering systems administration topics related to network storage and data security.

18. Some of my research has led to public software releases that have been used worldwide. I have publicly maintained the Amd Berkeley Automounter in a package called “am-utils” since 1992; this software helps administrators manage the multitude of file system mounts on dozens of different Unix systems, especially helping to automate access to multiple NFS/NAS storage volumes. Since 1997, I have maintained and released several stackable (virtualized) file system software projects for Linux, FreeBSD, and/or Sun Solaris, in a package called FiST. One of my stackable file system encryption projects, called Cryptfs, became the basis for IBM’s public release of eCryptfs, now part of Linux. Packet General Networks, for whom I have provided consulting services between 2009 and 2019, licensed another encryption file system called Ncryptfs. Another popular file system released in 2003, called Unionfs, offers virtual namespace unification, transparent shadow

copying (a.k.a. copy-on-write or COW), file system snapshotting (*e.g.*, useful for forensics and disaster recovery), and the ability to save disk space by sharing a read-only copy of data across several computers, among other features.

19. My research and teaching make extensive use of data security features. For example, each time I taught the graduate operating system course, the first homework assignment includes the creation of a new system call that performs new or added functionality, often for encrypting a file or verifying its integrity; many of my other assignments cover topics of user/process access control, anti-virus filtering, and more. Since 2001, over 1,000 graduate students were exposed to these principles directly through my teaching and research at Stony Brook University.

20. Moreover, in an undergraduate course titled “Advanced Systems Programming in Unix/C,” I cover many topics of system security and vulnerabilities, such as the structure of UNIX processes, and memory segments such as the heap and stack. This course covers details of several hundred Linux system calls. Often, the first assignment for this course is to develop a tool to encrypt/decrypt files using advanced ciphers, use digital signatures to certify the cipher keys used, and reliably recover files in case of failures. Since 2001, several hundred undergraduate students were exposed to these principles directly through my teaching and research at Stony Brook University.

21. In another undergraduate course, System Administration, I taught network configuration, security, and storage configuration and reliability. In a special topics course on Storage Systems, I covered many topics such as data deduplication, RAID, transactional storage, storage hardware including modern Flash based ones, virtual storage, backup/restore, snapshots and continuous data protection (“CDP”), NAS and SAN, and NFS.

22. Overall, in addition to the aforementioned experience, my technical experience relevant to the '549 Patent at the time of the alleged invention included the following. With respect to storage and disks, I've conducted extensive research, published, taught, managed systems, and released software related to storage media (*e.g.*, disks), file systems, and networked and cloud based storage systems. This included copy-on-write (CoW) and snapshotting systems. Much of my work relates to security of data, including anti-virus scanning of file systems, encrypted storage, integrity checking, and more. With respect to AI experience, I first studied AI in the late 1980s. In the late 1990s, I collaborative on early generation AI-assisted research to detect zero-day malware; this work led to several highly cited publications as well as two patents. Since then I integrated AI techniques into my research and teaching, including applied machine learning (*e.g.*, neural networks, decision trees, and more), meta-heuristics such genetic algorithms and simulated annealing, statistical regression-based techniques, and more. In 2018 I was part of a National Science

Foundation (NSF) multi-day summit to chart the course of AI and ML techniques in systems and storage research (a report was published). As AI grew more in popularity, I began to collaborate and even lead industry, state, and federally funded projects involving AI/ML techniques. In one example, I led a team that used advanced Natural Language Processing (NLP) techniques to solve complex networked file/storage systems; as part of that work, the team integrated Large Language Models (LLMs) into the project. Several of my publications are in the fields of ML systems, NLP, and visual analytics.

23. My research has been supported by many federal and state grants as well as industry awards, including an NSF CAREER award, two IBM Faculty awards, two NetApp Faculty awards, a Western Digital award, a Facebook award, several Dell-EMC awards, and several equipment gifts. I received the 2008 SUNY Chancellor's Excellence in Teaching award, and the 2022 SUNY Chancellor's Award for Excellence in Scholarship and Creative Activities (both awards can be given only once in a lifetime). In 2021, I was named an ACM Distinguished Member for "Outstanding Scientific Contributions to Computing."

24. My service record to the community includes serving as the co-chair for the USENIX Annual Technical Conference in 2020 (ATC'20); serving as the co-chair for USENIX File and Storage Technologies (FAST'15) in 2015 and on the FAST Conference Steering Committee from 2015 to 2023; serving on the ACM

HotStorage Steering Committee since 2021; and serving as the co-chair in 2012 and on the Steering Committee of the ACM SYSTOR conference since 2012. I have served as an Associate Editor to the ACM Transactions on Storage (“TOS”) journal from 2009 to 2022; in 2022, I was named the Editor-in-Chief for ACM’s TOS journal.

25. I am a named inventor on four patents, two titled “Systems and Methods for Detection of New Malicious Executables” (U.S. Patent No. 7,487,544, issued February 3, 2009; and U.S. Patent No. 7,979,907, issued July 12, 2011); and two more titled “Multi-Tier Caching,” (U.S. Patent No. 9,355,109, issued May 31, 2016; and U.S. Patent 9,959,279, issued May 1, 2018).

26. I have been disclosed as a testifying expert in 19 cases (including *inter partes* review (“IPR”) proceedings) in the past four years. I have been deposed 13 times and testified in trial twice.

27. A complete copy of my curriculum vitae, including further details on my education, experience, publications, patents, and other qualifications to render an expert opinion, is included as EX1009.

28. In writing this declaration, I have considered my own knowledge and experience, including my work experience in the fields of software engineering, cybersecurity, artificial intelligence and machine learning, and computer networking; my experience teaching those subjects; my experience consulting in

those subjects; and my experience creating standards and certifications for those subjects. In reaching my opinions in this matter, I have focused on the following references and materials, which I have considered from the viewpoint of a POSITA as of the earliest possible priority date of the '549 Patent:

| Exhibit | Description                                                                                                |
|---------|------------------------------------------------------------------------------------------------------------|
| 1001    | U. S. Patent No. 12,001,549                                                                                |
| 1002    | File History of U.S. Patent No. 12,001,549—Application No. 18/428,794                                      |
| 1003    | Declaration of Erez Zadok, Ph. D. (this document)                                                          |
| 1004    | U. S. Patent No. 11,303,666 to Peters et al. (“Peters”)                                                    |
| 1005    | U. S. Patent Application Publication No. 2024/0403428 (“Lal Patent Application Publication”)               |
| 1006    | U. S. Provisional Patent Application No. 63/472,227 (“Lal”)                                                |
| 1007    | Complaint from <i>Orca Security, Ltd. v. Wiz, Inc.</i> , No. 23-758-JHL-SRF (D. Del.)                      |
| 1008    | Order Issuing Stay from <i>Orca Security, Ltd. v. Wiz, Inc.</i> , No. 23-758-JHL-SRF (D. Del.)             |
| 1009    | Curriculum Vitae of Erez Zadok, Ph.D.                                                                      |
| 1010    | Joint Claim Construction from <i>Orca Security, Ltd. v. Wiz, Inc.</i> , No. 23-758-JHL-SRF (D. Del.)       |
| 1011    | Jules White et al., <i>A Prompt Pattern Catalog to Enhance Prompt Engineering with ChatGPT</i>             |
| 1012    | Simon Arvidsson and Johan Axell, <i>Prompt Engineering Guidelines for LLMs in Requirements Engineering</i> |
| 1013    | Microsoft, <i>Microsoft Active Directory Domain Services Overview</i> , Web Archive (August 16, 2022)      |
| 1014    | U. S. Patent Pub. No. 2018/004870 to Ohayon et al.                                                         |
| 1015    | U. S. Patent No. 8,146,146 to Coviello et al.                                                              |
| 1016    | November 30, 2022, OpenAI Announcement Regarding ChatGPT’s Public Release                                  |
| 1017    | RESTAPI Specification, Web Archive (February 16, 2023)                                                     |
| 1018    | JSON Specification, Web Archive (May 24, 2023)                                                             |
| 1019    | Merriam Webster, Dictionary Definition of “Network”                                                        |
| 1020    | Microsoft, <i>Passwords Technical Overview</i>                                                             |

| <b>Exhibit</b> | <b>Description</b>                                                                                        |
|----------------|-----------------------------------------------------------------------------------------------------------|
| 1021           | Microsoft, <i>Active Directory Security Groups</i> , Web Archive (May 21, 2023)                           |
| 1022           | Toloka, <i>Empowering AI Through LLM Training</i>                                                         |
| 1023           | Isabelle Nguyen, <i>What Is a Large Language Model (LLM)?</i>                                             |
| 1024           | Dorian Drost, <i>Different Ways of Training LLMs</i>                                                      |
| 1025           | Angie Lee, <i>No Hang Ups with Hangul: KT Trains Smart Speakers, Customer Call Centers with NVIDIA AI</i> |
| 1026           | Greyling, <i>12 Prompt Engineering Techniques</i>                                                         |
| 1027           | Kojima, <i>Large Language Models are Zero-Shot Reasoners</i>                                              |
| 1028           | Loem, <i>Mastering Prompt Engineering for Effective LLM Output: Tips, Techniques, and Warning</i>         |

29. Exhibits 1008-1028 are true and correct copies of the identified documents that I personally retrieved, received, copied, own, and/or scanned for purposes of submission with this IPR proceeding. I have also reviewed any other materials I refer to in this declaration in support of my opinions.

## **II. OVERVIEW AND LEGAL STANDARDS**

30. In formulating my opinions, I have been instructed to apply certain legal standards. I am not a lawyer. I do not expect to offer any testimony regarding what the law is. Instead, the following sections summarize the law as I have been instructed to apply it in formulating and rendering my opinions found later in this declaration. I understand that, in an inter partes review proceeding, patent claims may be deemed unpatentable if it is shown that they were anticipated or rendered obvious in view of the prior art. I understand that prior art in an inter partes review is limited to patents or printed publications that predate the priority date of the patent

at issue. I understand that questions of claim clarity (definiteness) and enablement cannot be considered as a ground for considering the patentability of a claim in these proceedings.

**A. *Person of Ordinary Skill in the Art***

31. My opinions are provided based on what a person of ordinary skill in the art in the technical field of the invention would have understood at the time of the purported invention of the '549 patent. I have been asked to assume the date of the purported invention for the '549 patent is August 28, 2023 (the “earliest possible priority date”).

32. I have been informed and understand that the content of a patent and prior art should be interpreted the way a person of ordinary skill in the art would have interpreted those references at the time of the invention of the patent. I understand that the person of ordinary skill is a hypothetical concept referring to one who thinks along the lines of conventional wisdom at the time. I also understand that the factors considered in determining the ordinary level of skill in the art may include: (i) the levels of education of the inventor; (ii) the types of problems encountered in the art; (iii) prior art solutions to those problems; (iv) the rapidity with which innovations are made; (v) the sophistication of the technology; and (vi) the educational level of persons working in the field.

33. I was asked to provide an opinion as to the level of one of ordinary skill in the art pertinent to the subject matter set forth in the '549 patent at the time of the earliest possible priority date. When I refer to the understanding of a POSITA, I am referring to the understanding of such a person as of August 28, 2023.

34. A POSITA at the time of the '549 Patent would have had a bachelor's degree in computer engineering, computer science, electrical engineering, or a similar degree, with about 2-3 years of experience in the fields of cybersecurity and artificial intelligence; additional education or experience might substitute for these requirements.

35. As of August 28, 2023, I had more than ordinary skill in the art. I am, however, familiar with the skills and knowledge possessed by those I would have considered to be of ordinary skill in the art as of that date. For example, I knew people working in the field while I earned my academic degrees, and I also contributed to the field through at least literature and inventions, as I discussed in my Background (§I.A above) and demonstrated in my CV (EX1009).

36. My opinions provided in this declaration would not change in view of minor modifications to this level of ordinary skill.

**B. *Obviousness***

37. I have been informed and understand that subject matter claimed in a patent is obvious under 35 U. S. C. §103 if a person of ordinary skill in the art at the

time that the alleged invention was made would have had reason to combine or modify the disclosures of one or more prior art references to arrive at the claimed subject matter. A claim may be unpatentable even if each and every claim limitation is not present or disclosed in a single prior-art item.

38. I have been informed and understand that, under the doctrine of obviousness, a claim is unpatentable if the differences between the invention and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which the subject matter pertains. A person of ordinary skill in the art is presumed to know about the relevant prior art at the time of the claimed invention.

39. I have been informed and understand that obviousness is based on the scope and content of the prior art, the differences between the prior art and the claim, the level of ordinary skill in the art, and secondary indicia of obviousness and non-obviousness to the extent such indications exist.

40. I have been informed and understand the determination of whether the asserted claims would have been obvious to a person of ordinary skill in the art and, therefore, invalid, is not governed by any rigid test or formula. A determination that a claim is obvious is, instead, based on a common-sense determination that the claimed invention is merely a combination of known limitations to achieve predictable results. Any of the following rationales are acceptable justifications to

conclude that a claim would have been obvious: (1) the claimed invention is simply a combination of known prior art methods to yield predictable results; (2) the claimed invention is a simple substitution of one known element for another to obtain predictable results; (3) the claimed invention uses known techniques to improve similar devices (methods, or products) in the same way; (4) the claimed invention applies a known technique to a known device (method, or product) ready for improvement to yield predictable results; (5) the claimed invention was “obvious to try”—choosing from a finite number of identified, predictable solutions, with a reasonable expectation of success; (6) there is known work in one field of endeavor that may prompt variations of it for use in either the same field or a different one based on design incentives or other market forces if the variations would have been predictable to one of ordinary skill in the art; or, (7) there is some teaching, suggestion, or motivation in the prior art that would have led one of ordinary skill in the art to modify the prior art reference to combine prior art teachings to arrive at the claimed inventions.

41. I have been informed and understand that a claim may be obvious in light of a single reference without the need to combine references and that any elements of the claim that are not found in the reference (if any) can be supplied by the common sense or knowledge of one of ordinary skill in the art or taught in different areas of the single reference.

42. I have been informed and understand that an analysis of whether a claimed invention is obvious must not rely on a hindsight combination of prior art. The analysis must proceed in the context of the time of the invention or claimed priority date and consider whether the invention as a whole would have been obvious to a person of ordinary skill in the art, taking into consideration any interrelated teachings of the prior art, the effects of demands known to the design community or present in the marketplace, and the background knowledge possessed by a person having ordinary skill in the art, all to determine whether there was an apparent reason to combine any known elements in the fashion claimed by the patent at issue.

43. I have been informed that it is improper to combine references where the references teach away from their combination. I am informed that a reference may be said to teach away when a person of ordinary skill in the relevant art, upon reading the reference, would be discouraged from following the path set out in the reference, or would be led in a direction divergent from the path that was taken by the patent applicant. In general, a reference will teach away if it suggests that the line of development flowing from the reference's disclosure is unlikely to be productive of the result sought by the patentee. I am informed that a reference teaches away, for example, if (1) the combination would produce a seemingly inoperative device, or (2) the references leave the impression that the product would not have the property sought by the patentee. I also am informed, however, that a

reference does not teach away if it merely expresses a general preference for an alternative invention but does not criticize, discredit, or otherwise discourage investigation into the invention claimed.

44. I have been informed that even if a prima facie case of obviousness is established, the final determination of obviousness must also consider “secondary considerations” if presented. In most instances, the patentee raises these secondary considerations of non-obviousness. In that context, the patentee argues that an invention would not have been obvious in view of these considerations, which include: (a) commercial success of a product due to the merits of the claimed invention; (b) a long-felt, but unsatisfied need for the invention; (c) failure of others to find the solution provided by the claimed invention; (d) deliberate copying of the invention by others; (e) unexpected results achieved by the invention; (f) praise of the invention by others skilled in the art; (g) lack of independent simultaneous invention within a comparatively short space of time; (h) teaching away from the invention in the prior art.

45. I have further been informed that secondary considerations evidence is only relevant if the offering party establishes a connection, or nexus, between the evidence and the claimed invention. The nexus cannot be based on prior art features. The establishment of a nexus is a question of fact. While I understand that the Patent Owner here has not offered any secondary considerations at this time, I reserve the

right to supplement my opinions in the event that the Patent Owner raises secondary considerations during the course of this proceeding.

46. Finally, I understand that obviousness in an IPR must be proven by a preponderance of the evidence.

**C. *Analogous Art***

47. I have been informed that for a prior art reference to be proper for use in an obviousness analysis, the reference must be “analogous art” to the claimed invention. I have been informed that a reference is analogous art to the claimed invention if: (1) the reference is from the same field of endeavor as the claimed invention (even if it addresses a different problem); or (2) the reference is reasonably pertinent to the problem faced by the inventor (even if it is not in the same field of endeavor as the claimed invention). In order for a reference to be “reasonably pertinent” to the problem, it must logically have commended itself to an inventor’s attention in considering his problem. In determining whether a reference is reasonably pertinent, one should consider the problem faced by the inventor, as reflected either explicitly or implicitly, in the specification. I believe that all of the references that my opinions in this IPR are based upon are well within the range of references a person having ordinary skill in the art would consult to address the type of problems described in the Challenged Claims.

**D. Claim Construction**

48. I understand that, for purposes of my analysis in this inter partes review proceeding, the terms appearing in the patent claims should be interpreted according to their “ordinary and customary meaning.” In determining the ordinary and customary meaning, I understand that the words of a claim are first given their plain meaning that those words would have had to a person of ordinary skill in the art at the time of the alleged invention. I also understand that the structure of the claims, the specification and file history may be used to better construe a claim. Moreover, I understand that even treatises and dictionaries may be used, albeit under limited circumstances, to determine the meaning attribute by a person of ordinary skill in the art to a claim term.

49. I have also been informed that both Petitioner and Patent Owner have provided the following constructions for two terms of the ’549 Patent in related litigation, though Petitioner does not contend that any claim construction is necessary in its Petition for *Inter Partes Review* of the ’549 Patent:

| <b>Term/Phrase</b>                                               | <b>Petitioner’s Proposed Construction</b>                                         | <b>Patent Owner’s Proposed Construction</b>                                                  |
|------------------------------------------------------------------|-----------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------|
| “wherein each scenario is associated with an incidence response” | Indefinite                                                                        | “incidence” should read “incident”.<br>Otherwise, no construction is necessary.              |
| “based on”                                                       | “dependent on”; the limitations of claims 1, 11 and 12 must be performed in order | No construction is necessary.<br>Alternatively, “influenced by”; the limitations of a method |

| Term/Phrase | Petitioner's Proposed Construction | Patent Owner's Proposed Construction             |
|-------------|------------------------------------|--------------------------------------------------|
|             |                                    | claim are not required to be performed in order. |

EX1010, Amended Joint Claim Construction Chart, 15.

50. I have also been informed that the district court's *Markman* hearing is "CONTINUED" and has not been rescheduled in view of Patent Owner's instituted IPRs and related stay. *See Orca Security Ltd. v. Wiz, Inc.*, 1:23-cv-00758-JLH-SFR, Dkt. 223, Oral Order (Dec. 17, 2024).

51. Regardless, it is my opinion that the proposed ground below renders the Challenged Claims obvious under each party's proposed construction of "based on," and under Patent Owner's Proposed Construction of "wherein each scenario is associated with an incidence response."

### III. BACKGROUND OF TECHNOLOGY

52. In the following sub-sections, I briefly describe technology relevant to the '549 Patent.

#### A. *Cybersecurity*

53. Cybersecurity generally involves protecting computer systems, networks, and data from unauthorized access, use, or other disruption. While cybersecurity has been recognized as a field for many years, as computer technologies and communication networks have grown in attack sophistication and

use over the last several decades, so too have the cybersecurity systems protecting them.

54. The first step in maintaining the security of a chosen network is identifying a potential breach. Traditionally, after a breach is identified, classified, and analyzed, a human cybersecurity analyst or administrator would make decisions on how to best mitigate or resolve the ongoing cybersecurity breach or threat. This approach requires a lot of human interaction, highly specialized experience concerning breaches and responses thereto, and network-wide knowledge.

55. For these reasons, artificial intelligence has often been incorporated into cybersecurity protocols with the aim of increasing scalability and efficiency of responding to cybersecurity threats more quickly.

## **B. *Artificial Intelligence and LLMs***

56. Generally, Artificial Intelligence (AI) is an umbrella term that captures technologies that allow computers and machines to exercise human-reasoning. AI is an incredibly broad field containing many subfields, like machine learning, natural language processing, and speech recognition. AI research is also very old, despite its growing popularity in recent years; AI research began in the mid-1950s.

57. Within the Machine Learning (ML) subfield of AI exists Large Language Models, or LLMs. LLMs, specifically, are “**machine learning algorithms** that can recognize, understand, predict and generate human languages

based on **huge text datasets.**” EX1025, Lee, 1 (emphases added). In fact, LLMs are denoted “large” in part because they are trained on such expansive swaths of text data. LLMs are a type of an AI ML model, albeit trained on a large corpus of information. Generally, ML models can make better predictions when they are trained on more data; thus, for decades, AI researchers have sought to train ever-growing models.

58. LLMs have been around for years, with examples of Open AI’s GPT-2 model released in 2019, Facebook’s RoBERTa released in 2019, and Microsoft’s DeBERTa released in 2020. Most notable was the release of Open AI’s ChatGPT chatbot in late 2022. The release of ChatGPT spurred public knowledge and interest in LLMs and AI more broadly.

59. To generate an output, LLMs must receive prompts, *i.e.*, input instructions from the user. These prompts then direct the LLM to perform one or more tasks like summarization, text generation, or translation, among many others. Prompting is crucial for LLM performance and directly impacts the LLM’s output. Due to this importance, many different prompting techniques have been developed to further refine LLM outputs. *See* EX1026, Greyling (explaining various prompting techniques); *see also* EX1027, Kojima, 3–4 (describing “prompting twice to extract both reasoning and answer[.]”).

60. Rather than providing a single prompt to elicit an answer from the LLM, users frequently engage in several rounds of prompting to guide the LLM toward more accurate or useful outputs. For example, a first prompt can be a short instruction, like “Summarize the following text.” EX1028, Loem, 4. Then, if desired, a user can iteratively refine the prompt. *See id.*, 3. For example, a second prompt like “Summarize the following text in a chart, keeping in mind that the audience is composed of highly educated experts in the field of marine biology” would encourage the LLM to produce a different output more attuned to the user’s intent.

61. In such “conversational” LLMs, “the input to the model at each time step is the previous output and the new user input... The model is therefore constantly being fed the previous outputs and the prompt[.]” EX1011, White, 13. In other words, LLMs are iterative learners that consider older prompts and outputs alongside newly input prompts to generate a realized output with even greater accuracy and specificity. Therefore, with sufficient training and intelligent prompting, LLMs can produce highly tailored and specialized outputs.

#### **IV. OVERVIEW OF THE ’549 PATENT**

##### ***A. Summary of the ’549 Patent***

62. The ’549 Patent “relates generally to cybersecurity incident response and specifically to initiating mitigation actions in response to detected cybersecurity

threats.” EX1001, ’549 Patent, 1:16-18. For example, the ’549 Patent describes “[a] system and method for providing cybersecurity incident response utilizing a large language model.” *Id.*, Abstract.

63. The ’549 Patent’s described system begins by receiving a cybersecurity “incident input based on a cybersecurity event.” *Id.*, 4:23-24. An incident input, according to the ’549 Patent, may take the form of “a natural language query, an unstructured query, an alert, a combination thereof, and the like.” *Id.*, 7:40-44. Once this input is received, it is processed by a large language model (LLM) to identify and associate the input with a relevant cybersecurity scenario or risk. *See id.*, claims 1, 11, 12. To do so, a prompt is generated for the LLM “based on the received incident input” from which the LLM “generate[s] an output based on the generated prompt.” *Id.*, 4:25-28. The incident input is then mapped “into a scenario of a plurality of scenarios” based on the LLM’s output. *Id.*, 4:29-31.

64. Such a scenario or risk can include “a workload with an exposed database [or] a workload connected to a public network[.]” *Id.* Following this mapping, the LLM formulates a query targeting a security database, “the security database including a representation of a computing environment...” *Id.*, 3:8-10, 4:35-37, 16:20, 16:41-43. This query is then executed, and its results cause the initiation of a corresponding mitigation action. *See id.*, 16:41-18, 17:15-21, FIG. 6.

**B. *Field of Endeavor***

65. I have been informed that the field of endeavor of the claimed invention can be determined by reference to explanations of the invention's subject matter in the patent application, including the embodiments, function, and structure of the claimed invention.

66. Based on this understanding, it is my opinion that the '549 Patent's field of endeavor is cybersecurity threat mitigation. *See e.g.*, EX1001, '549 Patent, Title, Technical Field, cl. 1.

**C. *Problems Sought to Be Solved by the Inventor***

67. The '549 Patent sought to solve problems associated with "natural language processing techniques" used in cybersecurity solutions. Some of these problems include the "reliance on statistics of what other users search for" which results in users receiving an "alert that lacks context and information" and that "does not provide a root cause, or indicate what, if at all, should be done to remediate, mitigate, and the like." *Id.*, 1:40-55. The '549 Patent also sought to solve the problem of analyzing large amounts of data in a network environment, including information in a machine-readable format that may not be readily understood by a human, in order to make decisions regarding potential cybersecurity threats. *Id.*, 1:23-58.

**D. *File History of the '549 Patent***

68. I have read and reviewed the '549 Patent's file history. *See* EX1002, '549 Patent File History. According to my review, the first and only office action issued against the '549 Patent identified allowable subject matter but rejected all claims (1-21 as originally presented) on grounds of nonstatutory double patenting in light of claims from Application No. 18/466,832. *Id.*, 131.

69. Patent Owner then filed a terminal disclaimer. *Id.*, 118–20. After doing so, the application was allowed by the Examiner, who found that the prior art did not teach “wherein each scenario is associated with an incidence response; generating a query based on the received incident input and the mapped scenario; executing the query on a security database.” *Id.*, 16.

70. In my opinion, the alleged point of novelty identified by the Examiner is disclosed by Peters in view of Lal, as detailed in §VI below.

**V. OVERVIEW OF THE PRIOR ART**

**A. *U. S. Patent No. 11,303,666 (“Peters”) (EX1004)***

71. U. S. Patent No. 11,303,666, titled “Systems and Methods for Intelligent Cyber Security Threat Detection and Mitigation Through an Extensible Automated Investigations and Threat Mitigation Platform,” issued on April 12, 2022, from a provisional application filed on October 14, 2020.

72. In my opinion, Peters is analogous art to the '549 Patent because it is in the same field of endeavor as the '549 Patent and is reasonably pertinent to at least one problem that the inventors of the '549 Patent were involved with. As explained in §IV.B above, it is my opinion that the field of endeavor of the '549 Patent is cybersecurity threat mitigation. In my opinion, Peters is in the same field of endeavor as the '549 Patent. Namely, Peters describes a cybersecurity detection and response system that incorporates machine learning. *See e.g.*, EX1004, Peters, 5:36-45, 6:5-17, 10:1-12, 10:14-21, 11:8-12, 12:41-47.

73. Peters is also reasonably pertinent to a particular problem addressed by the '549 Patent, like “problems relating to storing, accessing, deleting, and managing [more and more] data.” *See* EX1001, '549 Patent, 1:22-25. Peters explains how it addresses similar problems for handling the increased amounts of cybersecurity data. *See* EX1004, Peters, 1:33-41. Specifically, Peters seeks to address the “need in the cybersecurity field to create improved systems and methods for intelligently scaling detection capabilities of a security operations service while maintaining quality and improving its technical capabilities to efficiently respond to an increasingly large volume of security risks.” *Id.*, 1:48-53; *see also id.*, 1:37-47. Peters’s Figures 2 and 4 below outline such methods and systems:

SOURCING SECURITY ALERT DATA AND IDENTIFYING LIKELY  
SECURITY ALERTS S210

IDENTIFYING AN AUTOMATED INVESTIGATION WORKFLOW BASED  
ON VALIDATED SECURITY ALERT DATA S220

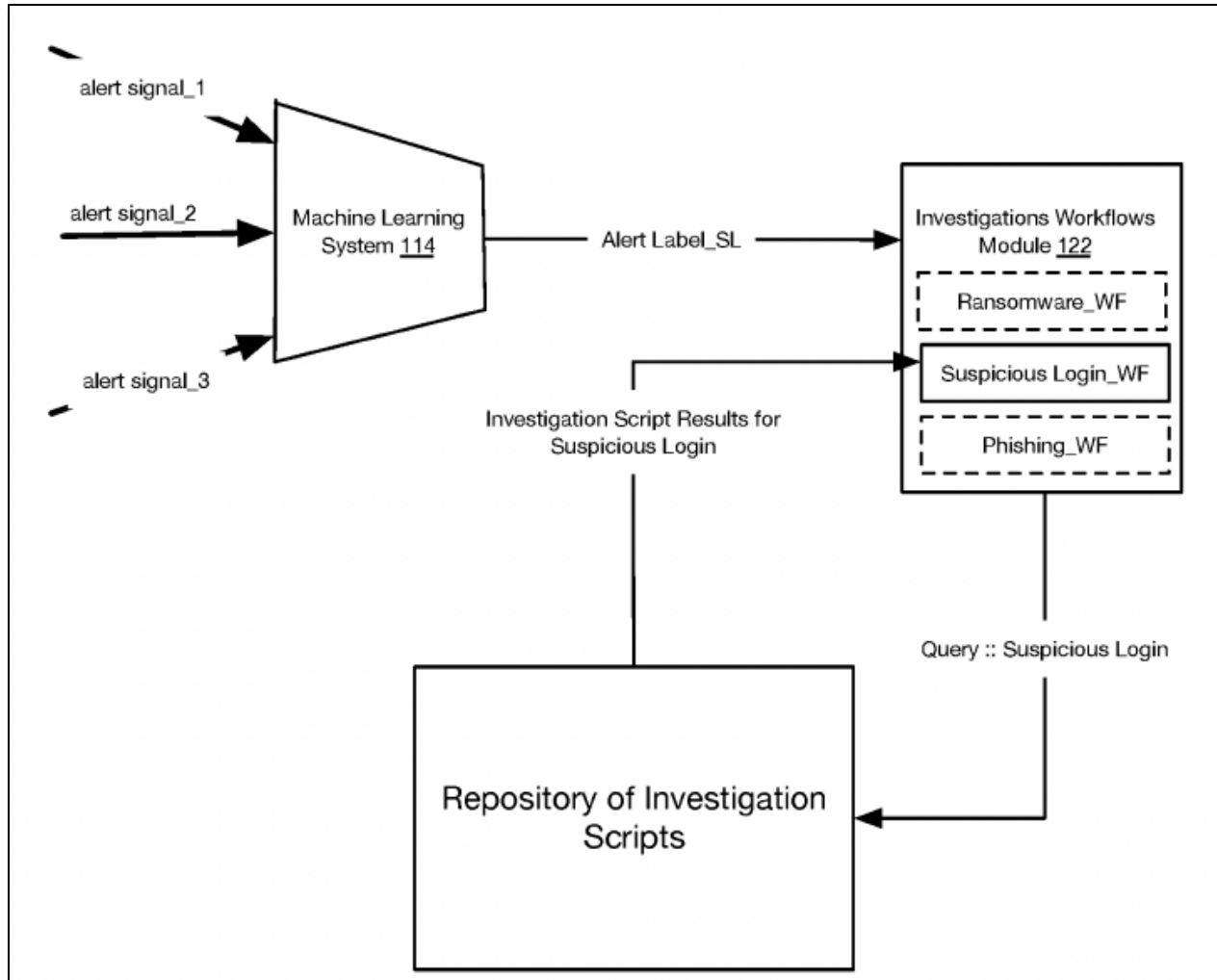
EXECUTING THE AUTOMATED INVESTIGATION WORKFLOW AND  
BUILDING A CORPUS OF INVESTIGATIVE DATA S230

AUTOMATICALLY CREATING THREAT INTELLIGENCE DATA AND  
CREATING THREAT RESPONSE COLLATERAL S240

IDENTIFYING A ROUTING FOR DISPOSING A CYBERSECURITY  
THREAT S250

**FIGURE 2**

*Id.*, FIG. 2.



*Id.*, FIG. 4.

74. For at least these reasons it is my opinion that Peters is in the same field of endeavor as the '549 Patent and is reasonably relevant to a problem with which the inventors of the '549 Patent were involved.

**B. U. S. Patent Application Publication No. 2024/0403428 (“Lal Publication”) (EX1005) and U. S. Provisional Patent Application No. 63/472,227 (“Lal”) (EX1006)**

75. U. S. Patent Publication No. 2024/0403428, titled “System and Method for Utilizing Large Language Models and Natural Language Processing Technologies to Pre-Process and Analyze Data to Improve Detection of Cyber Threats” published on December 5, 2024, and claims priority to U. S. Provisional Patent Application No. 63/472,227, titled “A Cybersecurity System,” filed on June 9, 2023, and which is incorporated by reference. For the purposes of this declaration, my opinions regarding Lal focus on Lal’s Provisional Application (EX1006, Lal). However, when referring to “Lal” in general, I consider both Lal’s patent publication (EX1005) as well as its provisional application (EX1006) as a single unit.

76. In my opinion, Lal is analogous art to the ’549 Patent. First, Lal is in the same field of endeavor of the ’549 Patent, which I have explained to be cybersecurity threat mitigation. Lal is generally directed to an “Artificial Intelligence based cyber security system.” *Id.*, [0004]. Furthermore, Lal is reasonably pertinent to a particular problem addressed by the ’549 Patent, specifically an operator “receiv[ing] an alert that lacks context and information which is ... machine readable but does not immediately convey context, does not provide a root cause, or indicate what, if at all, should be done to remediate, mitigate, and the like.” EX1001, ’549 Patent, 1:52-56. For example, Lal seeks to “enhance

the transparency of AI-based cybersecurity platforms... [to] build [users'] trust and confidence in the AI system's capabilities." *Id.*, [0044]. To do so, Lal describes using one or more large language models for a variety of cybersecurity-related tasks like "translat[ing] questions into API queries," *id.*, [0045], "suggest[ing] remediation actions... after a cyber attack," "mitigat[ing] an ongoing cyber-attack," "act[ing] as a first-line support for client queries," or "generat[ing] code that creates data visualization... to show cyber security breaches." *Id.*, [0017].

77. Additionally, Lal explains that its LLM is trained on several data types relevant to cybersecurity detection and analysis. For example, Lal's "LLM is trained on [] the entire the (sic) cyber security documentation corpus," *id.*, [0039], "the cyber security API," *id.*, [0054], and "existing module structures and event classification," *id.*, [0082], among others. Further, Lal provides its LLMs with prompts generated from templates. *See e.g., id.*, [0033], [0065], [0066].

78. For at least these reasons it is my opinion that Lal is in the same field of endeavor as the '549 Patent and is reasonably relevant to a problem with which the inventors of the '549 Patent were involved.

## **VI. OPINIONS REGARDING GROUND I**

79. In my opinion, Peters in view of Lal renders claims 1-5 and 11-16 obvious.

**A. *Motivations to Combine Peters and Lal***

80. Based on my knowledge of a POSITA at the time of the '549 Patent and the disclosures in both Peters and Lal, it is my opinion that there are at least ten reasons for which a POSITA would have been motivated to combine Peters and Lal.

81. **First**, Lal specifically describes the advantages of employing its cybersecurity-focused LLM within broader mitigation systems, such as those described in Peters. For instance, Lal states that the effectiveness of its LLM “lies in its ability to access and integrate diverse data sources” and to “tap into external data stores, such as threat intelligence platforms and vulnerability databases, to enrich its understanding of the threat landscape.” EX1006, Lal, [0134]. Lal further explains that “[t]he artificial intelligence-based cybersecurity analyst tool can handle data collection and extraction from multiple sources, reducing the manual effort required to gather relevant information for analysis.” *Id.*, [0133].

82. Peters similarly discloses querying external systems, such as “an Active Directory,” as part of its cybersecurity incident response method. EX1004, Peters, 14:44-48. Given this overlap and Peters’s aim of increasing scalability which I explained in §V.A above, it is my opinion that a POSITA would have been motivated to incorporate Lal’s LLM into the cybersecurity system described in Peters to enhance its ability to efficiently analyze and respond to an increased number of cybersecurity threats. In particular, the integration of Lal’s LLM would facilitate

and streamline the system's access to a broader range of data sources, thereby improving overall threat analysis, offering greater scalability, and reducing the manual workload on cybersecurity personnel.

83. **Second**, Lal teaches that its LLM is capable of enhancing threat detection and investigation processes, explaining that the LLM can “augment threat analysis and investigations to handle threat investigations at machine speed and scale.” EX1006, Lal, [0126]. Considering again Peters's expressed desire “for intelligently scaling detection capabilities of a security operations service,” it is my opinion that a POSITA would have been motivated to supplement Peters's system, namely its machine learning classification and query configuration, with Lal's more efficient LLM.

84. **Third**, Lal further discloses that integrating its LLM into cybersecurity platforms can enhance the transparency of AI-based decision-making, explaining that “organizations can enhance the transparency of AI-based cybersecurity platforms.” *Id.*, [0044]. Specifically, Lal explains that “[t]he LLM can interpret user queries, translate them into API queries, and then explain how the AI system arrives at its decisions.” *Id.* According to Lal, this transparency “helps build trust and confidence in the AI system's capabilities, as users can gain insights into the decision-making process and validate the system's outputs.” *Id.*

85. Peters similarly contemplates interaction between a human analyst or administrator and the cybersecurity system: “[t]he security mitigation user interface 130 may function to enable an analyst or an administrator to perform, in a parallel manner, monitoring, investigations, and reporting of security events, incidents, and/or resolutions to subscribers to the system 100[.]” EX1004, Peters, 7:47-51; *see also id.*, 7:51-67.

86. In light of Lal’s express teachings, it is my opinion that a POSITA would have been motivated to incorporate Lal’s LLM into Peters’s system and method to provide the analyst with greater visibility into how decisions are made within the system. It is my opinion that such integration would align with Lal’s express teachings regarding promoting transparency and would enhance user confidence and operational effectiveness through improved interpretability and transparency.

87. **Fourth**, it is my opinion that a POSITA would have been motivated to incorporate Lal’s LLM into Peters’s system, viewing it as a natural extension of Peters’s use of machine learning techniques to ones merely built from larger models. Combining Peters and Lal would have been obvious to try because Peters expressly contemplates using artificial intelligence to enhance the operation of its system, and Lal expressly teaches that its LLM system could have been used to improve methods

of responding to cybersecurity threats. *See, e.g.*, EX1005, Lal, [0009]-[0012]; EX1004, Peters, 6:10-7:3.

88. Peters expressly contemplates the use of artificial intelligence, like machine learning classifiers, to assist in the processing of security alerts. *Id.*, 11:3-6; *see also id.*, 6:18-7:3.

89. Considering the rapid advancements in LLMs following Peters's filing date, it is my opinion that a POSITA working with Peters's machine learning system would have sought to incorporate a more sophisticated type of machine learning models, like an LLM.

90. More specifically, a POSITA working with Peters's system would have sought out an LLM tailored to cybersecurity, like that described in Lal. Considering the compatibility between Peters's AI-based system architecture and Lal's cybersecurity-specific LLM, it is my opinion that a POSITA would have found it obvious to enhance Peters's system with Lal's LLM to improve threat detection, analysis, and response efficiency.

91. **Fifth**, in my opinion, a POSITA would have been motivated to incorporate Lal's LLM into Peters's cybersecurity system to consolidate Peters's ensemble of machine learning classifiers. *See e.g., id.*, 11:3-6, 6:18-7:3.

92. While Peters employs multiple distinct machine learning classifiers, each focused on a specific type of cybersecurity threat, it is my opinion that a

POSITA would have been motivated to substitute that ensemble with Lal's LLM for more flexible and scalable results compared to Peters's specialized classifiers. Because LLMs are built from more data and designed to work generically on a multitude of problems, a POSITA would have recognized the improved cybersecurity threat detection that LLMs would have yielded. It is my opinion that such a combination would have constituted a simple substitution of one known element (*i.e.*, Peters's machine learning) for another (*i.e.*, Lal's LLM) to obtain predictable results.

93. **Sixth**, it is my opinion that a POSITA would have been motivated to incorporate Lal's LLM into Peters's method for configuring API queries.

94. Peters discloses storing a variety of source-specific API configuration parameters used to formulate investigative queries. *See e.g., id.*, 14:55-61. A POSITA would have recognized that, to ensure Peters's parameters remain executable they would need to be regularly maintained. In contrast, Lal teaches an LLM trained with varying API specifications to generate API queries across different systems. *See e.g., EX1006, Lal*, [0045].

95. It is my opinion that utilizing Lal's LLM in place of Peters's static configuration parameters would streamline Peters's query generation and improve adaptability to changing data source requirements.

96. **Seventh**, it is my opinion that a POSITA would have been further motivated to combine Lal's LLM into Peters's system and method due to Lal's highly contextualized, cybersecurity-specific template-style prompts.

97. Namely, Lal describes prompting techniques that include placeholders for populated system-specific data, allowing the LLM to generate responses that are especially tailored to the particular threat or network environment. *See e.g., id., [0031]-[0033].*

98. In my opinion, integrating Lal's LLM and its template-style prompts into Peters's system would enable the LLM to more accurately classify cybersecurity alerts and to generate investigative queries directly responsive to the inbound security alert. It is my opinion that such a POSITA would have understood this combination a simple substitution of one known element (*i.e.*, LLMs and associated template prompts) for another (*i.e.*, Peters's machine learning classifiers and query configuration parameters) to obtain predictable results (*i.e.*, better and faster threat identifications and query generation).

99. **Eighth**, it is my opinion that a POSITA would have been motivated to introduce Lal's disclosed non-transitory storage mediums to Peters in order to store the instructions for Peters's method.

100. **Ninth**, a POSITA would have had a reasonable expectation of success in combining the teachings of Peters and Lal because both Peters and Lal are directed

to well-known cybersecurity systems and methods that were widely understood at the time and could have been implemented using a combination of software and hardware without undue experimentation. For example, Peters teaches a process flow for receiving cybersecurity alerts that can be validated, classified, investigated, and mitigated according to a basic process flow chart provided in Figure 2. EX1004, Peters, Fig. 2.

101. Such a process flow would have been straightforward for a POSITA to understand and could have been implemented using basic hardware and software techniques that were well within the skills of a POSITA at the time. And adding Lal's teaching of an LLM to the artificial intelligence-based system of Peters would have been nothing more than a straightforward implementation of the known technique of training and deploying an LLM. A POSITA would have had a reasonable expectation of success in making such a combination because it would not have required anything beyond the ordinary skills of a POSITA at the time.

102. **Tenth**, combining Peters and Lal would also have been the combination of known methods to yield predictable results. For example, the LLMs disclosed in Lal involved known methods of automating human decision-making tasks, and adding Lal's cybersecurity LLM to the cybersecurity system of Peters would have yielded the predictable result of improving the automation capability of Peters, as expressly taught by Lal. *See, e.g.*, EX1005, Lal, [0006]. Additionally, adding the

large resources of a trained LLM, as taught by Lal, would have yielded the predictable result of allowing the system to be operated by a less-trained human operator that could rely on the understanding of the LLM for guidance rather than requiring highly trained IT personnel to determine how to process and respond to cybersecurity threats.

103. For at least these reasons delineated above, it is my opinion that a POSITA would have been motivated to combine Peters’s cybersecurity system with Lal’s cybersecurity-minded LLM.

**B. Claim 1**

**1. (Pre): “A method for providing cybersecurity incident response, comprising:”**

104. To the extent the preamble is limiting, Peters discloses *a method for providing cybersecurity incident response*. Namely, Peters claims “[a] method for detecting a cybersecurity threat and automatically generating cybersecurity threat intelligence for mitigating the cybersecurity threat...” EX1004, Peters, cl. 1; *see also id.*, Abstract (“[a] cybersecurity system and method for handling a cybersecurity event...”). Additionally, as set forth next, Peters, in view of Lal, teaches every element of claim 1. Thus, it is my opinion that Peters, in view of Lal, renders claim 1 obvious.

**2. Claim 1(a): “receiving an incident input based on a cybersecurity event;”**

105. In my opinion, Peters teaches this limitation.

106. To start, I note that the '549 Patent provides many differing definitions of an incident input. For example, according to the '549 Patent an incident input can be “a natural language query, an unstructured query, an alert, a combination thereof, and the like[.]” EX1001, '549 Patent, 7:41-42. An incident input can also be “a query, a statement, and a combination thereof.” *Id.*, cl. 2, cl. 13, 3:33-34, 4:43-44, 5:58-59. Further, an incident input can “include[] an alert, a natura (sic) query, a combination thereof, and the like.” *Id.*, 14:23-25. An example of an alert provided by the '549 Patent is ““EC2 virtual machine with ID of i-012abcd34efghi56 infected with malware with SHA1 hash of 3395856ce81f2b7382dee72602f798b642f14141’.” *Id.*, 15:49-52. Considering such broad disclosures of what constitutes an incident input, it is my opinion that a POSITA would have understood that such an unstructured alert, like the example here, is an example of a statement.

107. In my opinion, Peters discloses *receiving an incident input*. Peters describes that “S210, which includes sourcing security alert data/event data and identifying likely security alerts, may function to collect or receive event data and raw security data from one or more distinct sources of data.” EX1004, Peters, 8:21-24. Peters goes on to explain that “the event data and/or security alert data may include, but should not be limited to, one or more of vendor alert data, continuous

telemetry data, subscriber data, investigative data, other data, and/or the like.” *Id.*, 8:25-28.

108. Additionally, it is my opinion that Peters teaches that the received *incident input is based on a cybersecurity event*. For example, Peters describes that its “security alert data, in some embodiments, preferably relate[s] to event data or activity data occurring with or involving one or more computing resources, computer network resources, data resources, cloud-based resources, any computer accessible digital entity or device, and/or the like[.]” *Id.*, 8:42-46. Therefore, it is my opinion that a POSITA would have understood the event data received by Peters’s system to be *based on a cybersecurity event* as described by the ’549 Patent.

109. **In sum**, for at least these reasons, it is my opinion that Peters’s receiving a security alert discloses *receiving an incident input based on a cybersecurity event*.

**3. Claim 1(b): “generating a prompt for a large language model (LLM) based on the received incident input;”**

110. In my opinion, Peters in view of Lal discloses this limitation.

111. As I explained in §VI.B.2 above, Peters discloses receiving an *incident input* based on a cybersecurity event.

112. In the following paragraphs, I expand on my explanation of prompts from §III.B above and their use with large language models (LLMs).

113. Prompts are used regularly in combination with LLMs to provide instructions to the LLM that direct the LLM’s output. Other contemporaneous literature supports this understanding and explains that:

A prompt... is a set of instructions provided to an LLM that programs the LLM by customizing it and/or enhancing or refining its capabilities. A prompt can influence subsequent interactions with—and output generated from—an LLM by providing specific rules and guidelines for an LLM conversation with a set of initial rules. In particular, a prompt **sets the context for the conversation and tells the LLM what information is important and what the desired output form and content should be.**

For example, a prompt could specify that an LLM should only generate code that follows a certain coding style or programming paradigm. Likewise, it could specify that an LLM should flag certain keywords or phrases in a generated document and provide additional information related to those keywords. By introducing these guidelines, **prompts facilitate more structured and nuanced outputs** to aid a large variety of software engineering tasks in the context of LLMs.

EX1011, White, 1 (emphases added).

114. Further, prompts can be prepared as templates that include placeholders for queried or otherwise received information. Such template-style prompts are beneficial for many reasons, like “provid[ing] a structure that is predefined which could reduce the risk of leaving out vital information [and being] reusable which could save time and effort.” EX1012, Arvidsson, 10.

115. For further example, the ’549 Patent describes that “[i]n some embodiments, the prompt includes a predefined template. In certain embodiments, the predefined template includes data fields which are modified based on the result.”

EX1001, '549 Patent, 17:38-41. The '549 Patent goes on to provide an example of how such a prompt is generated:

[T]he prompt is generated based on a template such as 'how did RESOURCE\_ID become infected with MALWARE\_ID'. In an embodiment, the data field of 'RESOURCE\_ID' is modified with a value received as a result of executing the database query, and MALWARE\_ID is a data field which is modified based on the incident input. In some embodiments, a plurality of data fields (e.g., both RESOURCE\_ID AND MALWARE\_ID), are modified based on the incident input.

*Id.*, 17:44-52.

116. In my opinion, Lal similarly relies on such template-style prompts to direct the output of its one or more cybersecurity-focused LLMs. *See* EX1006, Lal, [0031]. Lal then provides the following examples of template prompts:

The cyber security system has seen a login from a rare IP in Russia as well as regular events from Cambridge given the following possible actions <SET OF ACTIONS> which would be the best one to take. Explain why you made that decision.

*Id.*, [0033].

Given that this Month we have seen the following autonomous response actions: [block IP 1, block IP 2, block user 3, block user 4] produce me some security recommendations.

*Id.*, [0064].

Summarize John Boy's office365 SaaS events for the last week and decide if his current permissions set which is [X Y Z] is appropriate or whether they can be reduced.

*Id.*, [0066].

117. Lal further describes that the “LLM would be trained on the cybersecurity API and be aware of where information can be found, what kind of data each endpoint returns, and the data the cybersecurity AI analyst considers relevant for similar activity.” *Id.*, [0054].

118. Therefore, it is my opinion that a POSITA would have understood that each of the example prompts provided immediately above are tailored for specific tasks, including but not limited to providing security recommendations or summarizing a user’s recent events. Further, it is my opinion that a POSITA would have understood that Lal’s placeholders like “[X Y Z]” or “<SET OF ACTIONS>” are “data fields” as described by the ’549 Patent that are modified based on the incoming alert or other information relevant to the desired output.

119. Accordingly, it is my opinion that a POSITA would have understood that, in the Peters-Lal combination, Lal’s template-style prompts would incorporate information specific to Peters’s inbound cybersecurity alert into one of Lal’s placeholders in order to further clarify the intended task and desired output for the LLM. Put differently, whereas an AI ML model typically receives one input and produces an (inference) output prediction, an LLM goes through one or more such “conversational” inputs and outputs. *See also* §III.B above.

120. In my opinion, a POSITA would have been motivated to make this combination because a POSITA working with Peters’s system would have

recognized that Peters’s classification of alerts into cybersecurity threat types by “an ensemble of machine learning algorithms” could be handled with greater efficiency if done by a more sophisticated type of machine learning, like Lal’s LLM. *See e.g.*, EX1004, Peters, 6:10-14. A POSITA would have also been motivated to make this combination for the reasons described in §VI.A above.

121. **In sum**, Peters in view of Lal teaches *generating a prompt for a large language model (LLM) (e.g., Lal’s template-style prompts) based on the received incident input (e.g., Peters’s security alert).*

**4. Claim 1(c): “configuring the LLM to generate an output based on the generated prompt;”**

122. In my opinion, Peters in view of Lal teaches this limitation.

123. As I explained in §VI.B.3 above, Lal discloses a *generated prompt*. *See e.g.*, EX1006, Lal, [0033], [0070], [0064], [0066].

124. I first note that the ’549 Patent explains that “the prompt, when processed by the LLM, configures the LLM to output an explanation for the result.” EX1001, ’549 Patent, 17:41-43; *see also id.*, 14:49-51 (“the prompt, when executed by the LLM, configures the LLM to select a scenario from the plurality of scenarios”).

125. As I explained in §VI.B.3 above, a POSITA would have understood that prompts are fed to LLMs to direct their output by providing greater context

regarding the user's intent and desired output. In line with the '549 Patent's teachings that the LLM is "configured" by the prompt, it is my opinion that a POSITA would have understood that Lal's prompts, generated with information specific to the user's request, or if in combination with Peters's system, the inbound cybersecurity alert, would result in the *LLM generating an output based on the generated prompt*.

126. Thus, in view of Lal, Peters renders obvious *configuring the LLM (e.g., Lal's LLM) to generate an output based on the generated prompt (e.g., Lal's template prompt generated with information related to Peters's inbound security alert)*.

**5. Claim 1(d): "mapping the received incident input into a scenario of a plurality of scenarios based on the output of the LLM, wherein each scenario is associated with an incidence response;"**

127. In my opinion, Peters in view of Lal discloses this limitation.

128. I first note that the '549 Patent describes that:

In an embodiment, each scenario corresponds to a cybersecurity situation. For example, a cybersecurity situation is, according to an embodiment, a workload having a malware, a workload generating network traffic with a prohibited domain, combinations thereof, and the like... In an embodiment, the cybersecurity situation includes a cybersecurity risk, such as a workload with an exposed database, a workload connected to a public network (e.g., the Internet), combinations thereof, and the like.

EX1001, '549 Patent, 7:50-61.

129. In my opinion, Peters discloses *mapping the received incident input into a scenario of a plurality of scenarios*. Specifically, Peters describes using an “ensemble of machine learning classifiers that may function to predict at least one of a plurality of distinct cybersecurity threat types for a validated security alert.” EX1004, Peters, 11:4-6. Peters continues to provide examples of cybersecurity threat type classifications, including “suspicious login,” “authorized access,” or “violation of computer security policy.” *Id.*, 11:19-26. Moreover, the appropriate cybersecurity threat type for the target validated security alert is determined by “the distinct machine learning classifier of the ensemble producing a prediction with the highest probability value or confidence value.” *Id.*, 11:29-34. In my opinion, a POSITA would have understood that such a classification correlates to a *scenario of a plurality of scenarios* corresponding to a cybersecurity situation, as described by the ’549 Patent. *See* EX1001, ’549 Patent, 7:50-61.

130. Further, as explained in §VI.A above, it is my opinion that a POSITA would have been motivated to combine Peters’s system with Lal’s cybersecurity-focused LLM because it would have replaced Peters’s several machine learning classifiers with a single LLM. By doing so, a POSITA would have understood that the LLM’s output, rather than the individual machine learning classifier with the highest confidence value, would identify the corresponding cybersecurity threat type

for a particular security alert. As such, the classification of Peters's cybersecurity threat type in combination with Lal's LLM would be *based on the output of the LLM*.

131. Next, Peters teaches *wherein each scenario is associated with an incidence response*. Specifically, Peters describes that “a distinct automated investigation workflow may be defined for each recognized and/or distinct threat type classification or category of a validated security alert. That is, in some embodiments, each class or distinct type of validated security alert may be mapped or electronically linked to a distinct automated investigation workflow.” EX1004, Peters, 12:19-25. As a further example, Peters explains that “a phishing security threat may trigger a phishing automated investigation workflow” or “a suspicious login threat[] may trigger a suspicious login automated investigation workflow.” *Id.*, 12:34-40; *see also id.*, 12:25-34.

132. Peters also describes that:

[A]n automated investigation workflow comprises a sequence and/or a collection of automated investigative actions or investigation subroutines/scripts, that when executed, perform one or more automated investigative tasks including, but not limited to, automatically sourcing security investigation data, automatically routing, and automatically constructing one or more investigation response resources for successfully supporting a decisioning and/or a disposal of a given validated security alert.

*Id.*, 11:62-12:4. Therefore, it is my opinion that a POSITA would have understood Peters's investigation workflow to be *an incidence response*.

133. **In sum**, Peters in view of Lal teaches *mapping the received incident input (e.g., Peters’s security alert) into a scenario (e.g., Peters’s cybersecurity threat type classification) of a plurality of scenarios (e.g., Peters’s plurality of distinct cybersecurity threat types) based on the output of the LLM (e.g., Lal’s LLM output), wherein each scenario is associated with an incidence response (e.g., Peters’s investigation workflow).*

**6. Claim 1(e): “generating a query based on the received incident input and the mapped scenario;”**

134. In my opinion, Peters discloses this limitation.

135. First, it is my opinion that Peters teaches *generating a query*. Namely, Peters describes that S230 “may function to generate one or more investigation queries to the one or more sources requesting additional security alert information and/or details relating to the validated security alert.” EX1004, Peters, 14:15-28; *see also id.*, 14:23-25.

136. Furthermore, it is my opinion that Peters teaches that the generated *query is based on the received incident input and the mapped scenario*. Namely, Peters makes clear that its investigation queries are generated “**based on** the one or more investigation scripts or rules that may define data criteria and/or data requirements for a given **cybersecurity threat type** of the **validated security alert**.” *Id.*, 14:25-28 (emphases added). For example, “if a cybersecurity threat type

includes a suspicious login, one or more of the investigation scripts or rules may define API call parameters for configuring an API call for sourcing data relating to an agent identifier and/or IP address associated with the suspected suspicious login.” *Id.*, 14:28-33.

137. Moreover, Peters explains that “each source of investigations data or cybersecurity threat data may be associated with or otherwise mapped to a set of configuration parameters or requirements for creating a successful API request to the source.” *Id.*, 14:53-56. Additionally, Peters’s “S230 may function to collect, store, and process investigation data based on returned responses to one or more API calls or queries to the one or more distinct sources of security alert data.” *Id.*, 14:62-65.

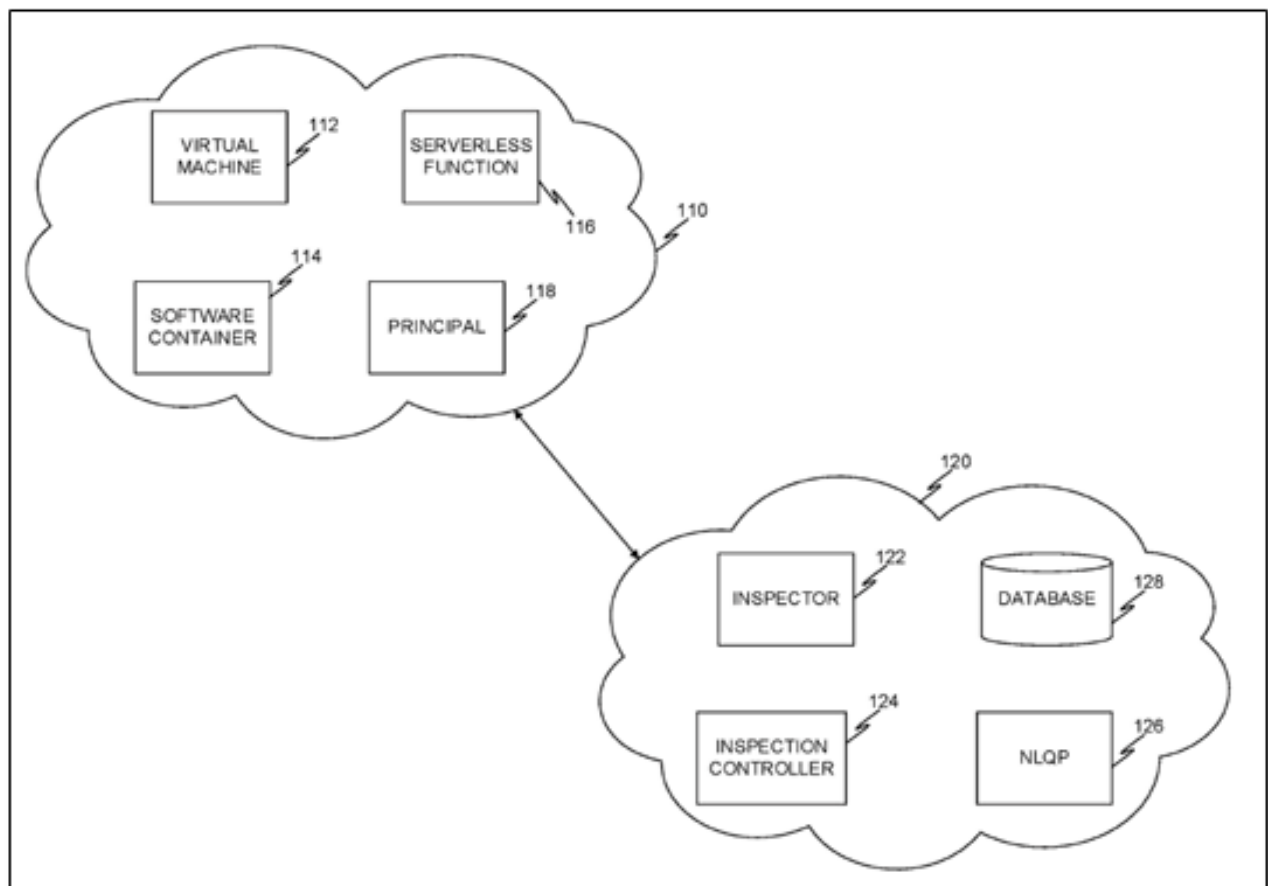
138. Because Peters’s API calls or queries describe “requests” for “sourcing data” that receive “returned responses,” it is my opinion that a POSITA would have understood Peters’s API call to be a *query*.

139. **In sum**, Peters discloses *generating a query* (e.g., Peters’s investigative queries) *based on the received incident input* (e.g., Peters’s security alert) *and mapped scenario* (e.g., Peters’s cybersecurity threat type).

**7. Claim 1(f): “executing the query on a security database, the security database including a representation of a computing environment; and”**

140. In my opinion, Peters teaches this limitation.

141. I first note that the '549 Patent describes that a *security database* can be a “columnar database, a graph database, a structured database, an unstructured database, a combination thereof, and the like.” EX1001, '549 Patent, 9:60-63. The '549 Patent goes on to explain that “[i]n certain embodiments, the security database 128 includes a representation of a computing environment, such as the computing environment 110 of FIG. 1[.]” *Id.*, 10:33-36. Figure 1 of the '549 Patent, which depicts both a *database* 128 and a *computing environment* 110 is shown below:



EX1001, '549 Patent, Figure 1.

142. In my opinion, Peters discloses *executing the query on a security database*. For example, after Peters’s investigation workflow is initiated, it can “configure and execute one or more API calls[.]” EX1004, Peters, 12:54; *see also id.*, 14:58-61 (“S230 may function to access API configuration parameters for each source of investigation data and automatically configure and execute the one or more API calls based on the API configuration parameters.”).

143. Furthermore, Peters describes that its API calls can be executed on “source[s] of security investigation data including, but not limited to, security APIs of third-party security services or vendors and/or native security APIs or native services of a subscriber to the system or service implementing the method[.]” *Id.*, 14:40-44. Peters then goes on to explain that “S230 may function to cause the security investigations engine to collect security investigation data from any suitable source including, but not limited to, an Active Directory and the like.” *Id.*, 14:45-48.

144. First, as I explained in §VI.B.6 above, it is my opinion that a POSITA would have recognized that an API call as described in Peters is equivalent to a *query*.

145. Second, it is my opinion that the security data sources described in Peters teach a *security database including a representation of a computing environment*. For example, Peters notes that security investigation data can be

collected “from any suitable source including, but not limited to, an Active Directory[.]” *Id.*, 14:45-47. Next, I explain why a POSITA would have understood Peters’s Active Directory to be a *database*.

146. Active Directory is a well-known service provided by Microsoft “that stores information about objects on the network.” EX1013, *Active Directory Domain Services Overview*. Such “objects on the network” “typically include shared resources such as servers, volumes, printers, and the network user and computer accounts.” *Id.*, Microsoft also explains that “Active Directory uses a structured data store as the basis for a logical, hierarchical organization of directory information.” *Id.* Notably, Microsoft’s Active Directory is comprised of “the schema[] that defines the classes of objects and attributes contained in the directory...,” “a global catalog that contains information about every object in the directory,” “a query and index mechanism, so that objects and their properties can be published and found by network users or applications[,]” and “a replication service that distributes directory data across a network.” *Id.* A POSITA would have recognized that databases often have a schema to describe the format and structure data logically and/or hierarchically.

147. Other art predating the ’549 Patent similarly describes Active Directory as a database. For example, U.S. Patent Application Publication No. 2018/0004870A1 to Ohayon et al. (“Ohayon”), published in January 2018 describes

that its disclosed machine “initially connects to a directory database 402 (e.g., an Active Directory database for a Microsoft Windows domain network)[.]” EX1014, Ohayon, [0068]. Ohayon further explains that information that can be collected from the Active Directory database includes “information regarding computing devices (e.g., computers and servers) that reside within the network, users of the network (e.g., authorized identities and valid credentials), directory sites, applications/services that make use of the network, network attributes/topology, organization hierarchy, etc.” *Id.* For at least these reasons, it is my opinion that a POSITA would have understood Active Directory to be a *database*.

148. It is also my opinion that a POSITA would have recognized Active Directory to be a *security database* because it stores information related to network security, like user accounts and permissions. For example, a Microsoft article published on July 29, 2021, describes that “[p]asswords at rest are stored in several attributes of the Active Directory database[.]” EX1020, *Passwords Technical Overview*. Another Microsoft article published on April 10, 2023, explains that Active Directory uses “security groups [to] provide an efficient way to assign access to resources on your network.” EX1021, *Active Directory Security Groups*. The article further explains that, through these security groups, a network administrator can “[a]ssign user rights to security groups in Active Directory” or “[a]ssign permissions to security groups for resources.” *Id.* For at least these reasons, it is my

opinion that a POSITA would have recognized that Active Directory is a *security database*.

149. My opinion is further supported by the plain and ordinary meaning of “network,” which suggests that a network is equivalent to a *computing environment*. For example, the Merriam-Webster Dictionary defines “network” as “a system of computers and peripherals that are able to communicate with each other.” EX1019, *Merriam Webster Dictionary Entry for “Network.”* It is my opinion that a POSITA would have recognized that a network is a *computing environment*.

150. Therefore, it is my opinion that a POSITA would have understood Peters’s Active Directory to be a *security database including a representation of a computing environment*.

151. **In sum**, Peters teaches *executing the query* (e.g., Peters’s executing API calls) *on a security database, the security database including a representation of a computing environment* (e.g., Peters’s Active Directory including account passwords and permissions).

**8. Claim 1(g): “initiating a mitigation action based on a result of the executed query.”**

152. In my opinion, Peters teaches this limitation.

153. As I explained in §VI.B.7 above, Peters describes that its method executes API queries on one or more sources of security data. *See e.g.*, EX1004,

Peters, 12:54, 14:58-61. After executing these API queries, Peters explains that “S230 may function to collect, store, and process investigation data based on returned responses to the one or more API calls or queries to the one or more distinct sources of security data.” *Id.*, 14:62-65. The resulting investigation data is then compiled “into a corpus of investigation data.” *Id.*, 14:66-67. From that corpus of investigation data, “cybersecurity intelligence data” is derived. *Id.*, 15:27-28. Specifically,

S240 may function to extract features or pieces of data from the corpus of investigation data that satisfy requirements for responding to the one or more investigative criteria of a given automated investigation workflow and derive the cybersecurity intelligence data based on a compilation of the extracted features or pieces of data from the corpus of investigation data.

*Id.*, 15:61-67.

154. Peters then explains that “S250, which includes identifying a routing for disposing of a cybersecurity threat, may function to identify[ing] (sic) a threat mitigation route or a threat disposal route based on the cybersecurity intelligence data.” *Id.*, 18:4-7. Because Peters’s selection of a threat mitigation route is based on the cybersecurity intelligence data (*i.e.*, particularly probative pieces of data received in response to the executed API queries), it is my opinion that a POSITA would have understood the identification of a threat mitigation route to be *based on a result of the executed query*.

155. I also note that the '549 Patent provides several examples of mitigation actions. For example, “a mitigation action includes generating a notification, generating an alert, updating an alert, generating a severity score, updating a severity score, generating a ticket, generating a risk score, updating a risk score, initiating a remediation action, initiating an incident response, a combination thereof, and the like.” EX1001, '549 Patent, 17:16-21.

156. Peters similarly explains that its threat mitigation routes can include but should not be limited to:

A first route for escalating the validated security alert to a confirmed state of malicious attack or incident, a second route for informing a subscriber regarding a threat risk state of the validated security alert, and a third route for deescalating the validated security alert to a determined state of non-malicious thereby informing a termination state for a validated security alert... It shall be recognized that the routes may include any suitable number or type of routes that be (sic) different than the above-mentioned routes.

*Id.*, 18:13-20. In my opinion, a POSITA would have understood that Peters's threat mitigation routes disclose *a mitigation action* as described by the '549 Patent.

157. Finally, Peters describes that “S250 may function [to] selectively prescribe one of a plurality of distinct threat mitigation and/or threat handling routes to a given validated security alert... or immediately execut[e] a disposal or triage of the validated security alert.” *Id.*, 18:39-44.

158. **In sum**, Peters teaches *initiating a mitigation action* (e.g., Peters's executing a threat mitigation route) *based on a result of the executed query* (e.g.,

Peters's cybersecurity intelligence data derived from results of one or more investigative or API queries).

**C. Claim 2: “The method of claim 1, wherein the incident input includes any one of: a query, a statement, and a combination thereof.”**

159. In my opinion, Peters discloses this claim.

160. First, it is my opinion that Peters discloses *the method of claim 1*, as I explained in §VI.B above.

161. It is also my opinion that Peters discloses *wherein the incident input includes any one of a query, a statement, and a combination thereof*.

162. I note, as explained in §VI.B.2 above, that the '549 Patent describes that an incident input can “include[] an alert, a natura (sic) query, a combination thereof, and the like.” *Id.*, 14:23-25. An example of an alert provided by the '549 Patent is “‘EC2 virtual machine with ID of i-012abcd34efghi56 infected with malware with SHA1 hash of 3395856ce81f2b7382dee72602f798b642f14141’.” *Id.*, 15:49-52. Again, considering such broad disclosures of what constitutes an incident input, it is my opinion that a POSITA would have understood that such an unstructured alert, like the example here, is a statement.

163. In Peters, “[a] validated security alert or a likely security alert, as interchangeably referred to herein, preferably relates to a service-generated indication or notification of a probable or a likely event that may include a

cybersecurity attack, a suspicious cyber event, a breach of cybersecurity policy and/or the like.” EX1004, Peters, 9:5-10. Peters also describes that its security alert engine can “publish one or more likely cybersecurity threats or validated security alerts based on the processing.” *Id.*, 8:33-34. In my opinion, a POSITA would have recognized that such a “service-generated indication or notification” to be “published” is at least a statement like the exemplary alert provided by the ’549 Patent.

164. Other patents published before the earliest possible priority date of the ’549 Patent similarly describe alerts as statements. For example, U. S. Patent No. 8,146,146 to Coviello et al. (“Coviello”) titled, “Method and apparatus for integrated network security alert information retrieval,” provides an example of a “network security alert” in its annotated Figure 3 shown below:

*FIG. 3*

|                                   |                     |           |                                                     |               |             |             |            |  |
|-----------------------------------|---------------------|-----------|-----------------------------------------------------|---------------|-------------|-------------|------------|--|
| A<br>L<br>E<br>R<br>T<br>S<br>(1) | 302                 | 303       | 301<br>304                                          | 305           | 306         | 307         | 308        |  |
|                                   | DATE/TIME           | ALERT ID  | EVENT                                               | SOURCE IP     | SOURCE PORT | DEST. IP    | DEST. PORT |  |
|                                   | 2004/07/08 18:48:44 | A12345678 | SYSTEM SCANNING ACROSS IP<br>ADDRESSES ON SAME PORT | 135.210.96.46 | 137         | 201.1.98.18 | 137        |  |

EX1015, Coviello, Fig. 3 (annotated). Coviello's "event" is described as "system scanning across IP addresses on same port," which a POSITA would have understood to be a statement. For these reasons, it is my opinion that a POSITA would have understood Peters's security alert to be at least *a statement*.

165. **In sum**, Peters in view of Lal discloses *the method of claim 1, wherein the incident input (e.g., Peters's security alert) includes any one of: a query, a statement, and a combination thereof*.

**D. Claim 3: "The method of claim 1, wherein the LLM is trained on any one of: a data schema utilized in representing the computing environment, incident data classified to a scenario, the plurality of scenarios, and a combination thereof."**

166. In my opinion, Peters in view of Lal teaches this claim.

167. First, it is my opinion that Peters in view of Lal discloses *the method of claim 1*, as explained in §VI.B above.

168. Second, in my opinion, Lal discloses *training the LLM*. For example, other contemporaneous references describe LLM training as the LLM "ingesting," being "exposed" to or "given" data. *See, e.g.,* EX1021, Nguyen, 2 ("these models also have to **ingest** huge amounts of data when they're trained."); EX1020, Toloka, 3 ("it's paramount that LLMs go through a pre-training phase, during which each model is **exposed** to vast amounts of data."); EX1022, Drost, 2 ("a large corpus of sentences is collected from various sources and **given** to the model in small

chunks.”) (emphases added). Therefore, it is my opinion that a POSITA would have understood that providing the LLM with such information is a form of training the LLM, as claimed.

169. Third, in my opinion, Peters in view of Lal discloses *the LLM [being] trained on at least a data schema utilized in representing the computing environment*. As I explained in §VI.B.7 above, Microsoft’s Active Directory includes a data schema “that defines the classes of objects and attributes contained in the directory.” EX1013, *Microsoft Active Directory Domain Services Overview*. In my opinion, a POSITA would have recognized that the Active Directory schema discloses a *data schema utilized in representing the computing environment*.

170. Additionally, Lal explains that “[t]he LLM would be trained on the cyber security API and be aware of where information can be found [and] what kind of data each endpoint returns[.]” EX1006, Lal, [0054].

171. Moreover, for the several reasons explained in §VI.A above, it is my opinion that a POSITA would have been motivated to combine Peters’s system with Lal’s LLM. Once Lal’s LLM is incorporated into Peters’s system, which uses an Active Directory as an exemplary data source, a POSITA would have understood that Lal’s LLM would necessarily be trained on at least Active Directory’s *data schema* so that the LLM can be “aware of where information can be found [and] what kind of data each endpoint returns.” *Id.*, [0054].

172. Fourth, it is my opinion that Lal discloses *the LLM [being] trained on at least the plurality of scenarios*. As I explained in §VI.B.5 above, the '549 Patent describes a “scenario” as a “cybersecurity situation includ[ing] a cybersecurity risk, such as a workload with an exposed database [and/or] a workload connected to a public network[.]” EX1001, '549 Patent, 7:50-60.

173. Lal explains that its one or more LLMs are trained on many types of cybersecurity data. *See e.g.*, EX1006, Lal, [0004], [0017], [0039], [0082], [0100]. Lal further explains that “[w]hen provided with a list of breaches and their triggers, an LLM can generate a summary and prioritize them based on specified parameters.” [0051]; *see also id.*, [0052], [0049]. Lal’s one or more LLMs can also be fed “the history of cyber security breaches/models (sic) breaches and their severity scores as well as current cyber threats trending currently[.]” *Id.*, [0063]. In my opinion, a POSITA would have recognized that such a “list of breaches” discloses *the plurality of scenarios* as described by the '549 Patent.

174. **In sum**, Peters in view of Lal teaches *the method of claim 1, wherein the LLM (e.g., Lal’s LLM) is trained on any one of: a data schema utilized in representing the computing environment (e.g., Peters’s Active Directory schema), incident data classified to a scenario, the plurality of scenarios (e.g., Lal’s list of breaches and their triggers or Peters’s distinct cybersecurity threat types), and a combination thereof.*

**E. Claim 4: “The method of claim 3, wherein generating the query further comprises: generating a second prompt for the LLM which when executed by the LLM outputs the query, wherein the second prompt is generated based on any one of: the received incident input, the data schema, the plurality of scenarios, and a combination thereof.”**

175. In my opinion, Peters in view of Lal discloses this claim.

176. As an initial matter, a POSITA would have known that LLMs are designed to be “conversational”—engaging in a series of inputs (e.g., from users) and producing some inference outputs. *See also* §III.B above.

177. First, as I explained in §VI.D above, Lal teaches *the method of claim 3*.

178. Further, it is my opinion that Lal discloses *generating the query*. To start, Lal explains that its one or more LLMs are trained to output queries. *See e.g.*, EX1006, Lal, [0059] (“The cyber security system can query detailed log data in a human-speech way... The LLM understands the intent behind the query and translates it into the appropriate search syntax, simplifying the process of extracting relevant log information for investigations or threat hunting.”).

179. Lal further clarifies that the LLMs “can be prompted to query[.]” *Id.*, [0098]. Notably, “LLMs can act as assistants that submit data to third-party services using pre-engineered prompts. For instance, an LLM can be prompted to query OSINT (Open-Source Intelligence) tools or a customer’s threat intelligence tools by connecting to their APIs. Thus, the LLM is pre-prompted to query services like

OSINT tools or the customer's own datalakes / threat intel tools, using an understanding of how to connect to their APIs.” *Id.*

180. It is also my opinion that Lal discloses *wherein generating the query further comprises: generating a second prompt for the LLM which when executed by the LLM outputs the query.* As explained in the preceding paragraph, Lal's one or more LLMs are “prompted to query.” Further, as explained in §VI.B.3 above, Lal's system generates prompts by inserting dynamic data into templates depending on the task. *See e.g.,* EX1006, Lal, [0031], [0033], [0064], [0066].

181. The first prompt generated by the Peters-Lal system, as further described in §VI.B.3 above, is tasked with classifying Peters's security alert into a variety of cybersecurity threat types. Because different tasks and intended LLM outputs require different prompts, it is my opinion that a POSITA would have understood Lal's prompt tasked with querying to be an additional, or *second prompt*.

182. Such understanding is also supported by Lal's disclosure that its one or more LLMs can cooperate with one another. For example:

The cyber security system can include several components configured to communicate and cooperate with each other including the cyber security appliance with its cyber threat detect engine, the cyber threat autonomous response engine, the cyberattack prediction simulator, the cyber-attack restoration engine, the artificial intelligence-based cyber security analyst, and other components. Large language models (LLMs) that can understand and generate human-like language and the associated generative AI-creating algorithms that can generate new

content based on patterns learned from existing data are used to enhance cyber security measures across various aspects of the field.

EX1006, Lal, [0017]. In my opinion, a POSITA would have understood that LLMs are conversational by nature and that their outputs are more refined when multiple rounds of prompts are incorporated. *See also* §III.B above.

183. Because Lal's components, like the "cyber threat detect engine" and "cyber threat autonomous response engine," can converse with one another, it is my opinion that a POSITA would have understood that Lal's one or more LLMs could build upon a previous prompt (*e.g.*, Lal's first prompt) to further narrow and/or clarify the LLM's ultimate output of a *query*.

184. Next, it is my opinion that Peters in view of Lal discloses *wherein the second prompt is generated based on at least the received incident input*.

185. Peters describes using unique API configuration parameters to generate successful API queries to different data sources. *See* EX1004, Peters, 14:50-56 ("each source of investigations data or cybersecurity threat data may be associated with or otherwise mapped to a set of configuration parameters or requirements for creating a successful API request to the source."). Conversely, and as explained in §VI.B.3 above and in the preceding paragraphs, Lal describes using template-style prompts to generate a prompt for an LLM with dynamic data specific to the desired prompt and/or output. For at least this reason, it is my opinion that a POSITA would

have been motivated to incorporate Lal's template-style prompts and cybersecurity-trained LLM into Peters's configuration of queries. A POSITA would have recognized that implementing Lal's template-style, iterative prompts into Peters's system would result in a more efficient configuration process and ultimately more tailored investigative queries that, for example, can incorporate information specific to Peters's inbound security alert, or *received incident input*.

186. Similarly, it is my opinion that Peters in view of Lal discloses *wherein the second prompt is generated based on at least the data schema*. For example, Lal explains that “[b]y providing the LLM with an API specification an LLM can understand how to make API requests and translate questions into API queries.” EX1006, Lal, [0045]. Furthermore, Lal's LLM “[is] trained on the cyber security API and [is] aware of where information can be found, what kind of data each endpoint returns, and the data the cyber security AI Analyst considers relevant for similar activity.” *Id.*, [0054]. Lal also provides the following example prompt: “Given our API docs find me all devices that communicated over TCP in the last 4 days.” *Id.*, [0046].

187. As explained in §VI.B.7 above, Peters describes Microsoft's Active Directory as an example of an accessible data source. In the proposed Peters-Lal system, a POSITA would have understood that, in order for Lal's LLM to output successful API queries to Peters's Active Directory, for example, the prompt-

template would need to include at least a *data schema* that “defines the classes of objects and attributes contained in the directory[.]” EX1013, *Microsoft Active Directory Domain Services Overview*. In my opinion, a POSITA would have understood that such information pertaining to the data source, like *the data schema* would be incorporated into Lal’s template-style prompts.

188. **In sum**, Peters in view of Lal discloses *the method of claim 3, wherein generating the query (e.g., Lal’s generating an API query) further comprises: generating a second prompt for the LLM (e.g., Lal’s template prompt specific to querying) which when executed by the LLM (e.g., Lal’s one or more LLMs) outputs the query (e.g., Lal’s API query), wherein the second prompt is generated based on any one of: **the received incident input** (e.g., Peters’s inbound security alert), **the data schema** (e.g., Peters’s Active Directory schema), *the plurality of scenarios, and a combination thereof.**

**F. Claim 5: “The method of claim 3, further comprising: training the LLM further on a plurality of database queries, each database query executable on the security database.”**

189. In my opinion, Peters in view of Lal teaches this claim.

190. Peters in view of Lal discloses *the method of claim 3* as discussed in §VI.D above.

191. Additionally, Peters in view of Lal discloses *training the LLM further on a plurality of database queries, each database query executable on the security*

*database*. In particular, Lal discloses that “[b]y providing the LLM with an API specification an LLM can understand how to make API requests and translate questions into API queries.” EX1006, Lal, [0045]; *see also id.*, [0054] (“The LLM would be trained on the cybersecurity API and be aware of where information can be found, what kind of data each endpoint returns, and the data the cybersecurity AI analyst considers relevant for similar activity.”), [0059] (“[t]raining an LLM on the search syntax for tools like the cyber security Advanced Search allows user to query detailed log data in a more user-friendly manner.”).

192. First, as I explained in §VI.B.6 above, it is my opinion that a POSITA would have first recognized that an API query or request is akin to a *database query* because it requests specific data from a data source.

193. It is also my opinion that a POSITA would have known that API specifications often include example queries that, as written, are executable on the particular data source. The inclusion of sample queries in API specifications is commonplace. Some notable examples are RESTAPI (EX1017) and JSONAPI (EX1018), both of which have API specifications featuring example queries. For example, RESTAPI’s API specification as available on February 16, 2023, provides the following example queries:

A simple query example:

```
https://mydb-fafc.restdb.io/rest/customers?q={"status": "GOLD"}
```

Which is the same as asking the database: *fetch all records with status* **GOLD**

## EX1017, RESTAPI Specification, 1.

Return documents that matches one or the other field.

```
https://mydb-fafc.restdb.io/rest/people?q={"$or": [{"name": "Jane"}, {"name": "Donald"}]}
```

## *Id.*, 2.

Return documents not matching the query.

```
https://mydb-fafc.restdb.io/rest/people?q={"name" : {"$not" : "Joe"}}
```

## *Id.*, 1.

194. JSONAPI's specification as available on May 24, 2023, similarly provides several exemplary queries:

```

HTTP/1.1 200 OK
Content-Type: application/vnd.api+json

{
  "links": {
    "self": "http://example.com/articles/1"
  },
  "data": {
    "type": "articles",
    "id": "1",
    "attributes": {
      "title": "JSON:API paints my bikeshed!"
    },
    "relationships": {
      "author": {
        "links": {
          "related": "http://example.com/articles/1/author"
        }
      }
    }
  }
}

```

EX1018, JSONAPI Specification, 20.

```
/?page[offset]=0&page[limit]=10
```

*Id.*, 38.

```

GET /articles HTTP/1.1
Accept: application/vnd.api+json

```

*Id.*, JSONAPI Specification, 18.

195. For these reasons, it is my opinion that a POSITA would have understood that training an LLM with API specifications would include training the LLM on *a plurality of database queries, each query executable* to teach the LLM how to properly configure API queries specific to the target data source.

196. Additionally, it is my opinion that a POSITA would have found it obvious to combine Peters's source of security data, *e.g.*, Active Directory (which I explained to be a *security database* in §VI.B.7 above) with Lal's LLM trained on API specifications and configured to output executable queries. Namely, a POSITA would have found it obvious that, in order for Lal's LLM to output a query compatible with Peters's Active Directory, Lal's LLM would have to be trained on at least an API specification or other search syntax specific to Active Directory.

197. **In sum**, Peters in view of Lal discloses *the method of claim 3, further comprising training the LLM (e.g., Lal's LLM) further on a plurality of database queries (e.g., Lal's API specifications including queries), each database query executable on the security database (e.g., Peters's Active Directory).*

#### **G. Claim 11**

1. **Claim 11(Pre):** “A non-transitory computer-readable medium storing a set of instructions for providing cybersecurity incident response, the set of instructions comprising: one or more instructions that, when executed by one or more processors of a device, cause the device to:”

198. To the extent the preamble is considered limiting, Peters in view of Lal discloses this limitation. Namely, Peters describes a cybersecurity system implemented using “one or more computing servers, a distributed network of computers (*e.g.*, a cloud network), computer processors, and the like.” EX1004, Peters, 6:14-17. In my opinion, a POSITA would have understood that

implementation of Peters's system in a cloud environment necessarily involves a non-transitory computer-readable medium storing the instructions required to execute the cybersecurity incident response method.

199. Lal similarly discloses that “[t]he method and system are arranged to be performed by one or more processing components with any portions of software stored in an executable format on a computer readable medium. Thus, any portions of the method, apparatus and system implemented as software can be stored in one or more non-transitory memory storage devices in an executable format to be executed by one or more processors.” EX1006, Lal, [0319]. A POSITA would have appreciated that, for Peters's method and system to be executed by the described computer processors, the relevant instructions must be stored on non-transitory storage media, such as those explicitly described in Lal.

200. Accordingly, to the extent Peters does not expressly disclose such storage media, it is my opinion that a POSITA would have been motivated to incorporate Lal's non-transitory storage devices to store the software necessary to implement the combined Peters-Lal cybersecurity system as well as for the reasons described in §VI.A above.

201. **In sum**, Peters in view of Lal discloses *a non-transitory computer-readable medium (e.g., Lal's non-transitory storage devices) storing a set of instructions for providing cybersecurity incident response (e.g., Peters's and Lal's*

methods for cybersecurity mitigation), *the set of instructions comprising: one or more instructions that, when executed by one or more processors of a device (e.g., Peters's computer processors), cause the device to*, as claimed.

**2. Claim 11(a): “receive an incident input based on a cybersecurity event;”**

202. *See* §VI.B.2 above.

**3. Claim 11(b): “generate a prompt for a large language model (LLM) based on the received incident input;”**

203. *See* §VI.B.3 above.

**4. Claim 11(c): “configure the LLM to generate an output based on a cybersecurity event;”**

204. *See* §VI.B.4 above.

**5. Claim 11(d): “map the received incident input into a scenario of a plurality of scenarios based on the output of the LLM, wherein each scenario is associated with an incidence response;”**

205. *See* §VI.B.5 above.

**6. Claim 11(e): “generate a query based on the received incident input and the mapped scenario;”**

206. *See* §VI.B.6 above.

**7. Claim 11(f): “execute the query on a security database, the security database including a representation of a computing environment; and”**

207. *See* §VI.B.7 above.

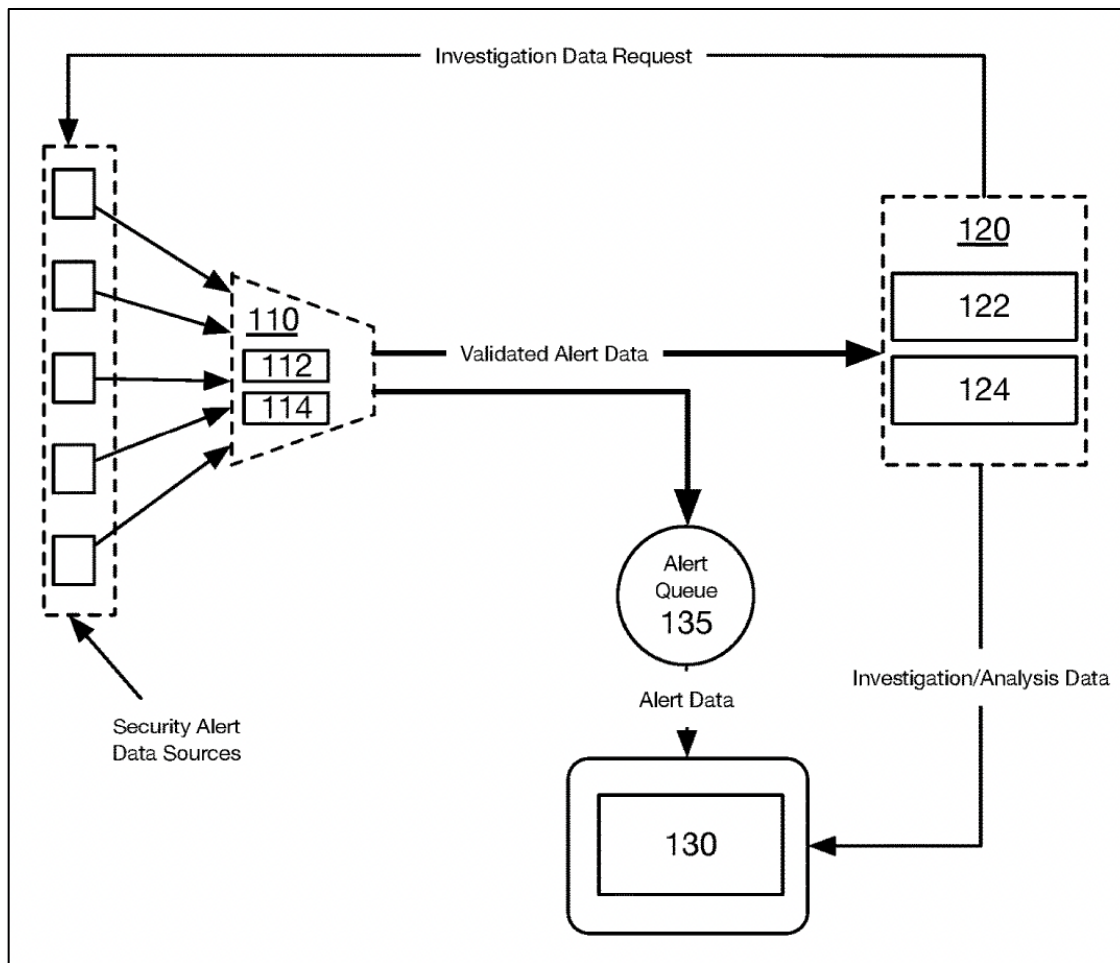
**8. Claim 11(g): “initiate a mitigation action based on a result of the executed query.”**

208. See §VI.B.8 above.

**H. Claim 12**

**1. Claim 12(Pre): “A system for providing cybersecurity incident response comprising:”**

209. To the extent the preamble is limiting, Peters discloses *a system for providing cybersecurity incident response* (e.g., Peters’s cybersecurity threat detection and threat mitigation system): “The system 100 may sometimes be referred to herein as a cybersecurity threat detection and threat mitigation system 100.” EX1004, Peters, 5:39-42. Peters’s cybersecurity threat detection and threat mitigation system 100 is shown in FIG. 1 below:



*Id.*, FIG. 1.

## 2. Claim 12(a): “a processing circuitry;”

210. In my opinion, Peters discloses *a processing circuitry* (e.g., Peters’s computer processors): “Additionally, the security alert machine learning system 114 may be implemented by the one or more computing servers, a distributed network of computers (e.g., a cloud network), computer processors, and the like of the system 100.” EX1004, Peters, 6:13-17; *see also id.*, cl. 15. Lal also describes that “any portions of the method, apparatus and system implemented as software can be stored

in one or more non-transitory storage devices in an executable format to be **executed by one or more processors.**” EX1006, Lal, [0319] (emphasis added).

211. A POSITA would have understood that a processing circuitry is comprised of computer processors and therefore have understood that such computer processors disclose a processing circuitry. It is also my opinion that a POSITA would have been motivated to incorporate Lal’s processing components in the Peters-Lal system to the extent not disclosed in Peters.

**3. Claim 12(b): “a memory, the memory containing instructions that, when executed by the processing circuitry, configure the system to:”**

212. As I explained in §VI.H.2 above, Peters discloses *processing circuitry* (e.g., Peters’s computer processors). *See also* EX1004, Peters, 6:13-17; cl. 15.

213. Lal discloses *a memory containing instructions*: “[a]ny portions of the method, apparatus and system implemented as software can be stored in one or more non-transitory memory storage devices in an executable format to be executed by one or more processors.... The computer readable medium could be, for example, a physical computer readable medium such as semiconductor memory or solid state memory, magnetic tape, a removable computer diskette, a random access memory (RAM), a read-only memory (ROM), a rigid magnetic disc, and an optical disk, such as a CD-ROM, CD-R/W or DVD.” EX1006, Lal, [0319].

214. In my opinion, a POSITA would have recognized that, for Peters's claimed method and system to be carried out by its computer processors, the necessary instructions must be stored in memory. Therefore, a POSITA would have been motivated to include Lal's memory component to store the software needed to implement Peters's cybersecurity system. A POSITA would have also found it obvious and have been motivated to modify Peters's system by integrating one or more LLMs as described by Lal for the many reasons explained in §VI.A above.

215. **In sum**, Peters in view of Lal discloses *a memory (e.g., Lal's memory storage devices) containing instructions (e.g., Peters's algorithms and other instructions for implementing its method) that, when executed by the processing circuitry (e.g., Peters's computer processors), configure the system (e.g., Peters's cybersecurity threat detection and threat mitigation system).*

**4. Claim 12(c): “receive an incident input based on a cybersecurity event;”**

216. *See* §VI.B.2 above.

**5. Claim 12(d): “generate a prompt for a large language model (LLM) based on the received incident input;”**

217. *See* §VI.B.3 above.

**6. Claim 12(e): “configure the LLM to generate an output based on a cybersecurity event;”**

218. *See* §VI.B.4 above.

**7. Claim 12(f): “map the received incident input into a scenario of a plurality of scenarios based on the output of the LLM, wherein each scenario is associated with an incidence response;”**

219. *See* §VI.B.5 above.

**8. Claim 12(g): “generate a query based on the received incident input and the mapped scenario;”**

220. *See* §VI.B.6 above.

**9. Claim 12(h): “execute the query on a security database, the security database including a representation of a computing environment; and”**

221. *See* §VI.B.7 above.

**10. Claim 12(i): “initiate a mitigation action based on a result of the executed query.”**

222. *See* §VI.B.8 above.

**I. Claim 13: “The system of claim 12, wherein the incident input includes any one of: a query, a statement, and a combination thereof.”**

223. *See* §VI.C above.

**J. Claim 14: “The system of claim 12, wherein the LLM is trained on any one of: a data schema utilized in representing the computing environment, incident data classified to a scenario, the plurality of scenarios, and a combination thereof.”**

224. *See* §VI.D above.

**K. Claim 15: “The system of claim 14, wherein the memory contains further instructions that, when executed by the processing circuitry for generating the query, further configure the system to: generate a second prompt for the LLM which when executed by the LLM outputs the query, wherein the second prompt is generated based on any one of: the received incident input, the data schema, the plurality of scenarios, and a combination thereof.”**

225. See §VI.E above.

**L. Claim 16: “The system of claim 14, wherein the memory contains further instructions which when executed by the processing circuitry further configure the system to: train the LLM further on a plurality of database queries, each database query executable on the security database.”**

226. See §VI.F above.

## VII. CONCLUSION

227. 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: June 3, 2025

By:   
\_\_\_\_\_  
Erez Zadok, PhD